



# الذكاء الرقمي 7

الصف السابع  
الفصل الدراسي الثاني







# الذكاء الرقمي 7

## إشراف

أ. منى سالم عوض سالم (رئيس اللجنة)

## تأليف

د. أشرف رضوان رضوان سليمان  
د. حسام فتحي سليمان وهبه  
أ. منى مرزوق مخلد العازمي  
أ. منار مصطفى عبدالحميد جمال  
أ. نورية حسين علي بوحمود  
أ. أحمد فاروق عبد العظيم عبد الرحمن

د. عمرو محمود إبراهيم حبيب

## تصميم

أ. ساره ياسين عبدالله الأمير  
أ. أفراح محمد هدروس البذالي

## إخراج

أ. ساره ياسين عبدالله الأمير

الطبعة الأولى

1447 هـ

2026/2025 م

المراجعة العلمية  
أ. وفاء حمد حمد البصيلي  
أ. فاطمة نجم جاسم الهولي

الفريق المساند  
أ. سنيه محمد علي المؤمن







حضرة صاحب السمو الشيخ مشعل أحمد الجابر الصباح  
أمير دولة الكويت

H.H. Sheikh Meshal AL-Ahmad Al-Jaber Al-Sabah  
Amir Of The State Of Kuwait





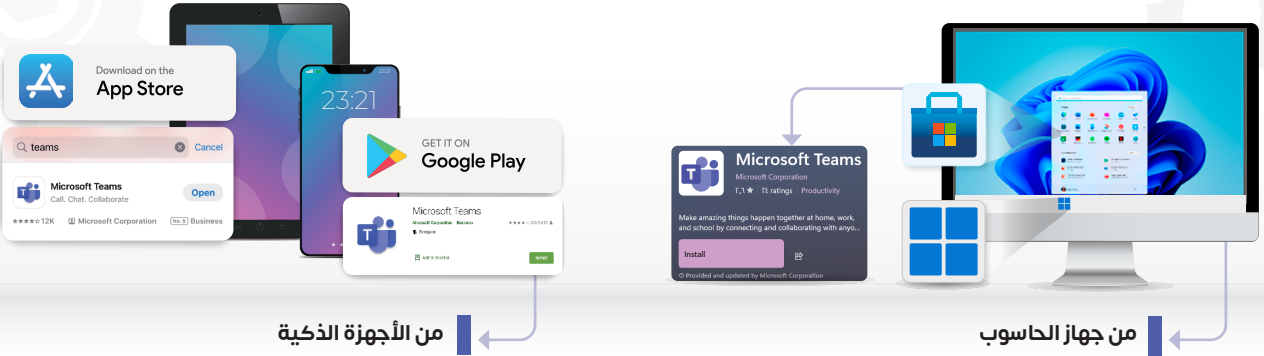
سَمُو الشَّيْخِ صَبَّاحٍ كَهَّالٍ الْحَمَّادِ الصَّبَّاحِ  
وَلِيِّ مَمْلَكَةِ كُوَيْتِ

**H. H. Sheikh Sabah Khaled Al-Hamad Al-Sabah  
Crown Prince Of The State Of Kuwait**

# الفصل الرقمي Digital Classroom

أولاً: تحميل وتثبيت تطبيق Microsoft Teams

الدخول على متجر تطبيقات الأجهزة الرقمية.



من الأجهزة الذكية

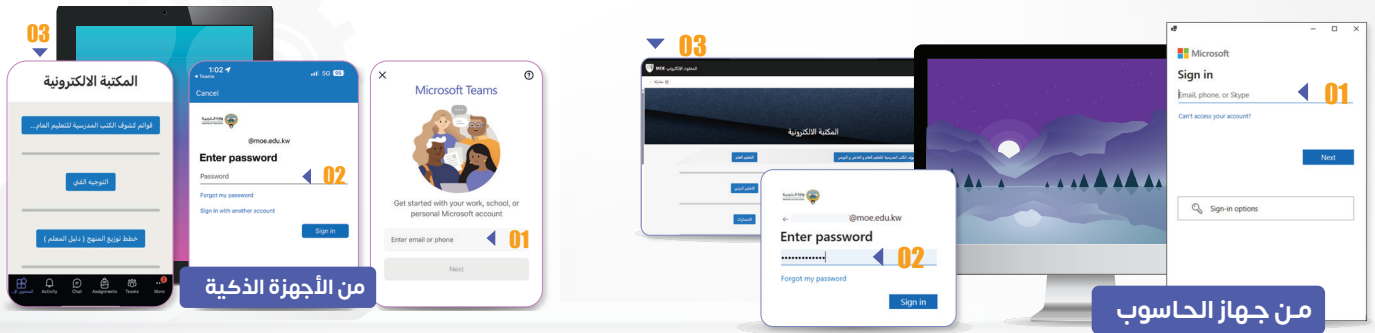
من جهاز الحاسوب

احرص على تحديث تطبيق Microsoft Teams بشكل دوري.

ثانياً: تفعيل Microsoft Teams على الجهاز الرقمي

لتشغيل التطبيق اتبع الخطوات التالية:

- 01 أدخل اسم المستخدم: البريد الإلكتروني الرسمي الخاص بحساب Teams.
- 02 أدخل كلمة المرور الخاصة بحساب Teams.
- 03 تنقل بين واجهات التطبيق مثل المكتبة الإلكترونية و فرق المواد الدراسية.



من الأجهزة الذكية

من جهاز الحاسوب

لا تشارك بياناتك مع أي شخص غير موثوق به.



احتفظ بكلمة المرور في مكان آمن لتسهيل تسجيل الدخول لاحقاً.



من هو حمد؟...

هو مُساعد تعليمي رقمي ذكي يعتمد على تقنيات الذكاء الاصطناعي، مُصمم لتقديم الدعم التعليمي التفاعلي لكافة المناهج الدراسية.

حمد دائماً معك... لتتعلم بثقة وتنجح بامتياز.



مع حمد Chat

## ما هي خدمة حمد شات؟

هي خدمة المحادثة الذكية المقدمة من وزارة التربية في دولة الكويت، تعتمد على الذكاء الاصطناعي، تتمثل وظيفته الأساسية في تسهيل عملية الفهم والاستيعاب من خلال تقديم الشروحات المبسطة، والإجابة على الاستفسارات، وتوجيه المتعلمين خلال مسيرتهم التعليمية.

## أين أجده؟

اكتب استفساراتك، وستلقى الإجابات بشكل فوري.

الضغط على صورة حمد شات



- زيارة الموقع الرسمي  
www.moe.edu.kw  
- أو تطبيق وزارة التربية

03



02



01



# المواطنة الرقمية

يلتزم

بالأمانة الفكرية.

يحترم

الثقافات والمجتمعات  
في البيئة الافتراضية.

يحافظ

على المعلومات  
الشخصية.

يحمي  
نفسه

ضد المعتقدات غير السليمة  
التي تنتشر عبر الوسائط.

يدبر  
الوقت

الذي يقضيه في  
استخدام التكنولوجيا.

يحذر

من التنمر الإلكتروني.

# مفتاح الكتاب

## نواتج التعلم

ما يُتوقع أن يكتسبه المتعلم من معارف، أو يفهمه من مفاهيم، أو يؤديه من مهارات، أو يتبناه من قيم وسلوكيات بنهاية الدرس



## الاستكشاف

ربط المحتوى الدراسي بحياة المتعلم اليومية من خلال مواقف ومشكلات واقعية، تُقدّم بأسلوب مبتكر يعزز شعوره بأهمية ما يتعلمه

## مصادر التعلم

استخدام رمز QR لتيسير وصول المتعلم إلى مصادر تعليمية إضافية تدعم فهمه للمادة العلمية



## التعلم

تقديم المحتوى العلمي بأسلوب مبسّط يناسب المرحلة العمرية للمتعلمين، ويُسهّل فهمهم للمعلومة واستيعابها

## عبّر عن - رأيك -

تدريب المتعلم على التقييم الذاتي لتعزيز قدرته على تحسين أدائه من خلال التعرف على نقاط القوة والضعف، مما يسهّل على ولي الأمر متابعة تقدّمه



## الأنشطة / المهام

مجموعة تدريبات توضح وتُعزّز مفهوم الدرس، مع تشجيع المتعلمين على تنفيذها وابتكار أفكار جديدة من خلالها.

## ملاحظات المعلم

ملاحظات يسجلها المعلم بعد متابعة أداء المتعلم، تعكس مدى فهمه لمفهوم الدرس وتحقيقه للكفاية المطلوبة.



## التطبيق / المشروع

مجموعة ممارسات عملية ممتعة تعزز فهم المادة العلمية وتشجّع المتعلم على تطبيقها بطريقة تفاعلية

## ملاحظات ولي الأمر

تمكّن ولي الأمر من متابعة كتاب المتعلم وتدوين ملاحظاته وتعليقاته لدعم التعلم ومتابعة التقدّم



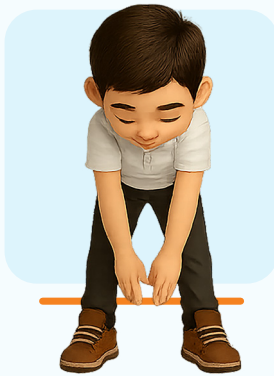
## في وقت فراغك

أنشطة منزلية يُنفذها المتعلم لتعزيز فهمه للمادة العلمية وترسيخ المفاهيم المكتسبة

# صحتك تهمنا



احرص على أداء بعض التمارين أثناء استخدامك جهاز الحاسوب



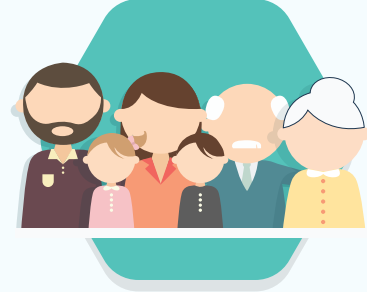
# حياة رقمية آمنة



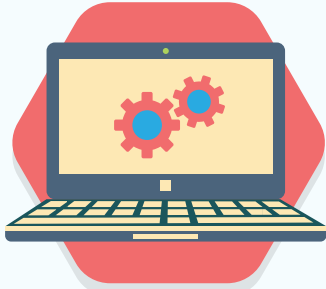
احم أجهزتك الرقمية ببرامج  
الحماية المناسبة



حافظ على التمرينات الرياضية  
بشكل دوري



اقض وقتاً أكبر مع أسرتك بعيداً  
عن العالم الرقمي



تأكد من صيانة وتحديث أجهزتك  
الرقمية بشكل مستمر



استخدم كلمات مرور مركبة



الترزم بأخلاقيات التخاطب  
والمحادثة مع الآخرين



لا تستخدم شبكات إنترنت غير  
موثوقة



تأكد من عدم رفع ملفاتك  
الشخصية على الإنترنت



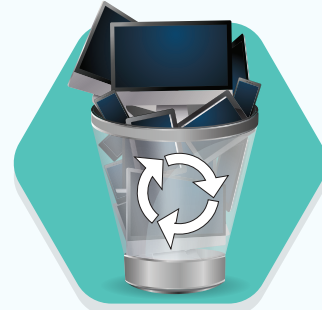
اكتسب معلومة وقدم فكرة كلما  
استخدمت الأجهزة الرقمية



لا تحاول الشراء من الإنترنت  
بمفردك



تأكد من ترتيب التوصيلات الكهربائية  
والأحمال المناسبة



تخلص من الأجهزة الرقمية في الأماكن  
الصحيحة التي تسمح بإعادة التدوير



# المحتوى

## الصفحة

## العنوان

17	المقدمة
	<b>الوحدة الأولى: الأدوات الرقمية Digital tools</b>
21	□ مقدمة عن الذكاء الاصطناعي ومحركات البحث.
33	□ مقدمة في هندسة الأوامر.
47	□ التحقق من المعلومات ومصداقيتها في عصر الذكاء الاصطناعي.
59	□ الحماية الرقمية ومسؤولية استخدام محركات البحث الذكي.
	<b>الوحدة الثانية: المعالجة الرقمية Digital Processing</b>
69	□ برمجة وتطوير الألعاب Game Design and Development.
71	□ برنامج Kodu Game Lab وصناعة الألعاب.
91	□ إضافة التضاريس.
105	□ الماء والمسارات وإعدادات العالم.
119	□ تابع: الماء والمسارات وإعدادات العالم.
131	□ الخطوة الأولى نحو البرمجة.
147	□ إنشاء وتتبع المسار.
171	□ برمجة الفوز والخسارة.
185	□ برمجة الأهداف والتتبع الذكي.
199	□ الحوارات التفاعلية ورفع مستوى اللعبة .
215	□ منظور الكاميرا والمؤثرات الصوتية.
227	□ خصائص متقدمة في اللعبة أكثر احترافية.
	<b>الوحدة الثالثة: المنتجات الرقمية Digital Products</b>
241	□ وحدة المشروعات Projects Unit.
243	□ مشروع برمجي متكامل.
245	□ مشروع لعبة The Kuwaiti Mission.
252	□ إجراءات تنفيذ المشروع.
256	□ ألعاب إلكترونية مقترحة.
262	□ المراجع.



## المقدمة

يشهد العالم اليوم تطوراً متسارعاً في التقنيات الرقمية، مما انعكس بصورة واضحة على مجالات التعليم، وأبرز الحاجة إلى مناهج دراسية تنمي المهارات الرقمية والبرمجية لدى المتعلمين، وتسهم في إعدادهم للتفاعل الواعي والمسؤول مع متطلبات العصر الرقمي.

ويأتي هذا الكتاب انسجاماً مع التوجه الوطني ورؤية دولة الكويت 2035 نحو بناء مجتمع معرفي قائم على الابتكار وتنمية رأس المال البشري، حيث يهدف إلى تزويد المتعلمين بالمعارف التقنية الأساسية، وتعزيز مهارات التفكير المنطقي، والعمل التعاوني، واتخاذ القرار، إلى جانب ترسيخ مفاهيم السلوك الرقمي الآمن.

عزيزي المتعلم، ستخوض من خلال هذا الكتاب رحلة تعليمية تقنية تتضمن ثلاث وحدات تعليمية متكاملة:

### الوحدة الأولى: الأدوات الرقمية Digital Tools

تُغنى بالتعريف بمفاهيم الذكاء الاصطناعي، ومحركات البحث، وأساليب التحقق من المعلومات، وأسس الحماية الرقمية، بما يسهم في بناء وعي رقمي سليم، وتنمية مهارات الاستخدام الآمن والمسؤول للتقنيات الحديثة.

### الوحدة الثانية: المعالجة الرقمية Digital Processing

تهدف إلى تعريف المتعلم بأسس تصميم وبرمجة لعبة إلكترونية، من خلال تعلم كيفية بناء عالم اللعبة، والتحكم في حركة الكائنات وتفاعلها، وتطبيق الأوامر البرمجية الأساسية، وصولاً إلى إنتاج لعبة رقمية بسيطة باستخدام برنامج Kodu Game Lab.

### الوحدة الثالثة: المنتجات الرقمية Digital Products

تركّز على تصميم المنتجات الرقمية من خلال تطوير الألعاب التفاعلية، وتخطيط المراحل، وبرمجة التفاعل، والعمل بالمشاريع الجماعية، وصولاً إلى إنتاج لعبة رقمية متكاملة باستخدام برنامج Kodu Game Lab.



# الذكاء الاصطناعي ومحركات البحث Artificial Intelligence and Search Engines

مقدمة عن الذكاء الاصطناعي ومحركات البحث  
Introduction to AI and search engines

1

مقدمة في هندسة الأوامر  
Introduction to Prompt Engineering

2

التحقق من المعلومات ومصداقيتها في عصر الذكاء الاصطناعي  
Verifying information and its credibility in the AI era

3

الحماية الرقمية ومسؤولية استخدام محركات البحث الذكي  
Digital safety and smart search responsibility

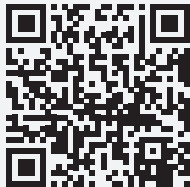
4



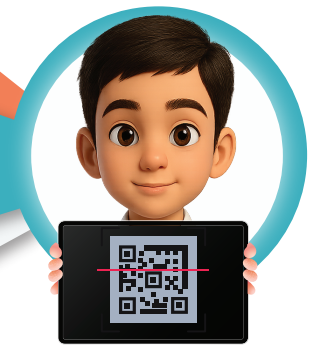
## مقدمة عن الذكاء الاصطناعي ومحركات البحث

### نواتج التعلم

- التعرف على مفهوم الذكاء الاصطناعي ودوره في تحليل البيانات واتخاذ القرارات.
- التمييز بين آلية عمل محركات البحث التقليدية ومحركات البحث المعتمدة على الذكاء الاصطناعي.
- مقارنة بين خصائص محركات البحث التقليدية ومحركات البحث المعتمدة على الذكاء الاصطناعي من حيث الدقة والسرعة والمخاطر والنتائج.
- تحليل الفرق بين المخرجات في نتائج البحث التقليدي والبحث المعتمد على الذكاء الاصطناعي.
- تطبيق أسلوب البحث التقليدي والذكاء الاصطناعي في موقف عملي.
- توضيح مواقف الاستخدام المناسبة لكل نوع من محركات البحث بناءً على طبيعة المهمة.



ملفات أوراق العمل  
ملفات مصادر التعلم



مقولة الدرس:

” إمكانات الذكاء الاصطناعي هائلة حقاً ”



## الاستكشاف



ليس بعد، فقد استغرقت وقتاً طويلاً أبحث في المواقع.

دون فتح روابط! هذه طريقة أسرع بكثير.

هل أنهيت البحث يا حمد؟

أنا استخدمت محركاً يعتمد على الذكاء الاصطناعي، فحصلت على ملخص سريع.

ابحث باستخدام محرك البحث التقليدي Google و محرك بحث يعتمد على الذكاء الاصطناعي ChatGPT عن " طرق التأكد من سلامة الرسائل الإلكترونية حقيقية أم احتيالية"، ثم:

■ سجّل أهم خطوتين للتحقق من الرسائل الإلكترونية.

■ أي طريقة بحث ساعدتك أكثر للوصول إلى المعلومة بسرعة ودقة؟

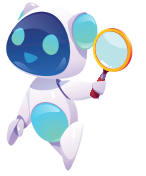


## التعلم



## الذكاء الاصطناعي (Artificial Intelligence)

هو قدرة الأنظمة الحاسوبية على محاكاة الذكاء البشري من خلال التعلم من البيانات، التفكير المنطقي، حل المشكلات، واتخاذ القرارات بطريقة تشبه الإنسان، وذلك لتصميم آلات تستطيع تنفيذ مهام تتطلب عادةً ذكاءً بشرياً مثل التعرف على الصور أو الفهم اللغوي أو حل المشكلات.



## تطور محركات البحث من محرك البحث التقليدي إلى الذكاء الاصطناعي



## كيف تعمل محركات البحث التقليدية ؟

محركات البحث التقليدية مثل (Google - Bing) تمر بمراحل رئيسية أثناء البحث:

### 1. الزحف Crawling:

برامج آلية تسمى "Web Crawlers" تزور المواقع وتجمع محتواها من خلال تتبع الروابط بين الصفحات لتكتشف المحتوى الجديد وتحديث النسخ القديمة.



### 2. الفهرسة Indexing:

تُخزن الصفحات في قاعدة بيانات كبيرة، ويتم تنظيمها حسب الكلمات المفتاحية والعناوين لتسهيل البحث لاحقاً.



### 3. الاستعلام والمطابقة Processing & Matching:

عند استعلام المستخدم عن كلمة أو جملة، يحلل المحرك الكلمات لبحث عنها في الفهرس لتحديد الصفحات المرتبطة بها في الفهرس.



### 4. الترتيب Ranking:

بعد تحديد الصفحات المرتبطة بالاستعلام يتم ترتيب النتائج حسب مدى ارتباطها، جودتها وعدد الروابط التي تشير إليها.



### 5. عرض النتائج Displaying Results:

تُعرض الروابط الأنسب مع مقتطفات قصيرة من محتواها ليسهل على المستخدم الاختيار، يرتب النتائج حسب مدى ارتباطها وجودتها.



#### مميزات محرك البحث التقليدي:

- سريعة وسهلة الاستخدام.
- النتائج واضحة المصدر ويمكن التحقق من مصداقيتها.

#### قيود محرك البحث التقليدي:

- لا تفهم نية المستخدم بدقة.
- تعتمد على الكلمات المفتاحية فقط، وليس على معنى السؤال.

#### النتيجة:

محركات البحث التقليدية تبحث في الكلمات، لا في المعاني، لذا من المهم اختيار كلمات دقيقة عند البحث للحصول على نتائج أفضل.

## كيف تعمل محركات البحث المعتمدة على الذكاء الاصطناعي؟

محركات البحث الحديثة مثل ( ChatGPT و Copilot و Perplexity ) لا تبحث فقط عن الكلمات، بل تحاول فهم المعنى والمقصد من السؤال، من خلال نماذج لغوية مدربة على بيانات ضخمة.

# مقدمة عن الذكاء الاصطناعي ومحركات البحث

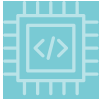
## 1. فهم النية Intent Understanding:

تستخدم تقنيات معالجة اللغة الطبيعية (NLP)<sup>1</sup> لفهم ما يقصده المستخدم، وليس فقط على ما يكتبه حرفياً.



## 2. التمثيل العددي Embeddings:

تُحوّل الكلمات إلى أرقام (vectors) داخل نظام يفهم المعاني المتقاربة، فيتعرف على العبارات المتشابهة حتى لو اختلفت صياغة الكلمات.



## 3. استخراج أو توليد الإجابة Retrieval & Generation:

■ بعض الأنظمة تسترجع معلومات من مصادر حقيقية وتلخصها.  
■ وأخرى تولّد إجابات جديدة تجمع بين عدة مصادر، كما تفعل النماذج اللغوية الكبيرة (LLMs)<sup>1</sup>.



## 4. التعلّم المستمر Feedback & Adaptation:

محركات البحث المعتمدة على الذكاء الاصطناعي لا تبقى كما هي دائماً، تتطور بمرور الوقت من خلال ملاحظات المستخدمين لتصبح أكثر دقة وذكاء لتتكيف مع احتياجات المستخدمين.



### مميزات محركات البحث المعتمدة على الذكاء الاصطناعي:

- تفهم الأسئلة الطبيعية والمعقدة بدقة أكبر.
- تعطي إجابات جاهزة أو ملخّصة مباشرة.
- تجربة استخدام ذكية وسلسة.

### قيود محركات البحث المعتمدة على الذكاء الاصطناعي:

- قد تُنتج أحياناً معلومات غير دقيقة (أخطاء توليدية).
- قلة الشفافية في معرفة مصدر المعلومة.
- تحتاج موارد حوسبة قوية ومكلفة.

### النتيجة:

محركات البحث المعتمدة على الذكاء الاصطناعي تفهم المعنى وتُجيب بذكاء، بينما المحركات التقليدية تبحث عن الكلمات فقط.

<sup>1</sup> Natural Language Processing : معالجة اللغة الطبيعية بهدف تمكين الحاسوب من فهم اللغة البشرية (المكتوبة أو المنطوقة)، وتحليلها، والتفاعل بها بطريقة تشبه تواصل الإنسان المعتمد على الذكاء الاصطناعي Large Language Models : نوع من برامج الذكاء الاصطناعي تستطيع فهم اللغة البشرية وإنشاء نصوص جديدة بطريقة تشبه أسلوب الإنسان في الكتابة والتحدث.

## الفرق بين محركات البحث التقليدية و البحث المعتمد على الذكاء الاصطناعي

محرركات البحث المعتمدة على الذكاء الاصطناعي	العنصر	محرركات البحث التقليدية
تفهم معنى السؤال ونية المستخدم باستخدام الذكاء الاصطناعي	طريقة العمل	تبحث في الكلمات المفتاحية الموجودة في الصفحات
يعتمد على المعنى والسياق	نوع البحث	يعتمد على الكلمات فقط
إجابات جاهزة أو ملخصة، مع إمكانية ذكر المصادر عند السؤال عنها	النتائج	قائمة روابط من مواقع مختلفة
أبطأ نسبياً لأنها تحلل وتولد الإجابة	السرعة	سريعة جداً لأنها تبحث في فهرس ثابت
عالية، تفهم اللغة الطبيعية والأسئلة المعقدة	الدقة في الفهم	محدودة، لا تفهم نية المستخدم تماماً
أقل، أحياناً لا تُعرض مصادر واضحة للمعلومة: قد يولد مصدر غير حقيقي	الشفافية	عالية، المستخدم يرى المصادر الأصلية مباشرة
عال، قد تُولد معلومات غير صحيحة أحياناً	احتمال الخطأ	منخفض، لأن النتائج من مواقع حقيقية
تجربة ذكية وتفاعلية وسهلة	تجربة المستخدم	يعتمد على اختيار المستخدم للروابط المناسبة
ممتازة لفهم الأسئلة المعقدة وتقديم إجابات ذكية	نوع البحث	جيدة للبحث عن معلومات محددة وسريعة

استخدم محركات البحث التقليدية عندما تبحث عن مواقع ومصادر مباشرة، واستخدم محركات البحث المعتمدة على الذكاء الاصطناعي عندما تحتاج فهماً أعمق وإجابات جاهزة.

## النشاط



استكشف كيف يبني الذكاء الاصطناعي كويت المستقبل الذكية، ابحث باستخدام محرك بحث تقليدي Google ومحرك بحث آخر يعتمد على الذكاء الاصطناعي ChatGPT عن أفكار لمشروعات ذكية تساعد في تطوير كويت المستقبل (مثل النقل الذكي، الطاقة النظيفة، أو التعليم الذكي)، ثم:

1. سجّل أهم فكرتين وجدتهما يمكن أن تساهم في بناء كويت المستقبل:  
■ الأفكار التي ظهرت في محرك البحث التقليدي Google.

■ الأفكار التي ظهرت في محرك البحث المعتمد على الذكاء الاصطناعي ChatGPT.

2. عبر عن رأيك أي نوع من محركات البحث ساعدك أكثر في الوصول إلى الأفكار بسرعة ووضوح؟ ولماذا؟

## التطبيق



### ورقة عمل 1: اكتشاف كيف يحافظ الذكاء الاصطناعي على البيئة في كويت المستقبل

ابحث باستخدام محرك بحث تقليدي Google ومحرك بحث آخر يعتمد على الذكاء الاصطناعي ChatGPT عن طرق استخدام الذكاء الاصطناعي في حماية البيئة في الكويت، ثم:

1. سجّل أهم فكرتين وجدتهما يمكن أن تساعد بهما في الحفاظ على البيئة:  
■ الأفكار التي ظهرت في محرك البحث التقليدي Google.

■ الأفكار التي ظهرت في محرك البحث المعتمد على الذكاء الاصطناعي ChatGPT.

2. اختر أي نوع من محركات البحث كان أكثر فائدة في الحصول على المعلومات الدقيقة، وبرّر اختيارك:

# مقدمة عن الذكاء الاصطناعي ومحركات البحث

ورقة عمل 2: صمم فكرة ذكية لخدمة المجتمع في كويت المستقبل

استخدم محرك بحث تقليدي Google ومحرك بحث آخر يعتمد على الذكاء الاصطناعي ChatGPT للبحث عن أمثلة لمشروعات رقمية أو تطبيقات تخدم المجتمع الكويتي، ثم:

1. استخرج فكرتين يمكن تطويرهما لتصبحا مشروعاً يخدم سكان الكويت في المستقبل:

2. اختر الفكرة الأفضل برأيك، واكتب سطرًا يوضح كيف يمكن للذكاء الاصطناعي أن يجعلها أكثر فاعلية:

## في وقت فراغك



استكشف كيف يجعل الذكاء الاصطناعي الحياة اليومية في الكويت أسهل، استخدم محرك بحث تقليدي Google ومحرك بحث آخر يعتمد على الذكاء الاصطناعي ChatGPT للبحث عن طرق استخدام الذكاء الاصطناعي في تسهيل حياة الناس في الكويت (مثل المساعدات الصوتية، الخدمات الحكومية الذكية، أو الرعاية الصحية الرقمية)، ثم:

1. اذكر مثالين لتطبيقات أو خدمات ذكية يمكن أن تحسّن حياة المواطنين والمقيمين في دولة الكويت:

2. قارن بين نتائج المحركين، وبيّن أيهما ساعدك أكثر في الوصول إلى المعلومة بسرعة ووضوح:



## عبر عن رأيك



أستخدم محركي البحث التقليدي والمعتمد على الذكاء الاصطناعي للبحث والحصول على المعلومات.

أقارن بين نتائج البحث التقليدي والبحث الذكي.

أميز الفرق بين نتائج البحث التقليدي ونتائج البحث الذكي.

أحدد الطريقة التي ساعدتني أكثر على الفهم والوصول للمعلومة بدقة.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

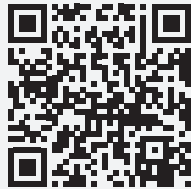
## ملاحظات ولي الأمر



# مقدمة في هندسة الأوامر

### نواتج التعلم

- تعريف مفهوم هندسة الأوامر وأهميتها في تحسين مخرجات أنظمة الذكاء الاصطناعي.
- بيان فوائد هندسة الأوامر في تحسين جودة المخرجات.
- تحليل مكونات الأمر الجيد من حيث السياق والمهمة والمخرجات المتوقعة.
- تطبيق عناصر الصياغة الجيدة عند كتابة Prompt فعال.
- تحسين صياغة الأوامر بناءً على التغذية الراجعة.
- تقييم جودة الأوامر من خلال تحليل مدى وضوحها وارتباطها بالمخرجات المطلوبة.



ملفات أوراق العمل  
ملفات مصادر التعلم



### مقولة الدرس:

”الذكاء الاصطناعي هو التكنولوجيا التي تجعل الآلات تتصرف بطريقة تبدو ذكية”



## الاستكشاف



استعن بمحرك بحث يعتمد على الذكاء الاصطناعي، ثم:

ابحث باستخدام الأمر التالي: " اكتب كل شيء عن الذكاء الاصطناعي وحماية البيئة "،  
ثم أجب عن السؤال التالي:

■ هل ساعدك محرك البحث في الوصول إلى إجابة دقيقة وواضحة عن طرق حماية البيئة في  
دولة الكويت ؟

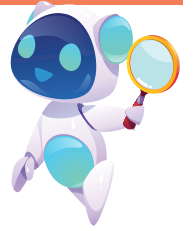
■ الآن لنقم بصياغة الأمر بصورة دقيقة وواضحة بكتابة الأمر التالي: " اذكر طريقتين  
يستخدم فيهما الذكاء الاصطناعي لحماية البيئة في دولة الكويت " ، ثم سجل  
النتيجة التي حصلت عليها.

## التعلم



### هندسة الأوامر (Prompt Engineering)

هي عملية إعداد وصياغة التعليمات الموجهة إلى أنظمة الذكاء الاصطناعي بطريقة مدروسة تساعد على فهم المطلوب بدقة وتقديم نتائج صحيحة ومناسبة. وتعتمد هذه العملية على تجربة وتحسين صياغة الأوامر بشكل متكرر حتى يتم الوصول إلى أفضل استجابة ممكنة من النظام.



## فوائد هندسة الأوامر Prompt Engineering

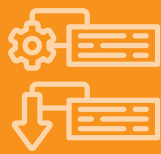
لأن نماذج (LLMs) لا تفهم السياق كما يفهمه البشر، فإن طريقة صياغة الأمر تؤثر بشكل كبير على جودة النتائج، فهندسة التوجيه الجيدة تساعد على:

### التحكم والتخصيص:

تساعد الأوامر الجيدة المستخدم على توجيه سلوك الذكاء الاصطناعي والتحكم في نتائجه من حيث:

#### عمق الرد:

نظرة عامة مختصرة  
أو شرح مفصّل.



#### هيكل الرد:

مقال، جدول،  
ملخص.



#### أسلوب الرد:

رسمي، ودي،  
أكاديمي.



### الإبداع وتوليد الأفكار:

يمكن للأوامر الجيدة أن تُحفّز الإبداع وتساعد على توليد أفكار جديدة في الكتابة أو التدريس أو حل المشكلات.

## الكفاءة و الإنتاجية:

تساعد الأوامر الجيدة على توفير الوقت من خلال:

1. تنفيذ المهام الكتابية المتكررة تلقائياً.
2. تلخيص النصوص أو إعادة صياغتها بسرعة.
3. إنشاء تقارير منظمة أو أكواد برمجية قصيرة.

## سهولة الوصول إلى قدرات الذكاء الاصطناعي:

تسهّل الأوامر الجيدة استخدام الأنظمة المعقدة، بحيث يمكن لأي شخص الاستفادة منها دون الحاجة إلى معرفة مسبقة بالبرمجة، فمجرد كتابة توجيه واضح يكفي للحصول على قدرات متقدمة مثل التحليل أو الترجمة.

## وسيلة فعالة للتعليم والتعلم:

تمكين المتعلم من تصميم أوامر لمحاكاة الدرس أو توليد الأسئلة التفاعلية أو تبسيط المفاهيم حسب مستوى المتلقي.

## محددات صياغة الأوامر Limitations of Prompt Formulation

### الغموض في الصياغة:

إذا كان التوجيه غامضاً أو ناقصاً، قد يُنتج الذكاء الاصطناعي إجابات غير دقيقة أو غير مرتبطة بالموضوع.

### التحيّز أو عدم الدقة:

حتى مع وجود توجيه مثالي، قد يُنتج الذكاء الاصطناعي معلومات غير صحيحة أو متحيزة بسبب طبيعة البيانات التي تم تدريبه عليها.

### غياب الفهم الحقيقي:

الذكاء الاصطناعي لا يفهم المعنى كما يفعل الإنسان، بل يعتمد على التنبؤ الإحصائي بالكلمات، لذلك قد يبدو واثقاً من إجابته رغم كونها غير صحيحة.

## الحساسية لتغيير الكلمات:

التغييرات البسيطة في الصياغة قد تؤدي إلى اختلاف كبير في النتائج.

## الاعتماد على جودة النموذج:

حتى أفضل التوجيهات قد تفشل إذا كان النموذج المستخدم ضعيفاً أو قديماً، إذ تتأثر النتائج بإمكانات الذكاء الاصطناعي نفسه.

## المخاطر الأخلاقية والخصوصية:

قد تؤدي التوجيهات الغير مدروسة إلى كشف معلومات شخصية أو حساسة عن طريق الخطأ، خاصة عند التعامل مع نماذج عبر الإنترنت.

## عناصر الصياغة الجيدة في هندسة الأوامر

### 1. الوضوح والدقة Clarity & Precision:

يجب أن تكون الأوامر واضحة، تُحدّد بالضبط ما تريده من محرك البحث، وتجنب الغموض أو التعميم.

“بدلاً من كتابة الأمر ”اكتب عن مستقبل الكويت“ استخدم الأمر ”اكتب فقرة من 200 كلمة تشرح كيف يمكن لمشروعات الطاقة الشمسية أن تدعم تحقيق رؤية الكويت 2035 نحو التنمية المستدامة“.

مثال

### 2. السياق والمعلومات المساعدة Context / Background:

السياق هو الإطار الذي يوضّح من المتحدث، ولماذا يتحدث، ولمن يوجّه حديثه، وفي أي موقف يُقدّم الطلب، أما المعلومات المساعدة فهي تفاصيل إضافية تمكّن الذكاء الاصطناعي من فهم البيئة والمجال والجمهور المستهدف بدقة.

”أنت خبير في التنمية البيئية تعمل ضمن فريق تخطيط رؤية الكويت 2035، ومهمتك إعداد مقترحات لتقليل الانبعاثات الكربونية باستخدام الذكاء الاصطناعي“.

مثال

### 3. تحديد الدور أو الشخصية :Role / Persona Prompting

توجيه الذكاء الاصطناعي ليتصرف كما لو كان شخصاً أو خبيراً في وظيفة محددة مثل معلم، مصمم، مبرمج، أو متعلم، يساعد هذا في جعل الإجابات أكثر واقعية وتخصصاً بما يناسب المهمة المطلوبة.

”كن مهندس تخطيط حضري في الكويت، وضح كيف يمكن لتقنيات المدن الذكية تحسين جودة الحياة في مدينة الحرير المستقبلية“.



### 4. تنسيق المخرجات/ القيود Output Format / Constraints

تنسيق المخرجات (Output Format) يعني تحديد الشكل والطريقة التي ترغب أن تُقدّم بها الإجابة مثل: جدول، قائمة، فقرة، عرض تقديمي، كود برمجي أما القيود (Constraints) فهي الحدود أو الشروط التي تضبط حجم أو أسلوب المخرجات، مثل عدد الكلمات أو اللغة أو مستوى الصعوبة.

”قدّم إجابتك في جدول من عمودين: العمود الأول يوضح التحدي البيئي في الكويت، والعمود الثاني يقدم حلاً مقترحاً يعتمد على الذكاء الاصطناعي. لا تتجاوز 5 صفوف“.



### 5. إعطاء أمثلة Few-Shot / Examples

تضمين مثال أو أكثر داخل الأمر ليوضح لمحرك الذكاء الاصطناعي الأسلوب أو النمط المطلوب في الإجابة، فبدلاً من الاكتفاء بالوصف، نعرض مثالاً عملياً يساعده على محاكاة الطريقة المطلوبة بدقة.

”التحدي هو ارتفاع درجات الحرارة في الكويت، والحل هو استخدام أنظمة ذكية لإدارة الطاقة في المباني. الآن، قدّم مثالين آخرين لتحديات بيئية في كويت المستقبل مع حلول ذكية مشابهة“.



## 6. التكرار والتحسين Iterative Refinement:

التكرار والتحسين هو عملية تعديل الأمر خطوة بخطوة بناءً على جودة الإجابة التي يقدمها الذكاء الاصطناعي، بهدف تحسين النتيجة تدريجياً حتى تصل إلى المستوى المطلوب من الدقة والجودة.

”الطلب الأول: اشرح دور الذكاء الاصطناعي في التعليم بالكويت.“  
بعد المراجعة والتحسين: ” اشرح دور الذكاء الاصطناعي في تطوير التعليم الرقمي في مدارس الكويت بحلول عام 2035، مع ذكر مثال تطبيقي.“

مثال

## 7. تشجيع التفكير خطوة بخطوة Chain of Thought Prompting:

أسلوب لصياغة الأمر يدفع النموذج إلى تفكيك المهمة إلى خطوات صغيرة متتابعة مع عرض القرارات بين الفكرة والنتائج بهدف لتقليل الغموض والأخطاء وزيادة قابلية التحقق.

” اشرح خطوة بخطوة كيف يمكن لدعم المشاريع الصغيرة والمتوسطة أن يسهم في تنويع الاقتصاد الكويتي ضمن رؤية الكويت 2035، بدءاً من التمويل، ثم الابتكار، ثم التوظيف.“

مثال



## دليلك لكتابة أمر Prompt فعال يوجّه الذكاء الاصطناعي بدقة

يساعدك هذا القالب على تنظيم أفكارك وكتابة أوامر واضحة ومحددة للذكاء الاصطناعي، لتصل في كل مرة إلى نتائج دقيقة ومبدعة تدعم مشاريعك المستقبلية.

السؤال الموجه	العنصر	الترتيب
ما المطلوب تحديداً من الذكاء الاصطناعي؟	الوضوح والدقة	1
من أنت؟ ولماذا تطلب هذه المهمة؟	السياق والمعلومات المساعدة	2
من تريد أن يكون الذكاء الاصطناعي؟ (خبير، مستشار، معلم، ..).	تحديد الدور	3
كيف تريد أن تكون النتيجة؟ (فقرة، قائمة، جدول، ..).	تنسيق المخرجات	4
هل يمكنك إعطاء نموذج صغير مشابه؟	مثال توضيحي	5
هل تطلب مراجعة أو اقتراحات إضافية؟	التكرار والتحسين	6
كيف تريد أن يعرض النتائج؟ (بالسلسل، بالشرح، بالمقارنة، ..).	التفكير خطوة بخطوة	7

### النتيجة النهائية:

تصرّف كخبير في ريادة الأعمال المستقبلية، وساعدني كمتعلم في المرحلة المتوسطة في دولة الكويت، على ابتكار فكرة مشروع مستقبلي لعام 2035 يخدم محور الاقتصاد المعرفي. قدم الإجابة في قائمة من خمس نقاط تتضمن اسم المشروع، فكرته، التقنية المستخدمة، الجمهور المستهدف، والأثر على رؤية الكويت 2035. وضح التفكير خطوة بخطوة، ثم اقترح تحسينين لتطوير الفكرة «مشروع EcoCharge لتخزين الطاقة الشمسية في المنازل الذكية عام 2035».



## النشاط

اكتب Prompt يوجّه محرك الذكاء الاصطناعي ليساعدك على تصميم مدينة سكنية كويتية ذكية عام 2035 م تستخدم الطاقة المتجددة وتقلل الازدحام والتلوث.

السؤال الموجه	العنصر	الترتيب
تصميم مدينة سكنية كويتية ذكية لعام 2035 تعتمد الطاقة المتجددة وتقلل الازدحام والتلوث.	الوضوح والدقة	1
أنا متعلم في المرحلة المتوسطة في دولة الكويت أعمل على مشروع عن كويت المستقبل.	السياق والمعلومات المساعدة	2
خبير تخطيط حضري مستدام ومستشار نقل ذكي.	تحديد الدور	3
ملخص + قائمة بالعناوين + جدول (التحدي/ الحل/ التقنية/ مؤشر الأداء) + خارطة طريق من 3 مراحل.	تنسيق المخرجات	4
<b>مثال:</b> حافلات كهربائية عند الطلب، أو خلايا شمسية لتوليد طاقة نظيفة.	مثال توضيحي	5
اطرح سؤالين واقترح ثلاث تحسينات قابلة للتنفيذ.	التكرار والتحسين	6
عرض النتيجة بالتسلسل المنطقي (تحديات - أهداف 2035 - حلول وتقنيات - أثر مؤشرات نجاح).	التفكير خطوة بخطوة	7

### النتيجة النهائية:

تصرف كخبير تخطيط حضري مستدام ومستشار نقل ذكي بخبرة في مدن الخليج. أنا متعلم في المرحلة المتوسطة في دولة الكويت أعمل على مشروع بعنوان كويت المستقبل، وأريد منك المساعدة في تصميم تصور لمدينة سكنية كويتية ذكية عام 2035 م تعتمد على الطاقة المتجددة وتعمل على تقليل الازدحام والتلوث.

قدم الإجابة على شكل ملخص قصير، يتبعه قائمة بعناوين رئيسية تشمل الطاقة المتجددة، النقل الذكي، البنية التحتية الرقمية، وجودة الهواء، ثم جدول بأربعة أعمدة (التحدي - الحل - التقنية - مؤشر الأداء)، وأضف خارطة طريق من ثلاث مراحل زمنية لتطبيق الفكرة.

أعرض النتائج بالتسلسل المنطقي من التحديات إلى الحلول وأثر مؤشرات النجاح، اختتم باقتراح سؤالين توضيحيين وثلاث تحسينات ممكنة لتطوير التصور.

## التطبيق



### ورقة عمل 3 : توجيه الذكاء الاصطناعي لتخيل مدارس الكويت في المستقبل.

اكتب Prompt يوجّه محرك الذكاء الاصطناعي لتخيل كيف ستكون عليه المدارس الكويتية في المستقبل، واقترح تطبيق أو منصة تعليمية ذكية تساهم في تطوير التعلم:

السؤال الموجه	العنصر	الترتيب
	الوضوح والدقة	1
	السياق والمعلومات المساعدة	2
	تحديد الدور	3
	تنسيق المخرجات	4
	مثال توضيحي	5
	التكرار والتحسين	6
	التفكير خطوة بخطوة	7

### النتيجة النهائية:

---

---

---

---

---

## ورقة عمل 4 : صمّم فكرة ذكية لخدمة المجتمع في كويت المستقبل.

اكتب Prompt يوجه الذكاء الاصطناعي لتخطيط مشروع يحمي بيانات المواطنين في المدن الذكية من الهجمات الإلكترونية عام 2035 م.

السؤال الموجه	العنصر	الترتيب
	الوضوح والدقة	1
	السياق والمعلومات المساعدة	2
	تحديد الدور	3
	تنسيق المخرجات	4
	مثال توضيحي	5
	التكرار والتحسين	6
	التفكير خطوة بخطوة	7

## النتيجة النهائية:

---

---

---

---

---

---

---

---

---

---

## في وقت فراغك



اكتب Prompt يوجّه محرك الذكاء الاصطناعي لتصميم نظام زراعة ذكي يقلل استخدام المياه ويزيد إنتاج الغذاء في بيئة الكويت عام 2035 م للوصول للاكتفاء الذاتي الزراعي.

السؤال الموجه	العنصر	الترتيب
	الوضوح والدقة	1
	السياق والمعلومات المساعدة	2
	تحديد الدور	3
	تنسيق المخرجات	4
	مثال توضيحي	5
	التكرار والتحسين	6
	التفكير خطوة بخطوة	7

النتيجة النهائية:

---

---

---

---

---

---

---

---



## عبر عن رأيك



أستخدم هندسة الأوامر في تحسين مخرجات الذكاء الاصطناعي.

أستطيع توضيح فوائد هندسة الأوامر في الحصول على نتائج أدق وأفضل.

أحلل مكونات الأمر الجيد مثل السياق والمهمة والمخرجات.

أطبق عناصر الصياغة الجيدة عند كتابة Prompt فعّال.

أتمكن من تحسين صياغة أوامري بعد مراجعة النتائج.

أقيم جودة أي أمر من خلال مدى وضوحه وارتباطه بالنتائج المطلوبة.

## ملاحظات المعلم



الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

## التحقق من المعلومات ومصداقيتها في عصر الذكاء الاصطناعي

### نواتج التعلم

- توضيح مفهوم مصداقية المعلومات على الإنترنت.
- التمييز بين نتائج موثوقة وغير موثوقة من محركات البحث.
- استخدام معايير للتحقق من صحة المعلومات في نتائج الذكاء الاصطناعي.
- تحليل مخاطر المعلومات المضللة والأخطاء التوليدية.
- التحقق من مصدر البيانات عند استخدام نتائج البحث الذكي.
- اتخاذ قرارات واعية عند اعتماد معلومة في بحث أو مشروع.



ملفات أوراق العمل  
ملفات مصادر التعلم



### مقولة الدرس:

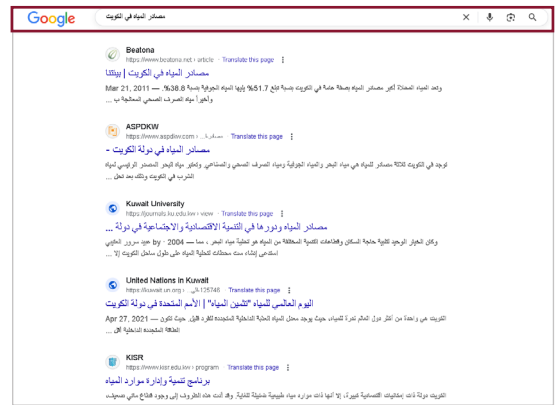
« الذكاء الاصطناعي جيد في توليد الأجوبة؛ لكنه ما زال بحاجة إلى أن تسأله الأسئلة الصحيحة »



# الاستكشاف



## تأمل الصورتين و أجب عن الأسئلة التالية:




أي الصورتين تشعرك بالاطمئنان أكثر؟ ولماذا؟

كيف يمكنك التأكد أن المعلومة صحيحة؟

## التعلم



### مصادقية المعلومات؟

هي قدرة المتلقي على الثقة بأن ما يصله حقيقي، دقيق، ويمكن التحقق منه، ويأتي من بيئة معلوماتية سليمة تحدد من التضليل والأخطاء. 

### لماذا التحقق مهم اليوم أكثر من قبل ؟

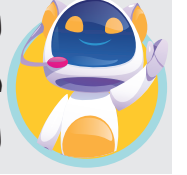
- 1** ازدياد ظاهرة المعلومات الخاطئة Information Disorder. انتشار محتوى ينتج بنية التضليل والتلاعب يجعل المتلقي أكثر عرضة للخداع ما لم يمارس التحقق الدقيق من صحة المحتوى.
- 2** تضخم دور محركات البحث والثقة المفرطة بنتائجها. المستخدمون كثيراً ما يثقون بنتائج البحث دون التحقق منها بعناية، مما يؤدي إلى تداول معلومات غير صحيحة تقدم على أنها موثوقة.
- 3** تزايد المحتوى غير الدقيق الناتج عن الذكاء الاصطناعي. نماذج اللغة الكبيرة قد تولد معلومات تبدو صحيحة لكنها غير حقيقية، مما يجعل التحقق ضرورة مضاعفة عند استخدام محتوى مولد آلياً.
- 4** اتساع فضاء المعلومات وتعدد مصادرها بلا رقابة كافية. نشر المعلومات لم يعد مقتصرًا على جهات مهنية، مما يزيد من احتمالية وقوع الأخطاء ويستدعي التحقق من موثوقية المنصات والمصادر.
- 5** أهمية الوعي المعلوماتي في تحديد ما يعد حقيقة. مع التطور الرقمي، أصبح تقييم صحة المعلومات مرتبطاً بفهم أعمق لطبيعتها وحدود المعرفة، وفق منظور الوعي المعلوماتي.

6

تقدم تكنولوجيا الذكاء الاصطناعي وتغلغلها في الحياة العامة.

توسع قدرات أنظمة الذكاء الاصطناعي في إنشاء وتوصيل المعرفة يجعل التحقق عنصراً أساسياً في الاستفادة الآمنة من هذه التقنيات.

التحقق اليوم ليس مجرد مهارة إضافية، بل ركيزة لحماية الفهم الصحيح في عصر تتدفق فيه المعلومات بسرعة ومن مصادر متعددة، ويختلط فيه الحقيقي بالمضلل والآلي بالبشري.



## كيف نتحقق من المعلومة؟

1

الاعتماد على مصادر موثوقة والتحقق من أصل المعلومة.

التأكد من هوية الجهة التي قدمت المعلومة ومدى موثوقيتها، خصوصاً عند البحث عبر الإنترنت.

2

مقارنة المعلومات مع مصادر أخرى مستقلة.

التوافق بين أكثر من مصدر يعزز الثقة ويقلل من الوقوع في التضليل أو الخداع المعلوماتي.

3

التأكد من الحداثة الزمنية.

المعلومات القديمة قد تصبح غير صحيحة في السياقات الحالية، خاصة في بيئة الإنترنت السريعة.

4

التحقق من الأخطاء الناتجة عن الذكاء الاصطناعي.

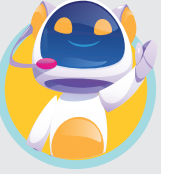
ضرورة فحص المعلومات المولدة آلياً وعدم قبولها دون تحقق، لأنها قد تبدو صحيحة وهي غير صحيحة.

5

تحليل نوايا الناشر والغرض من المحتوى.

من طرق مواجهة « خلل المعلومات » تحليل أسباب نشرها، وهل الدافع وراءها معرفي أم تجاري أم يهدف إلى التضليل؟

نحن نتحقق من المعلومة عبر التحقق من المصدر، المقارنة، التدقيق في دقة وحداثة المعلومة، وعدم الثقة العمياء بالمحتوى الرقمي وخاصة الذكاء الاصطناعي.



## الأخطاء الشائعة لمحركات البحث المعتمدة على الذكاء الاصطناعي

1. إنتاج محتوى غير مدعوم ببيانات صحيحة من قبل نماذج اللغة، يظهر بصياغة موثوقة ظاهرياً لكنه يخالف الوقائع ويعتمد على استنتاجات أو توليد لغوي غير مبني على معرفة حقيقية.
2. يمكن أن تقدم النماذج ملخصات أو نتائج غير دقيقة عند التعامل مع نصوص معقدة.
3. قدرات الذكاء الاصطناعي تتطور بسرعة، لكن رؤيته للعالم ليست خالية من الأخطاء مما يتطلب رقابة بشرية مستمرة.
4. تُقدم الإجابات بثقة عالية حتى عند غياب الأدلة أو عند توليد الإجابة، ما قد يضلّل المستخدم.
5. غالباً لا يستطيع النظام تحديد ما إذا كانت البيانات التي ينتجها صحيحة أم لا دون تدخل بشري أو مصادر موثوقة.



## النشاط



ابحث عن «مصادر الطاقة في دولة الكويت» وقارن بين مصدرين للمعلومة نفسها، استخدم محرك بحث تقليدي Google وآخر معتمد على الذكاء الاصطناعي ChatGPT / Perplexity / Copilot وأجب عن التالي:

نتيجة محرك البحث المعتمد على الذكاء الاصطناعي	نعم/لا	نتيجة محرك البحث التقليدي	نعم/لا	السؤال
النتيجة تولّدها الآلة دون ذكر المؤلف الا عندما يتم طلب ذكر المؤلف مع النتيجة	نعم ولا	الجهة المنتجة للمعلومة IEA وهي منظمة دولية موثوقة	نعم	هل للمعلومة مؤلف موثوق؟
لا توجد عادة روابط مباشرة لمصادر رسمية أو مراجع واضحة، أو قد تخلق روابط وهمية	لا	مصدر رسمي / حكومي معتمد في مجال الطاقة	نعم	هل وردت من مصدر معتمد؟
قد تتضمن ذكر تاريخ لكن ليس دائماً، وقد تكون المعلومات مختلطة بين تحديثات حديثة وقديمة	نعم	بعض البيانات حديثة (2024 - 2023)	نعم	هل هي حديثة ومناسبة للوقت؟
الهدف تزويد معلومات لكن قد يفتقر لتوضيح مستوى الدقة المطلوب	نعم	الهدف توعوي - تحليلي حول الطاقة في دولة الكويت	نعم	هل الهدف منها واضح؟
غالباً يحتاج المستخدم إلى بذل جهد إضافي للتحقق والاستقصاء لأن المصدر غير موثق بوضوح	لا	MEW - Our World in Data تؤكد الفكرة نفسها	نعم	هل يمكن التأكد منها من مصدرين آخرين؟

■ المحرك الذي قَدّم معلومات أكثر موثوقية هو محرك البحث التقليدي، لأنّ نتائجه كانت مرتبطة بمصادر رسمية مثل مواقع حكومية ومنظمات دولية للطاقة، وتحتوي على روابط واضحة وتواريخ نشر يمكن التحقق منها، بينما محرك البحث المعتمد على الذكاء الاصطناعي أعطى إجابات سريعة لكنها دون مراجع، وقد تتضمن معلومات غير دقيقة أو مختلطة.



■ الحصول على الإجابة بسرعة لا يعني بالضرورة أنها صحيحة أو يمكن الوثوق بها؛ لذلك من المهم التحقق من المعلومة، حتى وإن بدت الإجابة مقنعة وسهلة.

## التطبيق



ورقة عمل 5 : البحث عن الحقيقة

اختر معلومة من القائمة التالية ثم اختبرها :

البيئة والحياة البرية في الخليج

تطبيقات المدن الذكية

الذكاء الاصطناعي في التعليم

اكتب معلومة وجدتها على الإنترنت للموضوع الذي اخترته :

الآن اختبرها باستخدام الأسئلة :

ملاحظة	لا	نعم	السؤال
	<input type="checkbox"/>	<input type="checkbox"/>	هل للمعلومة مؤلف موثوق؟
	<input type="checkbox"/>	<input type="checkbox"/>	هل وردت من مصدر معتمد؟
	<input type="checkbox"/>	<input type="checkbox"/>	هل هي حديثة ومناسبة للوقت؟
	<input type="checkbox"/>	<input type="checkbox"/>	هل الهدف منها واضح؟
	<input type="checkbox"/>	<input type="checkbox"/>	هل يمكن التأكد منها من مصدرين آخرين؟



## في وقت فراغك



قارن بين مصدرين للمعلومة نفسها، استخدم محرك بحث تقليدي Google وآخر معتمد على الذكاء الاصطناعي ChatGPT / Perplexity / Copilot للبحث عن «مصادر الطاقة في الكويت» و أجب عن التالي:

نتيجة البحث الذكي	نتيجة البحث التقليدي	السؤال
		هل ذكر المؤلف أو الجهة ؟
		هل يوجد مصدر موثق ؟
		حدائثة المعلومة (قديمة / حديثة)
		دقة الجواب (عالية / متوسطة / ضعيفة)
		هل احتوى على خطأ ؟ (نعم/لا ) + مثال

### النتيجة النهائية:

المحرك الأكثر موثوقية بالنسبة لي هو :

السبب :



## عبر عن رأيك



أتحقق من صحة المعلومات على الإنترنت.

أميز بين مصدر موثوق وآخر غير موثوق.

أفحص نتائج الذكاء الاصطناعي وأتجنب أخطاءه.

أخذ قراراً واعياً قبل اعتماد المعلومة.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

# الحماية الرقمية ومسؤولية استخدام محركات البحث الذكي

## نواتج التعلم

- التعرف على الفرق بين الخصوصية الرقمية والأمن الرقمي.
- استنتاج مخاطر مشاركة البيانات الشخصية عبر الإنترنت.
- التمييز بين الممارسات الآمنة وغير آمنة عند استخدام محركات البحث والذكاء الاصطناعي.
- تطبيق طرق لحماية الخصوصية أثناء التصفح والتفاعل الرقمي.
- تحليل مسؤولية المستخدم الرقمي في حماية بياناته.
- التعرف على دور الذكاء الاصطناعي في الأمن السيبراني.
- استكشاف تطبيقات الذكاء الاصطناعي في كشف الهجمات وحماية الأنظمة.



ملفات أوراق العمل  
ملفات مصادر التعلم



## مقولة الدرس:

”مستقبلنا يعتمد على ما نحمله بقدر ما  
يعتمد على ما نبتكره”



## الاستكشاف



■ كيف يمكن أن تعرف إن كان الموقع الذي تزوره يحمي خصوصيتك؟

■ ما البيانات التي يمكن أن تُجمع عنك دون علمك؟

■ من يستطيع رؤية تلك البيانات التي تجمع عنك؟ كيف يمكن استغلال تلك البيانات؟

■ كيف يمكنك أن تحمي نفسك؟



## التعلم



### الخصوصية الرقمية (Digital Privacy)

هي قدرة الفرد على التحكم في بياناته الشخصية، من حيث نوع المعلومات التي يقدمها، والجهات التي يسمح لها بالاطلاع عليها، وكيف يمكن أن توظف تلك البيانات.



### الأمن الرقمي (Digital Security)

هو منظومة من الإجراءات والقواعد الشاملة المصممة لحماية الأصول الرقمية والبنى التقنية (أجهزة، شبكات، بيانات) من الاختراق، التلاعب، أو الإتلاف.



## مخاطر الذكاء الاصطناعي ومحركات البحث AI & Search Engine Risks

1. انتشار مشاركة البيانات في التطبيقات ومواقع البحث دون أن نلاحظ ذلك، وهذا يزيد من مخاطر فقدان الخصوصية.
2. كثرة الهجمات الإلكترونية وسرقة الحسابات مما يجعل الأمن الرقمي ضرورة لحماية المستخدم.
3. تطور الذكاء الاصطناعي الذي يمكنه تعقب وتحليل سلوك المستخدم والذي يمكن استغلاله بشكل مضر بيد جهات غير مسؤولة.
4. المعلومات الخاطئة أو الاستغلال التجاري للبيانات، وهذا يثير قضايا أخلاقية تتعلق بمن يتحكم ببياناتنا وكيف تُستخدم.
5. تحييز في الخوارزميات Algorithm Bias يؤدي إلى نتائج غير عادلة أو مضللة.

## مسؤولية المستخدم الرقمي عند استخدام محركات البحث والذكاء الاصطناعي Digital User Responsibility

1. عدم مشاركة البيانات الشخصية.
2. التأكد من موثوقية الإجابة قبل الاعتماد عليها.
3. قراءة التحذيرات وسياسة الخصوصية.
4. عدم فتح الروابط المشبوهة.
5. التفكير قبل نشر أو إرسال أي معلومة.
6. استخدام برامج مكافحة الفيروسات وجدران الحماية.
7. تفعيل خاصية المصادقة الثنائية للحسابات الرقمية.

نحن مسؤولون عن أماننا الرقمي، ومحركات البحث الذكية تساعدنا إذا استخدمناها بوعي واحترام للبيانات والخصوصية.



### دور الذكاء الاصطناعي في تعزيز الأمن السيبراني

1. تحسين اكتشاف التهديدات والتحليلات السلوكية.
2. الاستجابة التلقائية للحوادث وتقليل وقت التفاعل.
3. تعزيز كفاءة أدوات الأمن وتقليل الأخطاء الكاذبة.
4. توسيع نطاق الأمان ليشمل بنية تحتية أكبر (الشبكات، إنترنت الأشياء، والبيانات الكبيرة).

## تطبيقات الذكاء الاصطناعي في الدفاعات السيبرانية

### 1 الاكتشاف والتنبؤ بالانتهاكات والتهديدات

#### Intrusion & Threat Detection & Prediction

يستخدم الذكاء الاصطناعي خوارزميات تعلم آلي وتحليل سلوك لاكتشاف أنماط غير طبيعية أو نشاط مشبوه في الشبكة أو النظام، ما يساعد المؤسسات من الاستجابة المبكرة للتهديدات.

### 2 المراقبة المستمرة وتحليل حركة الشبكة

#### Continuous Monitoring & Network Traffic Analysis

تستخدم أدوات الذكاء الاصطناعي لمتابعة كميات كبيرة من البيانات والأنشطة لحظة بلحظة، مثل حركة الشبكة أو الأجهزة المتصلة بالإنترنت، وذلك للكشف عن أي أنشطة مريبة أو خبيثة.

### 3 استخدام الذكاء الاصطناعي في تحليل البرمجيات الخبيثة وتصنيفها

#### Malware Analysis & Classification

قدرة تقنيات التعلم العميق من تحليل الشيفرة أو سلوك البرمجيات الخبيثة وتحديدتها بشكل أسرع مما يمكن أن يفعله الإنسان وحده، وأحياناً في بيئات مشفرة أو متغيرة.

### 4 الاستجابة التلقائية للحوادث وتقليل زمن التفاعل

#### Automated Incident Response & Reduced Response Time

باستخدام الذكاء الاصطناعي، يمكن أن تتخذ قرارات دفاعية بسرعة مثل: ( حظر عنوان IP، عزل جهاز، إشعار تحذير) دون تأخير، مما يقلل من الأضرار المحتملة.

### 5 الاستخبارات التهديدية القائمة على الذكاء الاصطناعي

#### AI-Driven Threat Intelligence

يستخدم الذكاء الاصطناعي لجمع وتحليل كمّ هائل من المعلومات من مصادر متعددة (مثل السجلات، وسائل التواصل الاجتماعي، <sup>1</sup>Dark Web) لاستنتاج أنماط التهديد المستقبلية، وتحليل خصائص المهاجمين، وتوقع التحركات التالية.

<sup>1</sup> Dark Web : هو جزء من الإنترنت لا يمكن الوصول إليه بمحركات البحث العادية، ويحتاج إلى برامج خاصة للدخول إليه يُستخدم أحياناً لأغراض سرية أو غير قانونية.

## التطبيق



### ورقة عمل 7 :

أكمل الجدول التالي وفقاً لما تعلمته:

المفهوم	التعريف	مثال من واقع الإنترنت
الخصوصية الرقمية		
الأمن الرقمي		
أحد مخاطر الذكاء الاصطناعي		

ضع علامة ( ✓ ) أمام العبارة الصحيحة:

خطأ	صح	العبارة
<input type="checkbox"/>	<input type="checkbox"/>	عدم قراءة سياسة الخصوصية سلوك آمن.
<input type="checkbox"/>	<input type="checkbox"/>	الذكاء الاصطناعي قد يعطي معلومات خاطئة أو غير موثوقة.
<input type="checkbox"/>	<input type="checkbox"/>	من الأفضل تفعيل المصادقة الثنائية لحماية الحسابات.
<input type="checkbox"/>	<input type="checkbox"/>	يمكن مشاركة الرقم المدني في أي موقع لأن الجميع يفعلوه.
<input type="checkbox"/>	<input type="checkbox"/>	بعض محركات الذكاء الاصطناعي قد تكون متحيزة للبيانات التي درّبت عليها.

## ورقة عمل 8 :

اقرأ الجمل التالية وضعها في العمود المناسب:

- طلب ذكر المصادر
- مشاركة كلمة المرور
- التحقق قبل فتح الروابط
- تفعيل خاصية المصادقة الثنائية
- قبول الإجابة دون تفكير
- عدم مشاركة البيانات الشخصية

سلوك غير آمن	سلوك آمن

اطلب من محرك ذكاء اصطناعي معلومات حول «حماية البيئة في دولة الكويت» دون ذكر أي بيانات شخصية، مع طلب مصادر موثوقة، اكتب أمرك هنا؟

---

---

---

---

---

---

---

---

---

---

## في وقت فراغك



اقرأ الحوار التالي بين مستخدم ومحرك ذكاء اصطناعي، ثم حدّد الأخطاء والسلوك الصحيح:

حمد: مرحباً، هذا رقم هاتفي: 60012345...  
أحتاج مساعدة لاستعادة حسابي في منصة المدرسة.

محرك الذكاء الاصطناعي: بالتأكيد! اضغط على الرابط التالي:  
[www.verify-login-school-info-87.net](http://www.verify-login-school-info-87.net)

حمد: الرابط يطلب رفع صورة البطاقة المدنية... هل أرسلها لهم؟

محرك الذكاء الاصطناعي: نعم! كلما زوّدتهم ببيانات أكثر كانت الخدمة أسرع.

رقم	السلوك	صح	خطأ	لماذا	السلوك الصحيح
1	مشاركة رقم الهاتف الشخصي	<input type="checkbox"/>	<input type="checkbox"/>		
2	الضغط على رابط مجهول	<input type="checkbox"/>	<input type="checkbox"/>		
3	إرسال البطاقة المدنية للموقع	<input type="checkbox"/>	<input type="checkbox"/>		
4	الثقة الكاملة بالذكاء الاصطناعي	<input type="checkbox"/>	<input type="checkbox"/>		



## عبر عن رأيك



أتعرف على الفرق بين الخصوصية الرقمية والأمن الرقمي.

أتعرف على دور الذكاء الاصطناعي في تعزيز الأمن السيبراني.

لا أشارك بياناتي الشخصية مع محركات البحث.

أتجنب الروابط غير الواضحة أو المشبوهة.

أتحقق من مصادر المعلومات قبل الاعتماد عليها.

أساهم في نشر الوعي الرقمي بين أصدقائي وزملائي.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

# برمجة وتطوير الألعاب Game Design and Development

برنامج Kodu Game Lab وصناعة الألعاب	1
برنامج Kodu Game Lab	2
إضافة التضاريس	3
الماء والمسارات وإعدادات العالم	4
الخطوة الأولى نحو البرمجة	5
إنشاء وتتبع المسار	6
برمجة الفوز والخسارة	7
برمجة الأهداف والتتبع الذكي	8
الحوارات التفاعلية ورفع مستوى اللعبة	9
منظور الكاميرا والمؤثرات الصوتية	10
خصائص متقدمة في اللعبة	11



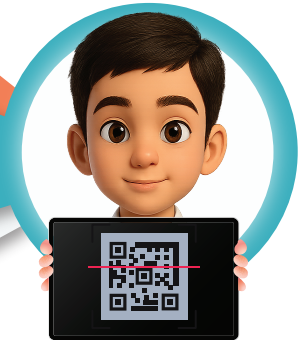
# برنامج Kodu Game Lab وصناعة الألعاب

### نواتج التعلم

- التعرف على مفهوم تطوير الألعاب وأهميته التعليمية والاقتصادية.
- تصنيف أنواع الألعاب وفقاً للمنصة المستخدمة وطريقة اللعب.
- تحديد المكونات الأساسية للعبة الالكترونية.
- استكشاف واجهة برنامج Kodu Game Lab وأدواتها الرئيسية.
- إنشاء عالم داخل البرنامج وتجهيزه للعب.
- إضافة وتحرير الكائنات.
- تجربة أدوات الكاميرا وضبط زوايا العرض وحركة المشهد.
- حفظ وتحميل المشاريع على جهاز الحاسوب الخاص.
- تصدير المشاريع على جهاز الحاسوب استعداداً لمشاركتها.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



نعم يا سارة باستخدام برنامج Kodu  
سنتعلم تصميم ألعابنا الإلكترونية  
وكأننا مطوّرو ألعاب حقيقيون!

سمعت يا حمد أننا سنتعلم كيفية  
تصميم ألعابنا الإلكترونية بأنفسنا!

■ اذكر اسم تطبيق لعبة المفضلة لديك؟

■ ما القواعد الأساسية التي تحكم هذه اللعبة؟

■ ما الأدوات أو الشخصيات التي تظهر في اللعبة؟

■ ما الذي يجعل هذه اللعبة ممتعة بالنسبة لك؟

## التعلم



### مفهوم اللعبة (Game Concept)

اللعبة هي عالم خيالي تفاعلي له قواعد وأهداف محددة، يؤدي فيه اللاعب أدواراً ويتخذ قرارات تؤثر في مسارها ونتيجتها، بهدف تحقيق غاية مثل الفوز أو التعلم وتنمية المهارات.



### مفهوم تطوير اللعبة (Game Development)

هو عملية تصميم وبناء لعبة تفاعلية بدءاً من توليد الفكرة وصولاً إلى الإصدار النهائي، وتشمل البرمجة، الرسم، القصة والتأثيرات الصوتية لجعلها ممتعة وجاهزة للنشر.

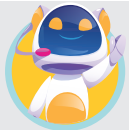


## أهمية تطوير الألعاب The Importance of Game Development

تطوير الألعاب ينمي مهارات متعددة لدى المتعلم، منها:

1. حل المشكلات عند مواجهة أخطاء أثناء البرمجة والتشغيل.
2. التفكير المنطقي من خلال كتابة أوامر واضحة ومنظمة للشخصيات.
3. التعاون في تصميم مشاريع جماعية.
4. الإبداع في رسم العوالم وتخيل القصص.
5. التعبير الذاتي بتحويل الأفكار إلى تجربة مرئية ممتعة.

تنفيذ لعبة تجمع بين البرمجة، التصميم، الفن، الصوت، وواجهات المستخدم، مما يجعلها مشروعاً متعدد التخصصات؛ فهي ليست مجرد ترفيه، بل وسيلة تعليمية قوية تساعدنا على التعلم من خلال اللعبة.



## تصنيفات الألعاب Game Genres

تتنوع الألعاب حسب الهدف وطريقة اللعب من بينها:

### 1 ألعاب المغامرة Adventure Games

يخوض اللاعب مغامرات مليئة بالتحديات يستكشف العوالم، يجمع الأدوات، ويتبع قصة مليئة بالأحداث لحل الألغاز والتقدم في الرحلة.

### 2 ألعاب لعب الأدوار Role-Playing Game, RPG

يؤدي فيها اللاعب شخصية داخل قصة، يطور قدراتها ويختار القرارات التي تؤثر على مسار الأحداث.

### 3 ألعاب الأحجية Puzzle Games

اللاعب يفكر ويحل المسائل أو الألغاز للوصول إلى الحل الصحيح والتقدم إلى المرحلة التالية.

### 4 ألعاب المنصة Platform Games

اللاعب يمر بمستويات وعوائق مختلفة، غالباً تتصف بالقفز والمناورة لإتمام المهمة بسرعة.

### 5 ألعاب الاستراتيجية Strategy Games

اللاعب يخطط ويدير الفرق والموارد أثناء سير اللعبة دون توقف، ويتخذ قرارات سريعة للفوز.

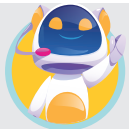
### 6 ألعاب الرياضة Sports Games

اللاعب يتحكم بشخصية رياضية أو فريق كامل بهدف المنافسة في مسابقات رياضية متنوعة مثل كرة القدم أو السلة أو التنس.

### 7 ألعاب السباق Racing Games

اللاعب يقود سيارة أو مركبة ويحاول الوصول إلى خط النهاية أو تنفيذ مهمات القيادة بدقة وسرعة.

هناك العديد من التصنيفات للألعاب الإلكترونية وقد تجمع بعض الألعاب بين أكثر من تصنيف في آن واحد.



## مكونات اللعبة الأساسية Basic Game Components

المكون	الوصف
الفكرة والمفهوم	ما الهدف من اللعبة وما التجربة التي ستمنحها للاعب؟
الشخصيات الرئيسية	الكائنات التي تتحرك أو تتفاعل داخل اللعبة.
العالم / بيئة اللعبة	المكان الذي تجري فيه أحداث اللعبة (بحر، صحراء، سماء...).
آليات اللعب	القواعد، الإجراءات، الخيارات التي يملكها اللاعب.
التحكم	طريقة تفاعل اللاعب مع اللعبة (الأسهم، الماوس، عصا التحكم، أو لوحة المفاتيح).
الفن و الرسوميات	تصميم الشخصيات، الألوان، الإضاءة، كل ما يراه ويسمعه اللاعب لخلق جو مناسب.
المنطق البرمجي	منطق الأوامر الذي يحدد متى وكيف يحدث كل شيء داخل اللعبة وكيف يتفاعل اللاعب مع اللعبة .
الاختبار	تجربة اللعبة وتحسينها قبل نشرها.

## التعرف على برنامج Kodu Game Lab

هي بيئة برمجية مرئية ثلاثية الأبعاد طورتها شركة Microsoft لتعليم مبادئ البرمجة وتصميم الألعاب بطريقة سهلة، حيث يمكن إنشاء عوالم وشخصيات ثلاثية الأبعاد.



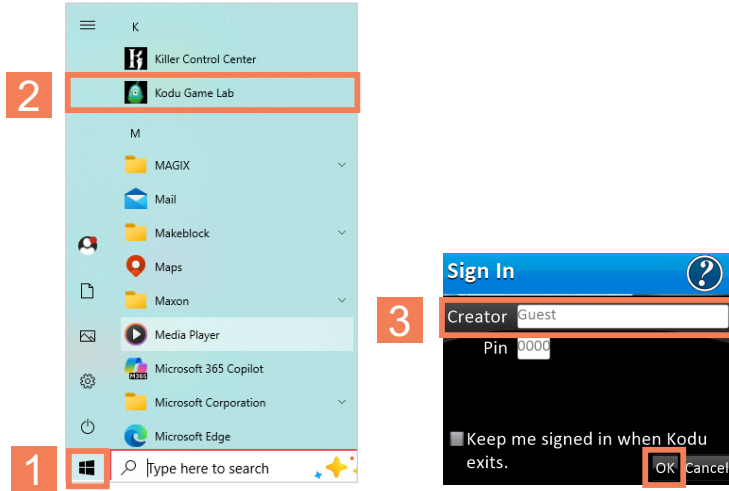
## مميزات برنامج Kodu Game Lab

1. برمجة بصرية سهلة باستخدام أوامر جاهزة على شكل tiles مما يسهل تعلم البرمجة للمبتدئين.
2. بيئة ثلاثية الأبعاد مما يعزز الإبداع ويقرب التجربة من تصميم ألعاب حقيقية.
3. تحكم متعدد (لوحة مفاتيح، فأرة، أو وحدة تحكم) مما يجعل الوصول للبرنامج أكثر تنوعاً وسهولة في بيئة الصف.

## خطواتي الأولى في برنامج Kodu Game Lab

### ■ أولاً: تشغيل البرنامج

1. اضغط على زر Start .
2. اختيار برنامج Kodu Game Lab من قائمة البرامج.
3. في صندوق حوار تسجيل الدخول Sign in يتم تسجيل اسم المستخدم ثم ضغط Ok.



### ■ ثانياً: بناء عالمك الأول

ستنشئ عالمك الأول في برنامج Kodu Game Lab، وهو المكان الذي ستبدأ فيه مغامرتك لبناء لعبتك الخاصة وإطلاق خيالك بلا حدود!

# التعرف على Kodu وصناعة الألعاب

لعمل ذلك اتبع التعليمات التالية:

■ من القائمة الرئيسية Main Menu اختر الأمر .NEW WORLD

■ تظهر لك خيارات لإنشاء عالم جديد من تصميمك أو البدء بعالم من مكتبة البرنامج.

■ سنبداً بأحد العوالم الجاهزة من مكتبة البرنامج.



■ **ثالثاً:** استكشاف واجهة البرنامج Kodu Interface

تتكون شاشة البرنامج من عدة مناطق أساسية، تساعدك على التحكم في العالم وبناء لعبتك بسهولة، وهي:

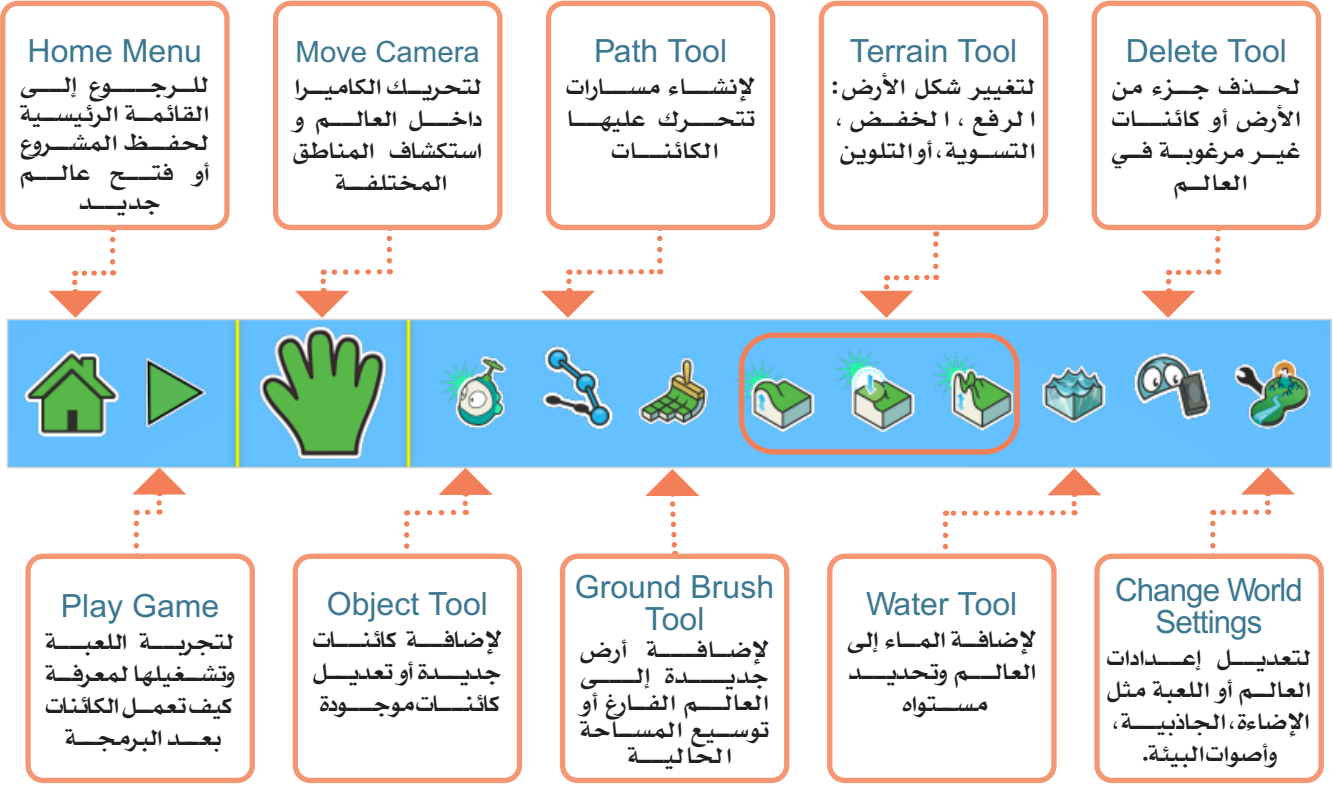
## منطقة العالم World View:

هي المساحة ثلاثية الأبعاد التي تنشئ فيها اللعبة وتضع الكائنات وتعديل البيئة.



## شريط الأدوات Tool Bar:

مكون من مجموعة من الأزرار التي تسمح لك بالتحكم في العالم وتعديل اللعبة بسهولة.



## ■ رابعاً: إضافة وتحرير الكائنات

يمكنك بناء محتوى لعبتك عن طريق اختيار الكائنات ووضعها في أماكن معينة، ثم تغيير شكلها أو لونها أو برمجتها لتتصرف بالطريقة التي تريدها.



## لإضافة كائن اتبع الخطوات التالية:

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط في أي مكان في العالم بالزر الأيسر للفأرة.
- ظهور Object Wheel.
- اختيار الكائن المطلوب اضافته.

يمكن التراجع عن اظهار Object Wheel من خلال الضغط على المفتاح Esc.



لتحرير كائن اتبع الخطوات التالية:

■ اختيار أداة Object Tool من شريط الأدوات.

■ الضغط على الكائن بالزر الأيسر للفأرة لتحريك الكائن.

■ من خلال السحب والإفلات.

■ الضغط على الكائن المطلوب التعديل على خصائصه بالضغط بالزر الأيمن للفأرة للفأرة ثم:

■ يظهر لوح الألوان Color Palette يمكن اختيار اللون من خلال السهمين (الأيمن - الأيسر) في لوحة المفاتيح والتوقف عند اللون المطلوب.

■ تظهر قائمة منسدلة Object Edit Menu تحتوي على عدة خيارات، حدد الخيار المطلوب.



## الوظيفة

## خيارات تحرير الكائن

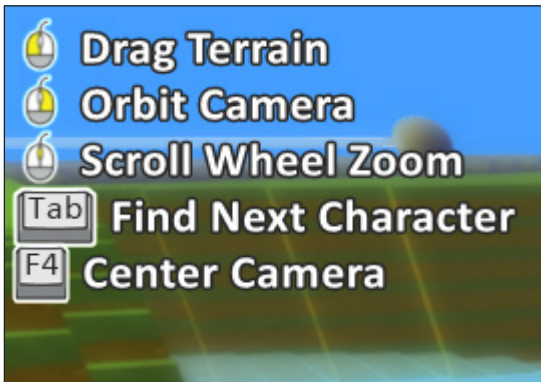
برمجة الكائن وتحديد سلوكه باستخدام أوامر (When ... Do) مثل الحركة أو جمع الأشياء.	Program
تغيير إعدادات الكائن مثل السرعة، نوع التحكم، الفريق، ونمط الحركة.	Change Settings
تغيير اسم الكائن ليسهل التعرف عليه.	Rename
قص الكائن من مكانه الحالي مع إمكانية لصقه في مكان آخر.	Cut
إنشاء نسخة مطابقة من الكائن ووضعها في مكان آخر داخل العالم.	Copy
تكبير أو تصغير الكائن حسب الحاجة في المشهد.	Change Size
تدوير الكائن وتغيير اتجاهه في العالم.	Rotate
رفع الكائن أو خفضه فوق سطح الأرض (مثلاً يطفو الكائن أو يجلس على التل).	Change Height

- بعد استخدام خيار القص أو النسخ يجب استخدام أزرار لوحة المفاتيح (ctrl + v) لعمل لصق للكائن.
- لحذف الكائن الضغط على مفتاح Delete.



## ■ خامساً: التحرك في العالم باستخدام الكاميرا

يمكنك استكشاف عالمك في Kodu من زوايا مختلفة، لتتمكن من رؤية كل تفاصيل اللعبة أثناء التصميم أو اللعب.



- تأكد من اختيار أداة Move Camera

من شريط الأدوات. 

خيارات قائمة المساعدة أعلى يسار الشاشة.

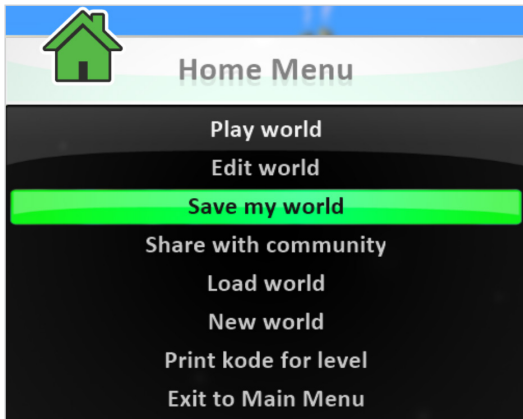


# التعرف على Kodu وصناعة الألعاب

الوظيفة	خيارات الكاميرا
الضغط على زر الفأرة الأيسر والسحب لتحريك المشهد في أي اتجاه داخل العالم.	Drag Terrain
الضغط على زر الفأرة الأيمن لتدوير الكاميرا حول الكائنات أو المشهد.	Orbit Camera
استخدام عجلة الفأرة للتقريب Zoom in أو الابتعاد Zoom out.	Scroll Wheel Zoom
الانتقال السريع إلى الكائن التالي في العالم.	Tab - Find Next Character
يعيد الكاميرا إلى مركز العالم أو إلى موضع الكائن الأساسي.	F4 - Center Camera

## سادساً: تخزين واسترجاع الألعاب

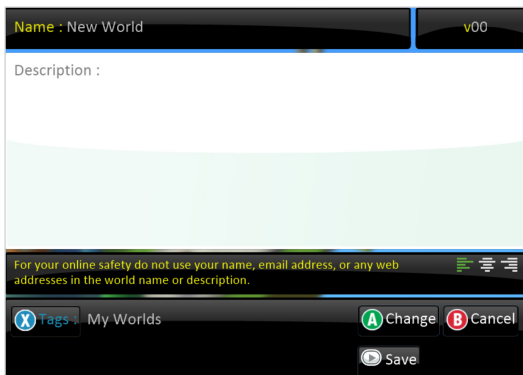
بعد أن تصمم عالمك في برنامج Kodu، من المهم أن تعرف كيف تحفظ عملك، وتحمله لاحقاً لمتابعة التعديل، وتصدره لمشاركته مع الآخرين أو نقله إلى جهاز آخر. لحفظ اللعبة داخل البرنامج على جهازك اتبع الخطوات التالية:



1. انقر على أداة Home Menu ، من شريط الأدوات.

2. اختيار الامر Save my world.

3. ظهور نافذة World Save Dialog Box ، ووظيفتها تحديد معلومات الحفظ لعالمك الجديد.



وفيما يلي شرح لمكوناته ووظائفها :

الوظيفة	العنصر
لإدخال اسم العالم الذي تريد حفظه.	Name
لإضافة وصف بسيط لعالمك يوضح فكرته أو مكوناته (اختياري).	Description
لتصنيف العالم ضمن مجلد أو مجموعة.	Tags
لتأكيد التغييرات أو تعديل المعلومات قبل الحفظ.	A - Change
لإلغاء عملية الحفظ والعودة دون حفظ التعديلات.	B - Cancel
لتنفيذ عملية الحفظ الفعلية وتخزين العالم على الجهاز.	Save
تذكير بعدم استخدام اسمك أو بريدك الإلكتروني للحفاظ على الأمان أثناء العمل.	التنبيه الأصفر في الأسفل

استدعاء لعبة أو عالم سبق حفظه اتبع الخطوات التالية:



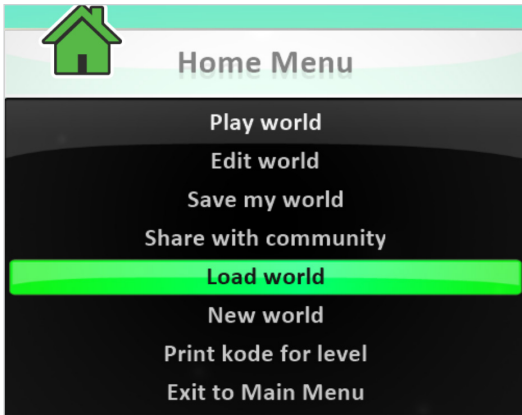
1. انقر على أداة Home Menu ،

من شريط الأدوات.

2. اختيار الامر Load World.

3. ظهور نافذة Worlds Library Screen ، ووظيفتها

هي عرض جميع الألعاب والعوالم المخزنة في البرنامج لتتمكن من فتحها أو تشغيلها.



# التعرف على Kodu وصناعة الألعاب

الوظيفة	العنصر
يعرض العوالم (الألعاب) التي حفظتها بنفسك.	My Worlds
يعرض العوالم التي تم تحميلها من الإنترنت أو من الآخرين.	Downloads
يحتوي على الدروس التعليمية أو العوالم التجريبية المقدمة من مطوري Kodu.	Lessons
يعرض جميع العوالم الموجودة في البرنامج في مكان واحد.	All
يعرض تفاصيل عن العالم المحدد مثل الاسم، وقت الإنشاء، والمصمم، ووصفه.	الوصف أسفل الشاشة
للخروج من شاشة التحميل والعودة إلى القائمة السابقة.	Esc

لتصدير اللعبة إلى ملف خارجي اتبع الخطوات التالية:



1. انقر على أداة Home Menu ،  
من شريط الأدوات.

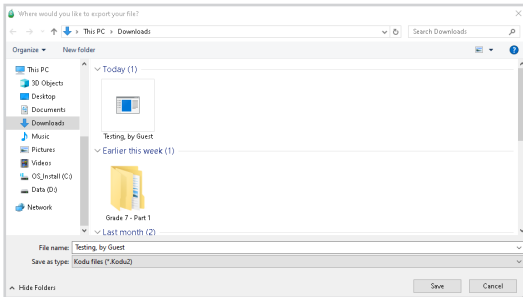
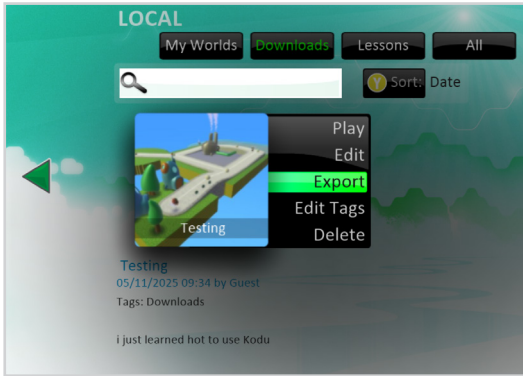
2. اختيار الامر Load World.

3. اضغط بزر الفأرة الأيسر على اللعبة المطلوب  
تصديرها.

4. اختيار الامر Export من القائمة المنسدلة.

5. ظهور صندوق حوار

، Where would you like to export your file  
تحديد المكان المطلوب للتصدير ثم اضغط على  
Save.



نوع الملف المصدر سيكون (\*Kodu2).



# النشاط



## New world

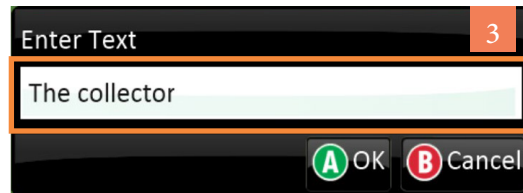


### 1. بدء المغامرة

- تشغيل برنامج Kodu Game Lab واختيار الأمر New World.
- من المكتبة الظاهرة أمامك، اختر أحد العوالم الجاهزة ليكون نقطة انطلاقك.

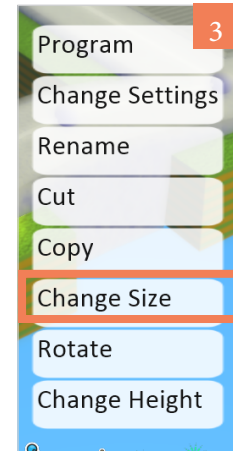
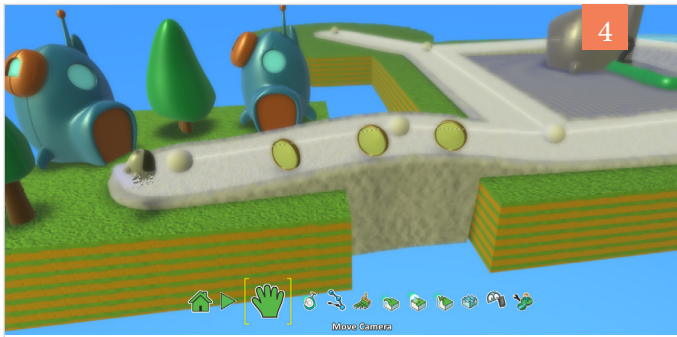
### 2. إضافة الكائنات إلى اللعبة

- إضافة شخصية Kodu إلى العالم، ثم تغيير اسمه إلى The collector.



# التعرف على Kodu وصناعة الألعاب

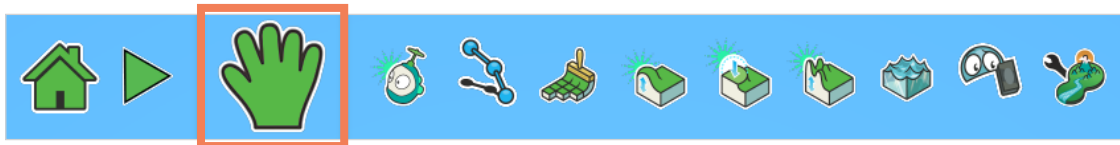
- إضافة كائن Coin مع تغيير لونه إلى لون لامع وجذاب، من خصائص الكائن يتم تغيير حجم العملة لتصبح واضحة، ثم نسخ العملة ثلاث مرات ووضع كل نسخة في مكان مختلف داخل العالم لتصبح اللعبة أكثر متعة واستكشافا.



## 3. اختبار العالم.

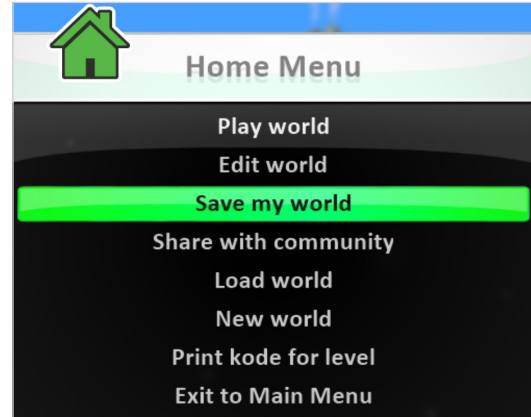
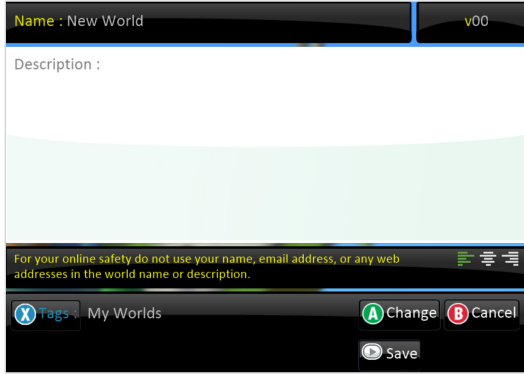
- استخدام الكاميرا لتحريك المشهد من زوايا مختلفة ومراقبة توزيع العناصر.
- التأكد من أن كل العملات واضحة ويمكن الوصول إليه.

أداة الكاميرا



#### 4. توثيق الانجاز

■ حفظ اللعبة باسم My First Game.



■ تصدير اللعبة إلى سطح المكتب باسم المتعلم الثلاثي لمشاركتها لاحقاً، باتباع الخطوات التالية:

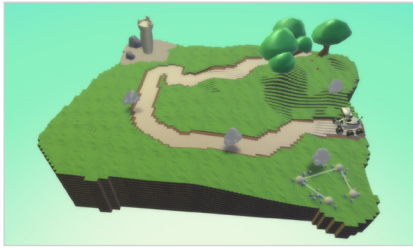
- الضغط على أداة Home Menu، من شريط الأدوات.
- اختيار الأمر Load world.
- الضغط بزر الفأرة الأيسر على اللعبة المطلوب تصديرها.
- اختيار الأمر Export من القائمة المنسدلة.
- تحديد المكان المطلوب للتصدير ثم الضغط على Save.

## التطبيق



### ورقة عمل 9 : مهمة الروبوت Rover في عالم الغيوم

صمم عالماً يقوم فيه الروبوت Rover باستكشاف البيئة لتحديد الأماكن التي يجب تنظيفها من سحب الحبر Ink Cloud المنتشرة في كل مكان.



#### 1. بناء العالم

- شغل برنامج Kodu Game Lab.
- من القائمة الرئيسية Main Menu اختر الامر NEW word.
- اختر عالماً جاهزاً من مكتبة البرنامج.

#### 2. إضافة الروبوت Rover

- أضف الروبوت Rover إلى العالم.
- ضع الروبوت في نقطة بداية مناسبة للمهمة.
- كبر حجم الروبوت وتأكد من أن موقعه واضح ويسهل الوصول إليه.

#### 3. توزيع سحب الحبر Ink Cloud

- أضف الكائن Cloud إلى العالم.
- غير لون الغيمة إلى اللون الأسود المشابه للحبر.
- انسخ الغيمة ووزعها في أماكن متعددة داخل البيئة.
- احرص على توزيع الغيوم بحيث يبدو المكان مليئاً بالغيوم الغامضة (قريبة - بعيدة - مخفية).

#### 4. التجول داخل العالم

- تجول داخل العالم للتحقق من : مواقع الغيوم، وضوح تصميم البيئة، واتجاهات الحركة داخل اللعبة.

#### 5. حفظ وتصدير اللعبة

- احفظ اللعبة باسمك متبوعاً بالكلمة EnviroRobot.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.

## ورقة عمل 10: مهمة السمكة Fish في عالم الكنوز



صمم عالماً مائياً مدهشاً تقوم فيه السمكة Fish باستكشاف أعماق المحيط للعثور على العملات الذهبية Coins المنتشرة بين الصخور والشعاب المرجانية.

### 1. بناء العالم

- شغل برنامج Kodu Game Lab.
- من القائمة الرئيسية Main Menu اختر الامر NEW world.
- اختر عالماً جاهزاً يعجبك من مكتبة البرنامج.

### 2. إضافة السمكة Fish

- أضف السمكة Fish إلى العالم.
- ضع السمكة في نقطة بداية مناسبة للمهمة.
- كبر حجم السمكة لتصبح واضحة وغير لونها إلى اللون الذهبي.

### 3. توزيع العملات Coins

- أضف الكائن Coin إلى العالم.
- كبر حجم العملة إلى 2.2 وغير لونها إلى اللون الأحمر.
- انسخ العملة ووزعها في أماكن متعددة داخل البيئة.
- احرص على توزيع العملات بحيث تبدو متناثرة في أعماق المحيط.

### 4. التجول داخل العالم

- تجوّل داخل العالم للتحقق من: مواقع العملات، انسجام العناصر، وتناسق تصميم البيئة.

### 5. حفظ وتصدير اللعبة

- احفظ اللعبة باسمك متبوعاً بالكلمة Treasure Hunt.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك

أستطيع التمييز بين أنواع الألعاب المختلفة.

أتعرف على مكونات اللعبة الإلكترونية وكيفية ارتباطها ببعضها.

أنشئ أول عالم خاص بي داخل برنامج Kodu بنجاح.

أستخدم الكاميرا لمشاهدة العالم من زوايا مختلفة.

أحفظ مشاريعي وأستدعيها لاحقاً عند الحاجة.

أصمم لعبة جديدة من فكرتي الخاصة.



## ملاحظات المعلم

الملاحظات:

التاريخ:

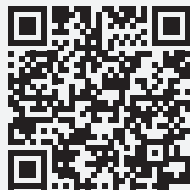
اليوم:

## ملاحظات ولي الأمر

# إضافة التضاريس

### نواتج التعلم

- التعرف على وظيفة أداة Ground Brush في بناء تضاريس عالم اللعبة.
- تمييز الفرق بين مواد التضاريس المختلفة وأشكال الفرش المتاحة.
- استخدام أداة Ground Brush لإنشاء تضاريس مسطحة وتوسيع مساحة اللعب.
- تغيير مادة التضاريس بدقة عبر مكتبة Terrain Material واختيار المادة المناسبة للبيئة.
- اختيار شكل الفرشاة المناسب وتوظيفه في تشكيل التضاريس بما يتوافق مع متطلبات تصميم العالم.
- استخدام الفرشاة لحذف أجزاء من التضاريس بدقة لتحسين شكل العالم وتنظيم مساراته.
- إنشاء مخطط أولي للعالم عبر توظيف مواد تضاريس متعددة وأشكال فرش متنوعة لرسم المسارات والحدود بدقة.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



Minecraft



Rocket League



Mario Kart

■ ما أكثر ما يجذبك في عالم اللعبة؟

■ هل تعتقد أن تصميم شكل العالم يؤثر على متعة اللعب؟

■ اكتب تصورك لعالم لعبتك الأول هنا: ( الشكل العام، العناصر الأساسية مثل التضاريس والشخصيات، وحدد الهدف الرئيسي للعبة).

## التعلم



عالم اللعبة هو الأساس الذي تُبنى عليه التجربة الرقمية، فهو يحدد البيئة التي يتحرك فيها اللاعب بما في ذلك، المسارات، الحدود، والعوامل التفاعلية. في حال عدم إنشاء بيئة متكاملة، تفقد اللعبة عمقها وتتحول إلى مساحة محدودة وغير مشجعة على الاستكشاف

## إنشاء تضاريس جديدة باستخدام أداة فرشاة الأرض Ground Brush Tool

سنتعلم طرقاً متعددة لتعديل تضاريس العالم الافتراضي، بما في ذلك إضافة الماء وغيرها من العناصر، ولكن قبل الانتقال إلى هذه الخيارات سنبدأ بخطوة بسيطة وهي توسيع المساحة التي يمكن للكائن التحرك داخلها.

1. البدء بإنشاء عالم جديد NEW WORLD.

2. تصغير المشهد بالاستعانة بعجلة الفأرة حتى تتمكن من رؤية المساحة المحيطة.

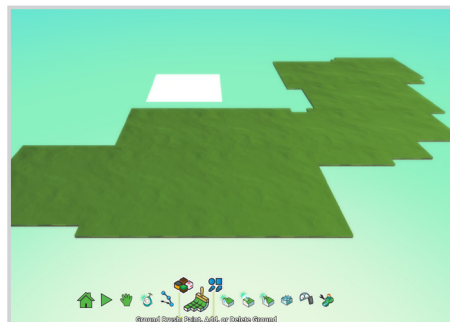


3. اختيار أداة فرشاة الأرض Ground Brush Tool من شريط الأدوات.

4. لاحظ ظهور أيقونتين صغيرتين فوقها (يميناً ويساراً).



5. عند اختيار أداة فرشاة الأرض حرّك مؤشر الفرشاة باستخدام الفأرة إلى الموقع المطلوب لإضافة قطعة جديدة ثم اضغط بالزر الأيسر للفأرة لإضافة التضاريس.

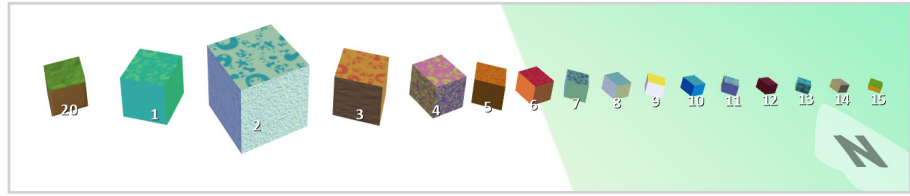


## التحكم في حجم الفرشاة

يمكنك تغيير حجم الفرشاة باستخدام السهمين الأيمن والأيسر من لوحة المفاتيح:

- الضغط على السهم الأيسر: تصغير حجم الفرشاة.
- الضغط على السهم الأيمن: تكبير حجم الفرشاة.

## تغيير مادة التضاريس Terrain Material

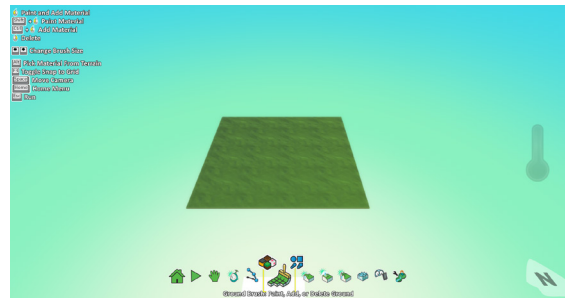
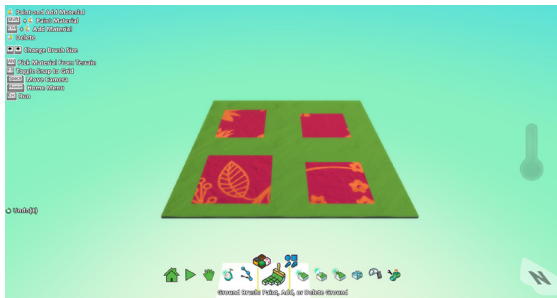


إذا لم تكن المادة الحالية تمنح التضاريس الشكل المطلوب، فيمكنك ببساطة استبدالها بمادة أخرى من مجموعة المواد المتاحة للحصول على مظهر أكثر دقة لطبيعة العالم الذي تصممه.

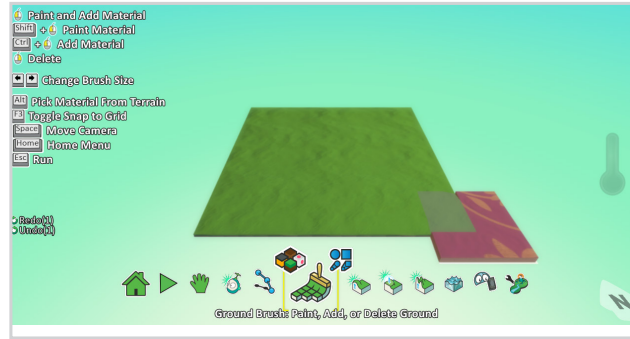


- اختيار أداة فرشاة الأرض من شريط الأدوات.
- ظهور أيقونتين صغيرتين فوقها (يميناً ويساراً).
- اختيار أيقونة Terrain Material الظاهرة في أعلى يسار أداة فرشاة الأرض.
- تظهر مكتبة تحتوي على مكعبات صغيرة تضم أكثر من 100 خيار لمواد مختلفة.

- استخدام عجلة الفأرة أو السهم الأيمن والأيسر من لوحة المفاتيح للتنقل بين الخيارات.
- الضغط على اللون أو المادة المفضلة لاعتمادها كمادة تستخدم في تشكيل التضاريس الجديدة.
- تحريك المؤشر إلى المنطقة المطلوبة، ثم الضغط بالزر الأيسر للفأرة لتطبيق المادة الجديدة.



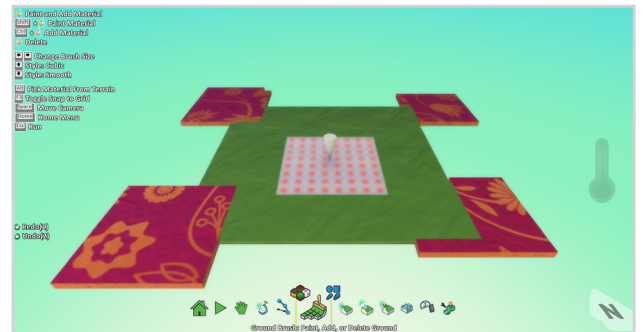
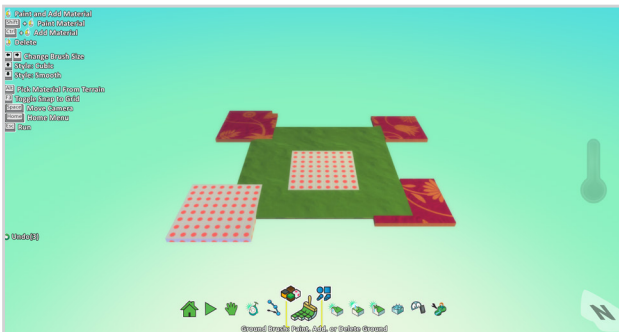
عند المرور بالفرشاة فوق تضاريس موجودة يتم استبدالها بالمادة الجديدة، لمنع التعديل على التضاريس الحالية اضغط على مفتاح Ctrl من لوحة المفاتيح أثناء السحب لتضاف التضاريس فقط في الأماكن الفارغة دون تغيير المادة القديمة.



- للحرص على الرسم داخل حدود تضاريس العالم الموجودة دون الرسم خارجها اضغط على مفتاح Shift من لوحة المفاتيح عند الإضافة.

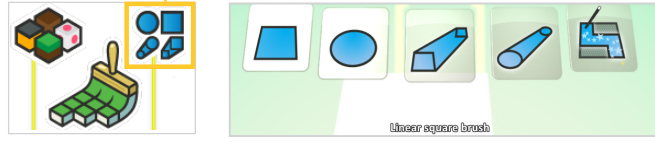
## تغيير مادة التضاريس بمادة موجودة في العالم نفسه

- وضع مؤشر الفأرة على المادة المطلوب أخذ عينة منها.
- اضغط على مفتاح Alt من لوحة المفاتيح.
- يظهر شكل مخروطي على المنطقة المحددة، و بالضغط على الزر الأيسر للفأرة يتم حفظ عينة المادة.
- انتقال مساحة التضاريس المطلوب تغيير مادتها والضغط عليها بالزر الأيسر للفأرة لتغيير المادة الحالية بالمادة الجديدة.



## تغيير شكل الفرشاة Brush Shape

### Brush Shape



إذا كان النمط المربع للفرشاة لا يحقق المظهر الذي تريده، فيمكنك ببساطة تبديل شكل الفرشاة إلى نمط دائري للحصول على تأثير أكثر سلاسة وانسيابية.

- اختيار أداة فرشاة الأرض من شريط الأدوات.
- ستظهر أيقونتان صغيرتان فوق الأداة (يميناً ويساراً).
- اختيار أيقونة Brush Shape الظاهرة في أعلى يمين أداة فرشاة الأرض.
- يمكن تعديل شكل الفرشاة بالانتقال من Square Brush إلى أحد الخيارات الأخرى مثل:

Hard round brush ■

Linear square brush ■

Linear hard round brush ■

Magic brush ■

- استخدام عجلة الفأرة للتنقل بين الخيارات، ثم اضغط بالزر الأيسر للفأرة على الشكل المطلوب للفرشاة لاعتمادها في تشكيل التضاريس الجديدة.



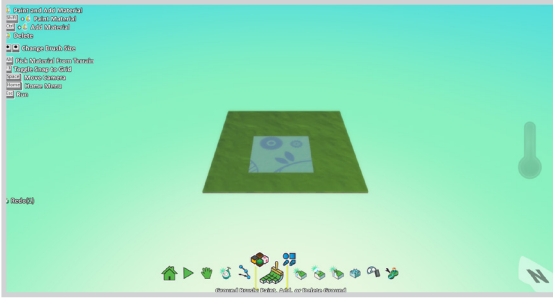
### جدول يوضح نوع الفرشاة ووظيفتها:

وظيفتها الأساسية	نوع الفرشاة
إنشاء تضاريس مربعة الشكل وهي الشكل الافتراضي عند رسم الأرض.	Square brush
إضافة تضاريس بحواف دائرية للحصول على شكل طبيعي أكثر للأرض.	Hard round brush
رسم خط مستقيم من التضاريس المربعة بين نقطتين محددتين (A, B).	Linear square brush
رسم خط مستقيم من التضاريس الدائرية بين نقطتين مع دقة أعلى في المحاذاة	Linear hard round brush
تغيير لون أو مادة التضاريس الموجودة ويمكن استخدامها لرفع أو خفض التضاريس .	Magic brush

## طرق إضافة التضاريس

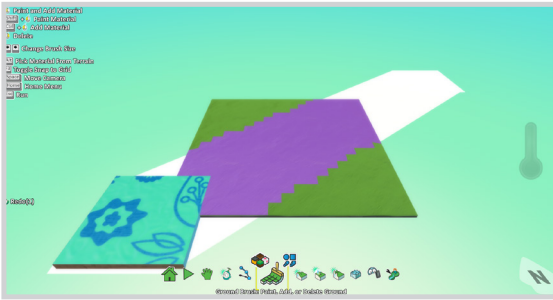
### ■ إضافة قطعة واحدة من التضاريس:

- الضغط بزر الفأرة الأيسر مرة واحدة في المكان المطلوب، ثم تركه لاعتماد الإضافة.



### ■ رسم خط من التضاريس:

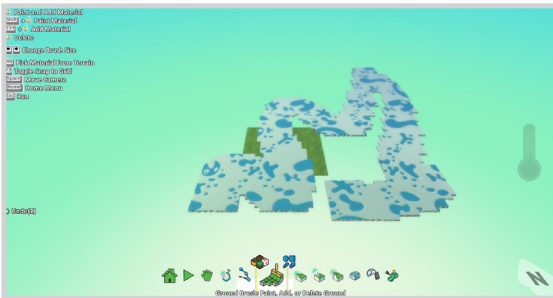
- وضع مؤشر الفأرة عند النقطة الأولى (A)، والاستمرار بالضغط على زر الفأرة الأيسر مع تحريك المؤشر باتجاه النقطة الثانية (B).



- عند الوصول إلى النقطة الثانية، يتم إفلات زر الفأرة لاعتماد الخط..

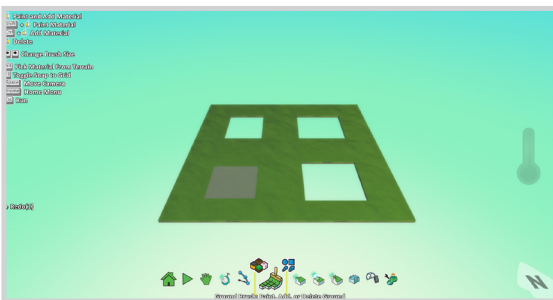
### ■ مسار متصل من التضاريس:

- الضغط باستمرار على زر الفأرة الأيسر وتحريك المؤشر في الاتجاه المطلوب، ليتم وضع تضاريس في كل نقطة يمر بها المؤشر.

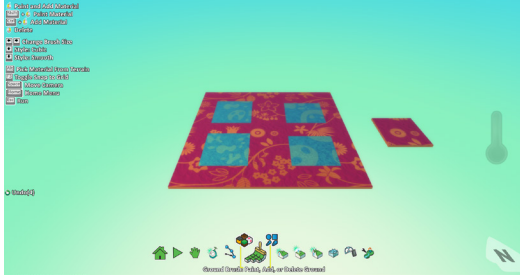


### ■ حذف قطعة من تضاريس:

- اختيار أداة فرشاة الأرض من شريط الأدوات، ثم وضع مؤشر الفأرة على القطعة المطلوب حذفها، ومن ثم ضغط الزر الأيمن للفأرة لإزالة القطعة.



أوضحنا مسبقاً كيفية استخدام مفتاح Alt مع الفرشاة لاستبدال لون التضاريس الموجودة بلون من العالم نفسه، ولكن لو استخدمت هذه الطريقة على قطعة كبيرة من التضاريس، فسيؤدي ذلك إلى تغيير لون جميع الأجزاء الصغيرة داخلها، تتيح أداة الفرشاة السحرية Magic Brush تحديد لون جزء معين فقط من التضاريس وتغييره دون التأثير على الأجزاء المحيطة، مما يمنح دقة أكبر في التعديل.



## النشاط



تخيل أنك مهندس متاهة ستستخدم أداة Ground Brush لرسم الممرات ومسح الزوائد، حتى تصنع متاهتك الأولى وتختبر ذكاء أصدقائك فيها.  
1. بدء المغامرة.

- تشغيل برنامج Kodu Game Lab واختيار الأمر New World.
- اختيار العالم الجديد من المكتبة الظاهرة، ليكون نقطة الانطلاق.



## 2. إضافة التضاريس إلى اللعبة.

- اختيار أداة فرشاة الأرض من شريط الأدوات:
- اختيار الأيقونة Terrain Material الظاهرة في أعلى يسار الأداة، وتحديد مادة المتاهة الأولى.
- اختيار الأيقونة Brush Shape الظاهرة في أعلى يمين الأداة، وحدد الفرشاة Square Brush لرسم أرضية وحوائط المتاهة المطلوبة.
- التحكم بحجم الفرشاة (تكبير/ تصغير) حسب حجم المتاهة التي ترغب بها.
- رسم أول جزء من المتاهة.
- تكرار عملية تغيير المادة إلى المادة الثانية للمتاهة، وتصغير حجم الفرشاة عند رسم كل طبقة.
- اختيار أيقونة Brush Shape الظاهرة في أعلى يمين أداة فرشاة الأرض، وتحديد الفرشاة Linear Square Brush، ثم استخدام المادة الأولى للمتاهة للإيحاء بمسح المساحات المطلوبة لتشكيل المتاهة.



### 3. إضافة الكائنات إلى اللعبة

- أضف بعض الكائنات إلى عالمك وأطلق العنان لخيالك.
- (كائن Tree - كائن rock) أو غيرها من الكائنات التي يمكن تخيلها وتناسب اللعبة.

### 4. اختبار العالم

- استخدام الكاميرا لتحريك المشهد من زوايا مختلفة ومراقبة توزيع العناصر.

### 5. توثيق الإنجاز

- حفظ اللعبة باسم المتعلم متبوعاً بكلمة Maze.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقاً.

## التطبيق



### ورقة عمل 11: رحلة إلى جزيرتي الرقمية

تخيل أنك مستكشف جزيرة رقمية جديدة، ستستخدم أداة Ground Brush لرسم شكل الجزيرة وتشكيل شواطئها وتضاريسها، حتى تصنع عالمك الخاص وتدع الآخرين يستكشفونه.



#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر New World.
- من المكتبة الظاهرة أمامك اختر العالم الجديد ليكون نقطة انطلاقك.

#### 2. تشكيل تضاريس الجزيرة

- اختر أداة فرشاة الأرض Ground Brush من شريط الأدوات.
- اختر أيقونة Terrain Material الظاهرة في أعلى يسار أداة فرشاة الأرض، وتحديد مادة خضراء لإنشاء قلب الجزيرة.
- اختر أيقونة Brush Shape الظاهرة في أعلى يمين أداة فرشاة الأرض وتحديد الفرشاة Square Brush أو Hard Round Brush لرسم شكل الجزيرة الأساسي في المنتصف.
- رسم أربع مناطق مختلفة حول مركز الجزيرة:
  - منطقة غابة (مادة خضراء).
  - منطقة رملية (مادة ذات لون أصفر أو برتقالي فاتح).
  - منطقة صخرية (مادة رمادية داكنة).
  - منطقة مائية (استخدام مادة أو لون أزرق حول حواف الجزيرة لتمثيل الماء).

#### 3. إضافة لمسات جمالية للجزيرة

- يمكنك إضافة بعض الكائنات البسيطة إلى عالمك لتجسيد الغابة، الصخور والماء على الجزيرة.
- اختر أماكن مناسبة لوضع هذه الكائنات بحيث تبدو الجزيرة واقعية ومنظمة.

#### 4. التجول داخل العالم

- استخدم الكاميرا لتحريك المشهد من زوايا مختلفة والتأكد من شكل الجزيرة وتوزيع المناطق الأربعة.
- تأكد من أن شكل الجزيرة متماسك، وأن الماء يحيط بها من الأطراف.

#### 5. حفظ وتصدير اللعبة

- احفظ العالم باسمك متبوعاً بكلمة Island.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك






أفهم وظيفة أداة فرشاة الأرض وكيفية استخدامها في بناء تضاريس العالم.




أنجح في إنشاء تضاريس مسطحة وتوسيع مساحة اللعب.




أتمكن من تغيير مادة التضاريس وتطبيقها بدقة.




أختار شكل الفرشاة الأنسب حسب نوع التضاريس التي أرغب في بنائها.




أحذف التضاريس غير المرغوبة بدقة لتحسين شكل العالم.




قادر على تصميم مخطط أولي منظم للعالم باستخدام مواد وفرش متعددة لرسم الممرات والحدود.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر





# الماء والمسارات وإعدادات العالم

### نواتج التعلم

- التعرف على وظيفة أدوات تشكيل التضاريس في بناء عالم اللعبة.
- تمييز الفرق بين تأثير أدوات تشكيل التضاريس في شكل الأرض وطبيعتها.
- استخدام أداة Up لرفع أجزاء من أرض اللعبة وتكوين جبال وهضاب.
- استخدام أداة Down لحفر وديان ومسارات لعب.
- استخدام أداة Flatten للحصول على تضاريس طبيعية أكثر وملمس واقعي للأرض.
- استخدام أداة Roughen لإضفاء مظهر متموخ وواقعي على التضاريس.
- استخدام أدوات التضاريس لابتكار تكوين مناسب لطبيعة اللعبة بشكل متكامل وجذاب.



ملفات أوراق العمل  
ملفات مصادر التعلم

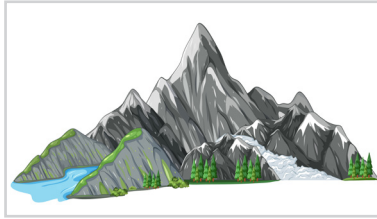




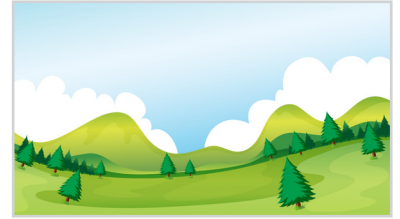
## الاستكشاف



وديان



جبال



تلال

■ كيف يؤثر اختلاف ارتفاعات الأرض في طريقة اللعب؟

■ ارسم تصورك الأولي لشكل عالم لعبتك.

## التعلم



تضاريس العالم هي الأساس الذي تبنى عليه تجربة اللعب، فارتفاع الأرض، انخفاضها، ونعومتها أو خشونتها تتحكم جميعها في حركة الكائنات وتحديات اللاعب والشعور بالواقعية داخل اللعبة.

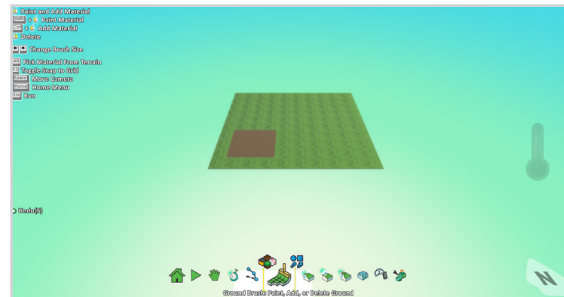
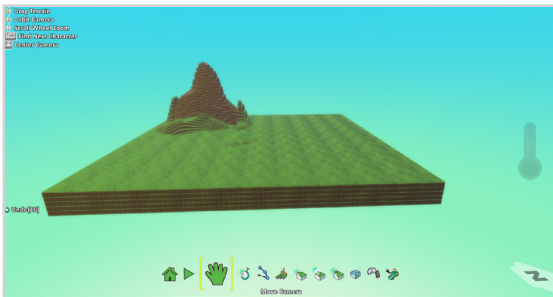
## أولاً: أداة رفع/خفض الأرض (UP/Down Tool)

### رفع الأرض



تستخدم أداة رفع الأرض لبناء جبال، منصات مرتفعة، أو أسطح عالية تتحرك فوقها الكائنات، ويتم رفع الأرض من خلال اتباع الخطوات التالية:.

- إنشاء عالم جديد.
- تصغير المشهد بالاستعانة بعجلة الفأرة لرؤية المساحة المحيطة.
- تكبير مساحة الأرض المسطحة لتوفير مساحة كافية لحركة الكائنات.
- اختيار أداة UP/Down Tool من شريط الأدوات.
- اختيار أيقونة Brush Shape الظاهرة في أعلى يمين الأداة.
- اختيار شكل الفرشاة المناسب لعالمك.
- وضع المؤشر في المكان المطلوب الضغط بالزر الأيسر للفأرة لرفع الأرض.
- الاستمرار بالضغط لزيادة الارتفاع تدريجياً.
- لتوسيع المساحة المرتفعة يتم تحريك الفأرة مع الاستمرار بالضغط.



يمكن استخدام السهمين (الأيمن / الأيسر) من لوحة المفاتيح لتكبير وتصغير قطر التأثير الخاص بالفرشاة

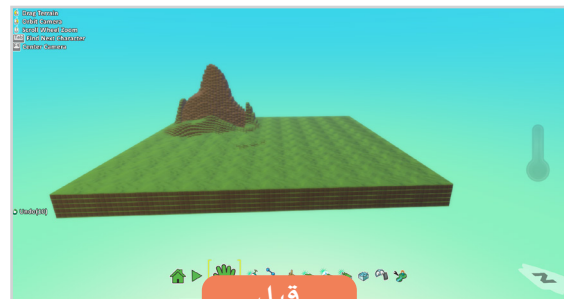
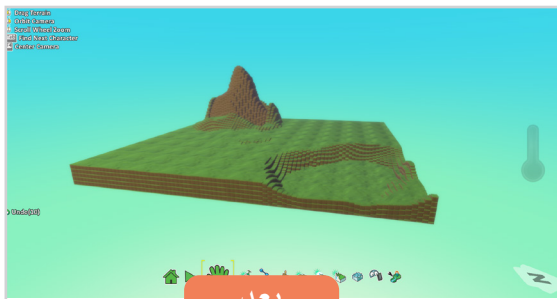


وظيفة الأساسية	نوع الفرشاة
تعديل التضاريس بشكل مربع مفيد للحواف، المنصات والأشكال الهندسية.	Square brush
رفع أو خفض التضاريس بدائرة ذات حواف واضحة مناسب للأعمال الدقيقة.	Hard round brush
تعديل التضاريس بدائرة متوسطة الحدة جيد لعمل تلال بسيطة.	Medium round brush
إنشاء تلال ووديان طبيعية ذات انحدار لطيف وحواف ناعمة.	Soft round brush
إنشاء تضاريس عشوائية وخشنة مثل الصخور والمناطق الطبيعية غير المنتظمة.	Mottled brush
تعديل تلقائي للتضاريس أو ملء منطقة كاملة بطريقة ذكية وسريعة.	Magic brush

### خفض الأرض

تساعدنا أداة خفض الأرض على صنع حفر ومنخفضات في العالم يمكننا حفر بحيرات، خنادق، أو مسارات حركة للكائنات.

- اختيار أداة UP/Down Tool من شريط الأدوات.
- اختيار أيقونة Brush Shape الظاهرة في أعلى يمين الأداة.
- اختيار شكل الفرشاة المناسب.
- وضع المؤشر في المكان المطلوب، ثم الضغط بالزر الأيمن للفأرة لخفض الأرض.
- الاستمرار بالضغط لزيادة الانخفاض تدريجياً.
- لتوسيع المساحة المنخفضة يتم تحريك الفأرة مع الاستمرار بالضغط.



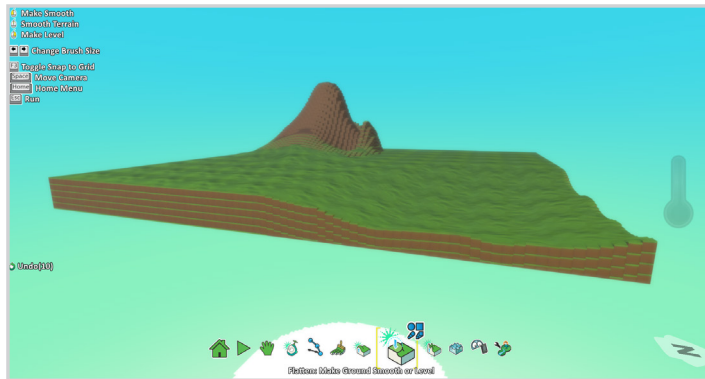
## ثانياً: أداة تسوية الأرض (Flatten Tool)



تستخدم أداة تسوية الأرض لتقليل حدة الجبال وإزالة الحواف غير المرغوبة، وهي مثالية لإنشاء مناطق آمنة ومسارات لعب سلسلة.

- اختيار أداة Flatten Tool من شريط الأدوات.
- اختيار أيقونة Brush Shape الظاهرة في أعلى يمين الأداة.
- اختيار شكل الفرشاة المناسب.
- وضع المؤشر في المكان المطلوب والضغط بالزر الأيسر للفأرة لتسوية الأرض.
- الاستمرار بالضغط لزيادة التسوية تدريجياً.
- لتوسيع مساحة التسوية، يتم تحريك الفأرة مع الاستمرار بالضغط.

الضغط بالزر الأيمن للفأرة باستخدام نفس الأداة لا يؤدي إلى تسوية للأرض.



## ثالثاً: أداة إنشاء أرض خشنة (Roughen Tool)



تُستخدم هذه الأداة لإضفاء خشونة وعدم استواء على الأرض، حيث تضيف نتوءات وتعرجات تجعل المسارات أكثر صعوبة وتحدياً للكائنات، من خلال اتباع الخطوات التالية:

- اختيار أداة Roughen Tool من شريط الأدوات.
- اختيار أيقونة Brush Shape الظاهرة في أعلى يمين أداة Roughen Tool.
- اختيار شكل الفرشاة المناسب لعالمك.

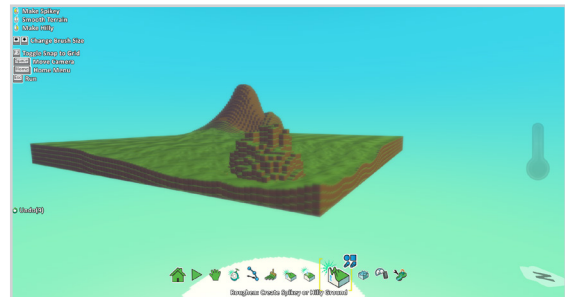
تضيف الأداة ارتفاعات أو انخفاضات عشوائية على الأرض:

- الزر الأيسر يضيف نتوءات طويلة جداً ومرتفعة بشكل كبير.
  - الزر الأيمن يضيف مرتفعات صغيرة ومنتظمة بمستويات أقل ارتفاعاً.
- أي أن الزر الأيسر في أداة Roughen Tool يصنع جبالاً عالية وحادة، بينما الزر الأيمن يصنع جبالاً أقل ارتفاعاً وأكثر نعومة.



ارتفاعات عالية وحادة

الزر الأيسر:



ارتفاعات صغيرة

الزر الأيمن:

## النشاط



تخيل أنك مصمم عالم مغامرات ستستخدم الأدوات الثلاث لإنشاء وادٍ محاط بجبال ومسارات منحدرية.

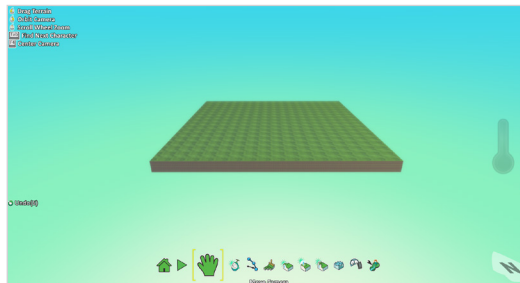
### 1. بدء المغامرة.

- تشغيل برنامج Kodu Game Lab واختيار الأمر NEW WORLD.
- اختيار عالم جديد ليكون نقطة الانطلاق.

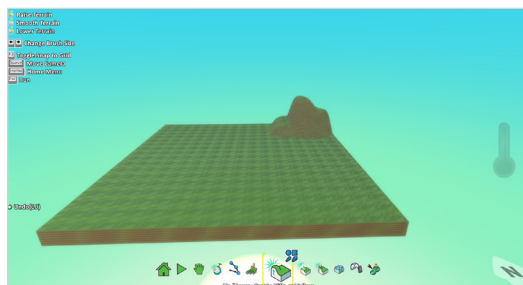


### 2. إضافة التضاريس إلى اللعبة.

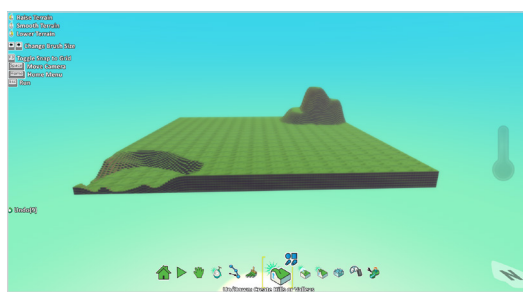
- اختيار أداة فرشاة الأرض من شريط الأدوات.
- رسم قطعة أرض كبيرة تسمح للكائنات بالتجول بحرية.
- اختيار أداة UP/Down Tool من شريط الأدوات.
- اختيار أيقونة Brush shape الظاهرة في أعلى يمين الأداة.
- اختيار شكل الفرشاة Magic brush ثم الضغط بالزر الأيسر للفأرة على الأرض للحصول على الارتفاع المناسب.



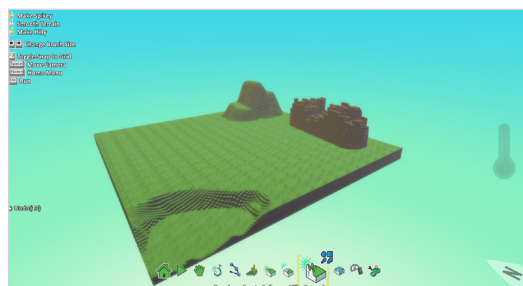
- تحديد الفرشاة Medium round brush لتكوين المنطقة الجبلية بالضغط على الزر الأيسر للفأرة بالمكان المطلوب.



- الضغط بالزر الأيمن للفأرة لخفض الأرض وتجهيز مساحة للبحيرة.

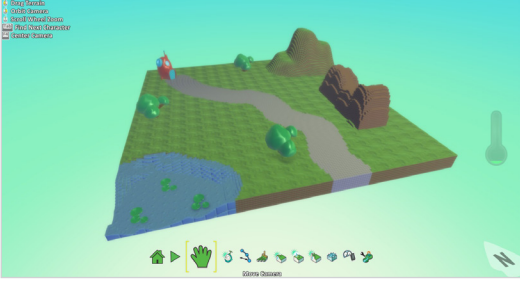


- إضافة خامة بلون مناسب للجبال الوعرة.
- اختيار أداة Roughen Tool لإضافة مرتفعات صغيرة ومنتظمة في منطقة الجبال الوعرة.
- استخدم أداة Flatten Tool مع Magic brush لتقليل حدة الجبال وإزالة الحواف غير المرغوبة.



### 3. إضافة الكائنات والخامات

- إضافة كائنات مثل Hut و Tree أو غيرها ليكون العالم أكثر واقعية..
- إضافة خامات مختلفة لث الحياة في عالم اللعبة ليبدو أقرب إلى الحقيقة.



## 4. اختبار العالم.

- استخدام الكاميرا لتحريك المشهد من زوايا مختلفة ومراقبة توزيع العناصر.

## 5. توثيق الإنجاز.

- حفظ اللعبة باسم المتعلم متبوعاً بكلمة Adventure.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.

## التطبيق

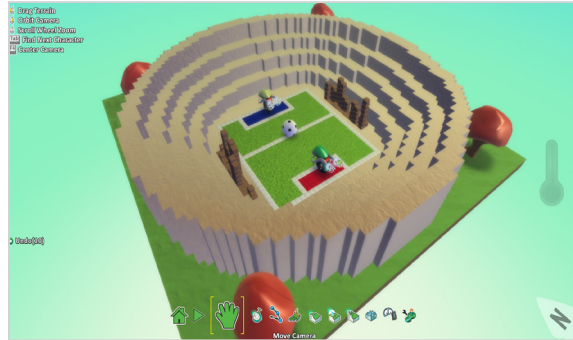


### ورقة عمل 12: مدرج التحدي

تخيل أنك مصمم ألعاب محترف، مهمتك اليوم هي إنشاء مدرج دائري متعدد الطبقات، وفي وسطه ملعب كرة قدم جاهزة للمنافسة بين روبوتين.

#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر NEW WORLD.
- اختر عالم جديد (مسطح) ليكون نقطة انطلاقك.



#### 2. تشكيل تضاريس الملعب.

- ارسم قطعة أرض كبيرة مناسبة للمشروع.
- ارفع الأرض.
- ارسم قطعة أرض مناسبة لملعب كرة القدم (فرشة دائرية الشكل).
- ارفع أرض الملعب لأعلى نقطة تناسب عالمك.
- أنشئ مدرج الملعب من خلال تصغير حجم فرشة الملعب ومهارة خفض الأرض بالتدرج.
- ارسم منطقة ملعب كرة القدم.
- أضف حاجزاً على جانبي الملعب (يميناً ويساراً) لمنع الجماهير من النزول إلى منطقة اللاعبين.

## 3. إضافة العناصر الأساسية للمنافسة

- أضف لاعبين Cycle مع تغيير لون وحجم كل لاعب.
- أضف كرة قدم Ball وغيّر حجمها لتناسب المشروع.
- أضف كائن Tree مع تغيير لونه وحجمه ثم كرره أربع مرات.
- اختر أماكن مناسبة لوضع هذه الكائنات بحيث يبدو الملعب واقعي ومنظم.

## 4. التجول داخل العالم

- استخدم الكاميرا لتحريك المشهد من زوايا مختلفة والتأكد من شكل الجزيرة وتوزيع المناطق الأربع .

## 5. حفظ وتصدير اللعبة

- احفظ العالم باسمك متبوعاً بكلمة Arena.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أوظف أدوات تشكيل التضاريس لتعديل شكل الأرض في عالم اللعبة.

أميّز بين تأثير كل أداة على تضاريس العالم.

أنشئ جبال ومنخفضات وتضاريس طبيعية.

أصمم مخطط تضاريس بهدف إثراء تجربة اللعب وتعزيز التفاعل.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

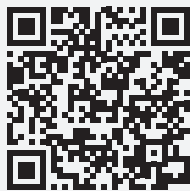
## ملاحظات ولي الأمر



## تابع: الماء والمسارات وإعدادات العالم

### نواتج التعلم

- التعرف على وظيفة أداة Water Tool في إضافة المسطحات المائية داخل العالم الرقمي .
- إضافة المسطحات المائية والتحكم في منسوب الماء وتدرجه.
- التمييز بين أنماط الماء المختلفة الظاهرة في Water Palette واختيار النمط المناسب للعبة.
- استخدام أداة Change World Setting لتشكيل البيئة البصرية للعالم الرقمي.
- ضبط عناصر المساعدة وبدء اللعبة حسب الحاجة.
- تصميم عالم رقمي متكامل يوظف الماء وخصائص العالم بشكل إبداعي.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ ما العناصر التي تجعل عالم اللعبة يبدو واقعياً؟

■ هل تعتقد أن تغيير الإضاءة والسماء يمكن أن يغير شعور اللاعب داخل اللعبة؟

■ عندما ترى في اللعبة سطح مائي ما الذي تتوقع أن يضيفه هذا العنصر للتجربة؟

■ لماذا تحتاج بعض الألعاب إلى عدّ تنازلي قبل البدء، بينما تبدأ أخرى مباشرة؟

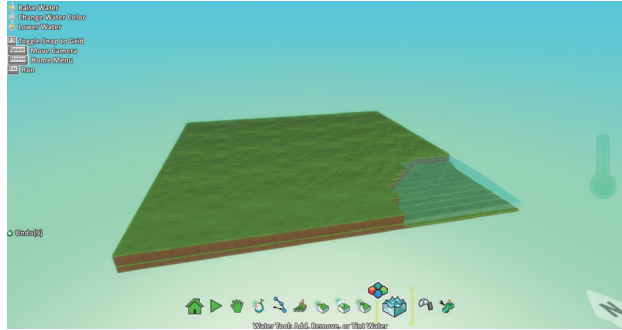
## التعلم



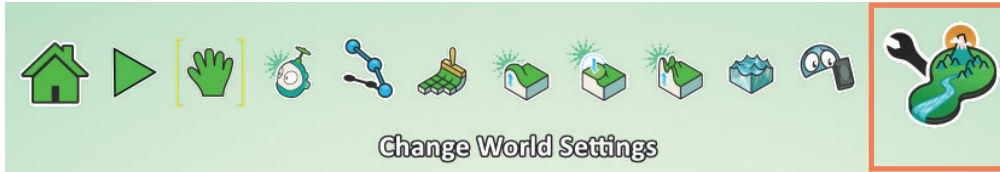
مع كل أداة نضيف عنصراً جديداً يجعل اللعبة أكثر واقعية وتنبض بالحياة، سنمنح اللعبة مظهرها الخاص من إضافة الماء ورسم المسارات إلى تغيير السماء والإضاءة، كل أداة ستقربنا خطوة نحو عالم رقمي متكامل وجذاب يعبر عن خيالك.



لتقليل منسوب الماء يتم اتباع خطوات إضافة الماء مع استخدام الزر الأيمن للفأرة.




## ثانياً: أداة تغيير إعدادات العالم Change World Settings Tool



تستخدم أداة تغيير إعدادات العالم مفتاح التحكم الكامل في خصائص العالم الافتراضي قبل بدء البرمجة أو أثناءها، فمن خلالها يتم ضبط عناصر أساسية مثل الإضاءة، السماء، الماء، الرياح، وزوايا الكاميرا، لتحول العالم الرقمي بيئة بسيطة إلى بيئة نابضة بالحياة وأكثر واقعية، مما يشري تجربة المستخدم ويعزز جاذبية اللعبة.

### عناصر المساعدة داخل العالم

الوظيفة	الشكل	العنصر
يُظهر بوصلة صغيرة أسفل يمين الشاشة تشير نحو الشمال (الأمام بالنسبة للكائن).		
يعرض مقياساً لكمية الموارد المستخدمة في العالم (عدد الكائنات، التأثيرات..الخ) ليساعد المصمم على التعرف على الحد الأقصى، ويظهر في حالة التصميم بشكل مقياس يمين الشاشة.		
يفعل نظاماً يحد من عدد العناصر التي يمكن إضافتها، حتى لا تصبح اللعبة ثقيلة أو بطيئة أثناء التشغيل.	----	

لإلغاء تفعيل أي من خيارات إعدادات العالم، يكون زر  لاحظ أن زر التشغيل يكون باللون الأخضر، بالضغط عليه ضغطة واحدة بالزر الأيسر للفأرة يتم إلغاء تفعيل الخيار.

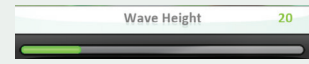


## إعدادات عنصر الماء

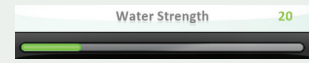
### الوظيفة

### الإعداد

التحكم في ارتفاع الأمواج على سطح عنصر الماء، القيمة الصغيرة تعطي ماء هادئاً بينما القيمة الأكبر تعطي أمواجاً عالية وواضحة



التحكم في قوة حركة الماء وشدة تموج سطحه كلما زادت القيمة أصبح الماء أكثر حركة.



## إعدادات السماء والإضاءة

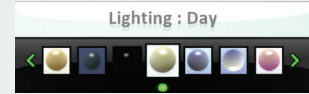
### الوظيفة

### الإعداد

تغيير خلفية السماء وألوان الأفق يمكن اختيار سماء نهائية زرقاء، غروب، سماء ضبابية... وغيرها من الخيارات.



ضبط شدة ونوع الإضاءة العامة للعالم مما يؤثر على ألوان الكائنات وظلالها .



- عند اختيار إعداد إضاءة داكن (Dark)، سيظهر العالم مظلماً فقط عند تشغيل اللعبة، أما أثناء مرحلة التحرير فستظل الإضاءة ساطعة للتمكن من رؤية العالم بوضوح.
- الجمع بين إعدادات السماء (Sky) والإضاءة (Lighting) يساعد في إنشاء الجو المناسب للعبة.

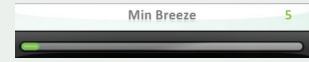


## إعدادات الرياح

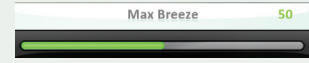
### الوظيفة

### الإعداد

ضبط نعومة هبوب النسيم أثناء مروره بين الأشجار داخل اللعبة.

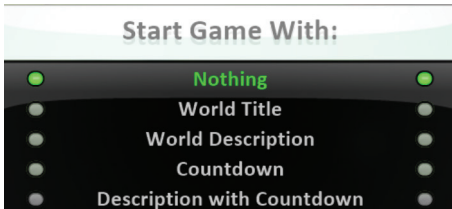


ضبط أقصى شدة هبوب النسيم أثناء مروره بين الأشجار داخل اللعبة.



## ما يظهر عند بدء اللعبة

هو الخيار المسؤول عن كيفية ظهور العالم عند بدء اللعب.



### الوظيفة

### الخيار الفرعي

يبدأ اللعب مباشرة دون إظهار أي عنوان أو نص.

Nothing

يعرض اسم العالم لمدة 3 ثوان ثم يبدأ اللعب تلقائياً.

World Title

يعرض اسم العالم ويطلب من اللاعب الضغط على زر A لبدء اللعب، حيث يعتمد العرض على الوصف الذي تم ادخاله عند حفظ العالم.

World Description

يعرض عدداً تنازلياً (1، 2، 3) قبل بدء اللعبة وهذا مفيد جداً في ألعاب السباق.

Countdown

يعرض وصف العالم مع عد تنازلي، ثم يبدأ اللعب.

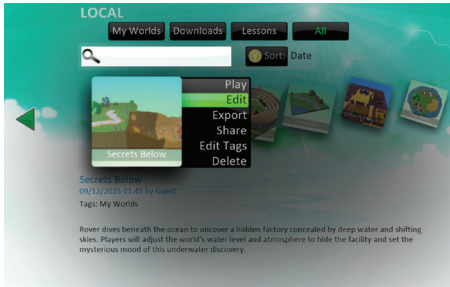
Description with  
Countdown

## النشاط



أنت في مهمة لإخفاء مصنع تحقيق الأحلام تحت الماء ستستخدم الأدوات لإضافة مسطحات مائية مناسبة للعبة مع التحكم بالجو العام للعبة باحترافية من خلال خيارات إعدادات العالم.

### 1. بدء مغامرتك



- تشغيل برنامج Kudu Game Lab واختيار الأمر IMPORT FROM DESKTOP.

- استدعاء مشروع Secrets Below من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.

### 2. إضافة الماء إلى اللعبة.



- اختيار أداة Water Tool من شريط الأدوات.
- اختيار نمط الماء المناسب من Water Palette.
- إضافة الماء لغمر المصنع وإخفائه مع المخلوقات البحرية بشكل مناسب.

### 3. ضبط بيئة اللعبة.

- اختيار أداة Change World Settings من شريط الأدوات للتحكم في:

■ إعدادات عنصر الماء:

90 = Wave Height

100 = Water Strength



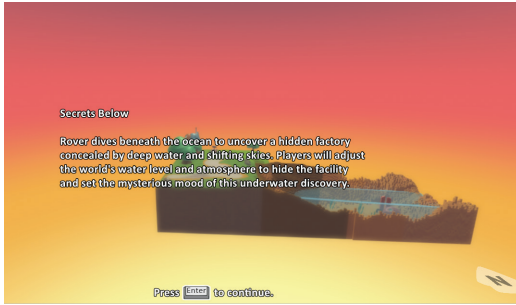
#### ■ إعدادات السماء والإضاءة:

20

= Sky

Dream

= Lighting



#### 4. ما يظهر عند بدء اللعبة.

■ اختيار أداة Change World Settings

■ من شريط الأدوات.

■ اختيار Start Game With.

■ من بطاقة الخيارات يتم اختيار World Description

■ ليظهر وصف اللعبة للاعب عند بداية التشغيل.

#### 5. اختبار العالم.

■ تشغيل اللعبة وملاحظة كيف أثرت الإضافات

■ (الماء، الإضاءة، السماء) على شكل العالم

■ أثناء اللعب.

■ ملاحظة ظهور الوصف للاعب عند التشغيل.

#### 6. توثيق الإنجاز.

■ حفظ اللعبة باسم المتعلم متبوعا بكلمة Hidden.

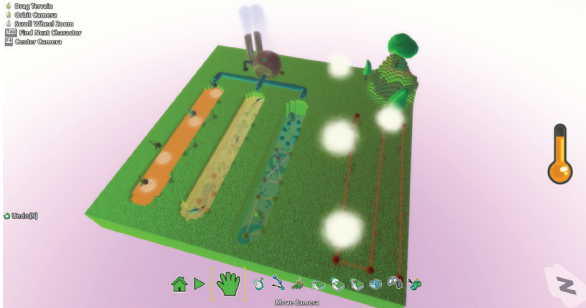
■ تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقا.



## التطبيق



### ورقة عمل 13: مزرعة الأسماك



تخيل أنك مصمم ألعاب محترف، مهمتك اليوم هي تصميم مزرعة أسماك صناعية داخل عالم رقمي نابض بالحياة، ستعمل على إضافة وتعديل لون الماء وارتفاع الأمواج، ثم ضبط السماء والإضاءة بهدف خلق تجربة لعب مريحة ومشوقة وكأن اللاعب يزور مزرعة حقيقية.

#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر .IMPORT FROM DESKTOP
- استعد مشروع Aqua Fish Farm من مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. إضافة الماء إلى اللعبة.

- اختر نمط ماء مختلف من Water Palette لتمييز كل حوض.
- أضف الماء لغمر الأسماك في كل حوض بشكل مناسب.
- اضبط إعدادات الماء:

60 = Water Strength

85 = Wave Height

#### 3. اضبط بيئة اللعبة:

Venus = Lighting

12 = Sky

80 = Max Breeze

50 = Min Breeze

Description with countdown = Start Game with

#### 4. اختبار اللعبة.

- استخدم تشغيل اللعبة ولاحظ كيف أثرت إضافاتك (الماء، الإضاءة، السماء) على شكل العالم أثناء اللعب.
- لاحظ ظهور الوصف للاعب عند التشغيل وعند الضغط على مفتاح Enter من لوحة المفاتيح يبدأ العد التنازلي للعبة.

#### 5. توثيق الإنجاز.

- حفظ اللعبة باسم المتعلم متبوعاً بكلمة Farm.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك

أضيف الماء للعالم وأتحكم بخصائصه.

أختار نمط الماء المناسب للعبة.

أتحكم في إعدادات العالم لتشكيل بيئة لعب متكاملة.

أستخدم عناصر المساعدة وواجهة البدء وفق حاجة اللعبة.

أصمم عالماً رقمياً متكامل بشكل إبداعي.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

# الخطوة الأولى نحو البرمجة

### نواتج التعلم

- التعرف على برمجة الكائنات داخل Kodu Game Lab.
- التمييز بين أوامر الشرط When وأوامر الاستجابة Do.
- اختيار Tiles المناسبة لبناء سلوك صحيح للكائن داخل اللعبة.
- تعديل الأوامر البرمجية من خلال خيارات Tiles المختلفة.
- حذف الأوامر البرمجية بطريقة صحيحة.
- تنظيم أسطر الأوامر البرمجية وترتيبها حسب أولوية التنفيذ.
- تطبيق مبدأ التجربة والتصحيح لتحسين سلوك الكائن داخل اللعبة.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ ما الفرق بين عالم يحتوي على كائن ساكن وآخر يمكنه الحركة؟

---

---

■ اذكر اسم لعبة تحتاج الشخصية فيها إلى القفز؟ ولماذا؟

---

---

■ كيف ستبدو تجربة اللعب لو لم نتمكن من التحكم بالشخصية؟

---

---

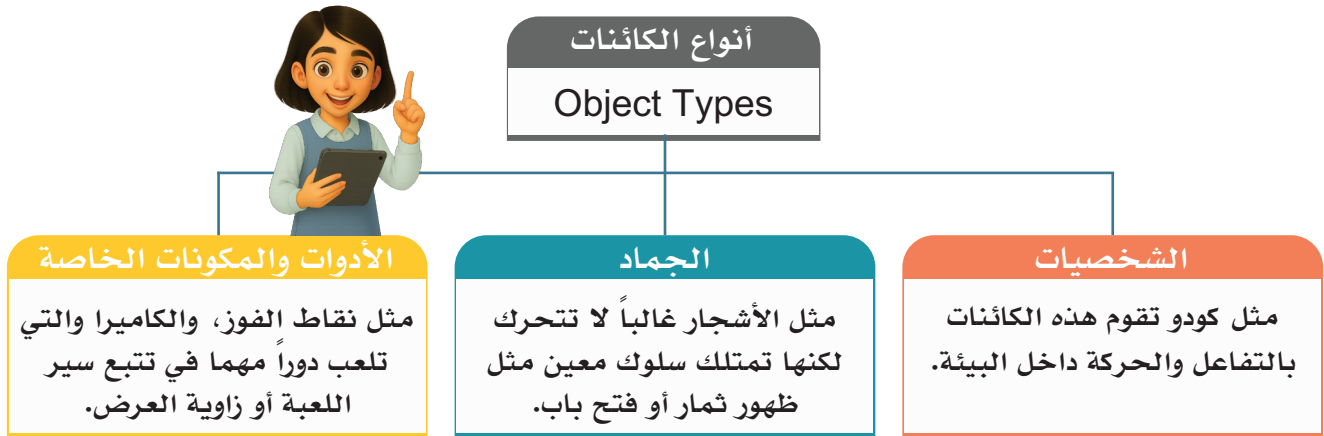
## التعلم



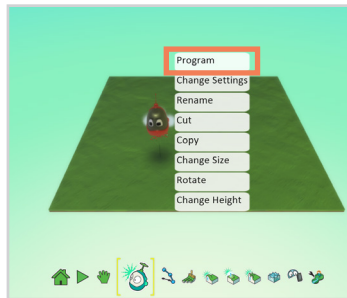
تعرفنا حتى الآن على طريقة إضافة الكائنات إلى عالمك الرقمي لكنها تقف بلا حركة،  
حان الوقت لنمنح الكائنات القدرة على الاستجابة للأوامر والبدء بالحركة، وهي أول  
خطوة حقيقية نحو البرمجة داخل اللعبة.

## برمجة الكائنات Object Programming

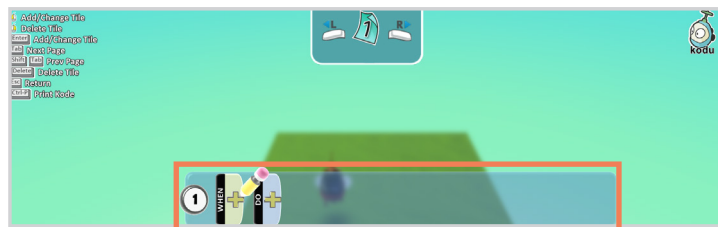
يعتمد البرنامج على أسلوب برمجة الكائنات حيث يتعامل مع كل عنصر داخل العالم الافتراضي باعتباره كائناً مستقلاً يمتلك سلوكاً خاصاً به يمكن برمجته والتعديل عليه، ولكل كائن صفحة برمجة خاصة يتم فيها تحديد استجابته للأحداث داخل اللعبة



يتم برمجة كل كائن بشكل مستقل من خلال تحديد استجابته للمؤثرات داخل اللعبة، بعدها يتم اختيار أداة Object Tool من شريط الأدوات وتحديد الكائن المطلوب برمجته ثم الضغط بالزر الأيمن للفأرة واختيار الأمر Program.



عند الدخول إلى وضع البرمجة ستظهر صفحة البرمجة الخاصة بالكائن وبها مستطيل:



أول سطر برمجي مرقم يتضمن مربعين أساسيين هما **When** و **Do**، وبجانب كل منهما رمز + الذي يشير إلى إمكانية إضافة أوامر لكل منهما.



- مؤشر القلم يساعدك في معرفة أي خانة تعمل عليها في تلك اللحظة.
- في كل مرة تبدأ بوضع Tile في السطر البرمجي يتم إضافة سطر جديد تلقائياً بعده.

## أولاً: البرمجة Programming Tiles

تعد Tiles هي أساس بناء الأوامر البرمجية في Kodu Game Lab، ويعتمد هذا النظام على صيغة بسيطة هي:

1. عندما يحدث شيء ما When: الحدث أو الشرط الذي يجب أن يحدث أولاً.
2. تنفيذ إجراء معين Do: الاستجابة التي ينفذها الكائن عند تحقق الحدث أو الشرط.

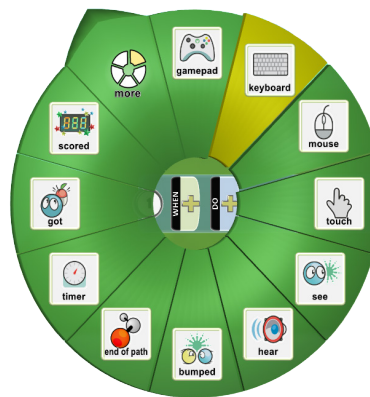
مثال:



عندما يتم ضغط أزرار الأسهم من لوحة المفاتيح، يتحرك الكائن باتجاه السهم المضغوط بسرعة

### WHEN أوامر

عند الضغط على علامة + بجانب مربع When تظهر عجلة الخيارات التي تحتوي على الأحداث أو الشروط التي تحدد متى يبدأ الكائن في تنفيذ الأوامر، وهي:



متى يحدث	الحدث أو الشرط
عند ضغط اللاعب على زر في ذراع التحكم.	gamepad
عند ضغط اللاعب على مفاتيح معينة في لوحة المفاتيح.	keyboard
عند ضغط اللاعب على أزرار الفأرة أو تحريكها.	mouse
عند لمس الكائن لكائن آخر.	touch
عند رؤية الكائن لكائن آخر أو عنصر معين في بيئة اللعبة.	see
عند سماع الكائن صوتاً.	hear
عند اصطدام الكائن بكائن أو عنصر معين في بيئة اللعبة.	bumped
عند وصول الكائن لنهاية المسار المحدد له.	end of path
عرض الخيارات الإضافية غير الظاهرة في القائمة الحالية .	more
بعد مرور وقت معين (مؤقت).	timer
عند حصول الكائن على شيء أو التقاط غرض.	got
عند تغيير نقاط اللاعب (ربح / خسارة).	scored

## أوامر Do

عند الضغط على علامة + بجانب مربع Do تظهر عجلة الخيارات Tiles التي تحدد ما سيفعله الكائن عند تحقق الحدث أو الشرط الخاص به.



الممثل الأسود

## ما يحدث عند تحقق الحدث

## الاستجابة

التحرك (باتجاه معين، تتبع مسار، سرعة الحركة... إلخ).	move
الاستدارة (يمين، يسار، اتجاه معين، سرعة الدوران... إلخ).	turn
أكل أو التقاط شيء.	eat
تنفيذ أفعال عامة (القفز، التحدث، إنشاء عنصر جديد، تغيير لون الكائن... إلخ).	actions
تعديل إعدادات الكائن (تغيير حجم الكائن، تغيير شكل السماء، تحديد كيف يحمل الكائن الأشياء... إلخ).	settings
الانتقال إلى صفحة برمجة أخرى.	switch
السماح بإضافة أكثر من فعل داخل نفس السطر دون فتح سطر برمجي جديد.	inline
إطلاق قذائف أو صواريخ.	shoot
أوامر (هجوم، دفاع، شفاء... إلخ).	combat
التعامل مع الأشياء التي يمسكها الكائن (رمي، إسقاط، استخدام... إلخ).	holding
أوامر مرتبطة باللعبة (النقاط، الفوز، الخسارة... إلخ).	game
التحكم بمنظور اللاعب والكاميرا (زاوية الرؤية، متابعة كائن... إلخ).	view

■ ظهور مثلث أسود على بعض شرائح الأوامر (كما هو موضح بالشكل السابق)، يعني أن هذا الأمر يحتوي على خيارات إضافية تظهر عند اختياره.

■ ظهور علامة + بجانب Tile بعد إضافتك له، تعني أن هذا الأمر يحتاج إلى تفصيل أكثر يجب تحديده حتى يفهم البرنامج متى ينفذ الأمر وذلك حسب احتياجات اللعبة.

### مثال:

■ في مربع When عند اختيارك Keyboard Tile تظهر علامة + بعدها، عند الضغط عليها تظهر لك تفاصيل لوحة المفاتيح ويمكنك اختيار Arrows Tile وبذلك يكتمل الأمر بشكل واضح للبرنامج.

■ عند إضافة خيار not إلى الشرط فإنك تطلب من البرنامج تنفيذ الأمر عندما لا يتحقق الشرط بدلاً من تحققه، هذا يساعدك على إنشاء أوامر أكثر تنوعاً وذكاءً داخل اللعبة.

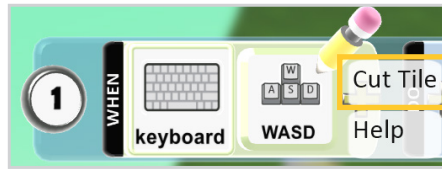


## ثانياً: حذف وتعديل الأوامر البرمجية

أثناء برمجة الكائنات قد لا ينفذ الكائن تصورنا لسلوكه من المرة الأولى، لهذا يمكن بسهولة تعديل الأوامر أو حذفها لتجربة أفكار جديدة أو تصحيح الأخطاء، وذلك لتحسين اللعبة وتطويرها خطوة بخطوة بطريقة بسيطة.

### حذف الأوامر

- الضغط بالزر الأيمن للفأرة على الأمر Tile المطلوب حذفها.
- تظهر قائمة فرعية يتم اختيار الأمر Cut Tile.



بعض Tiles تمثل الأمر الأساسي مثل Tile Keyboard، بينما تمثل الأخرى تفاصيل لهذا الأمر مثل WASD Tile. عند حذف الأوامر التفصيلية مثل WASD Tile سيبقى الأمر الأساسي موجوداً في السطر البرمجي ولن يختفي، ولكن لن يتم تنفيذه فعلياً إلى أن يتم إضافة تفاصيل له. عند حذف الأمر الأساسي مثل Keyboard Tile يتم إزالة الأمر كاملاً وما يتبعه من تفاصيل.



### تعديل الأوامر

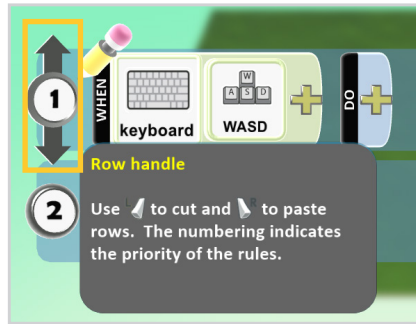


عند الضغط بالزر الأيسر للفأرة على أي أمر برمجي Tile تظهر عجلة الخيارات والتي تساعد على تعديل الأمر واختيار بدائل تناسب الفكرة، من خلالها يمكن تغيير نوع الشرط أو تفاصيله.

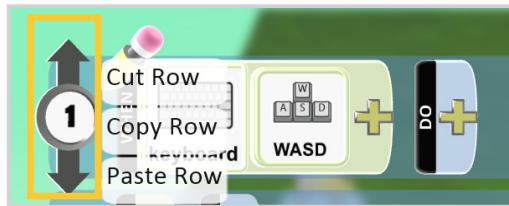
## تحرير السطر البرمجي Row handle

كل سطر برمجي له رقم يحدد أولويته في التنفيذ، ويمثل سلوكاً مستقلاً للكائن، حيث يقرأ البرنامج الأسطر من الأعلى إلى الأسفل مما يساعدك على فهم أي سلوك يتم تنفيذه أولاً.

لتحرير السطر البرمجي كاملاً يتم الضغط بالزر الأيسر للفأرة على رقم السطر المطلوب، ستظهر مقابض حول رقم السطر، مما يعني أنه جاهز للتحريك أو التعديل:



- استخدام السحب والإفلات بالضغط على الزر الأيسر للفأرة لتحريك موقع السطر البرمجي (أعلى/أسفل) الصفحة.
- الضغط بالزر الأيمن للفأرة واختيار الأمر Cut Row من القائمة لقص السطر البرمجي كاملاً لنقله إلى مكان آخر عن طريق لصقه من خلال الأمر Paste Row.
- الضغط بالزر الأيمن للفأرة واختيار الأمر Copy Row من القائمة لنسخ السطر البرمجي كاملاً ثم لصق نسخه منه في المكان المطلوب من خلال الأمر Paste Row.



## النشاط



أنت في مهمة لمساعدة Kodu على التحرك في جميع الاتجاهات للوصول إلى الغابة وتجنب عوائق الطريق من خلال القفز فوقها.

### 1. بدء المغامرة

- تشغيل برنامج Kodu Game Lab واختيار الأمر IMPORT FROM DESKTOP.
- استدعاء مشروع The Forest من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.



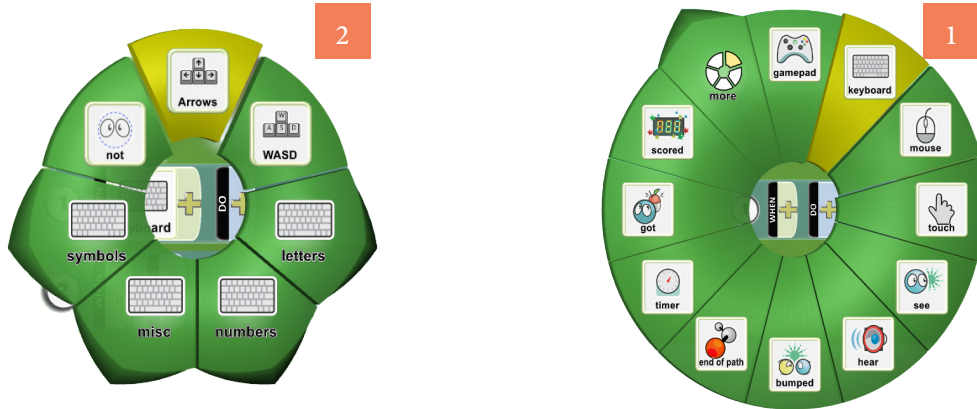
### 2. تحريك الكائن باستخدام مفاتيح الأسهم

- برمجة (الحدث أو الشرط).
- 1. اختيار أداة Object Tool من شريط الأدوات.
- 2. الضغط بالزر الأيمن للفأرة على الكائن Kodu.
- 3. اختيار الأمر Program من القائمة الفرعية.
- 4. الضغط على علامة + الظاهرة بجانب مربع When.



### 5. اختيار Keyboard Tile من الخيارات الظاهرة.

6. الضغط على علامة + الظاهرة بجانب Keyboard Tile لإضافة تفصيل التحرك باستخدام مفاتيح الأسهم، وذلك باختيار Arrow Tile.

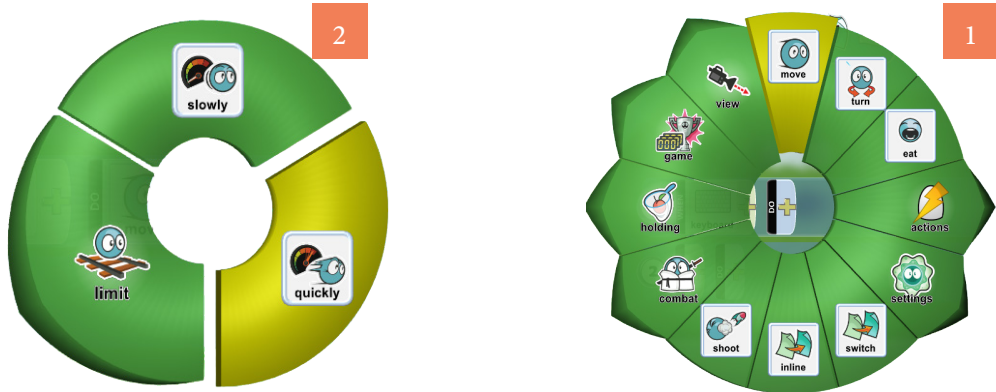


■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.

2. اختيار Move Tile من الخيارات الظاهرة.

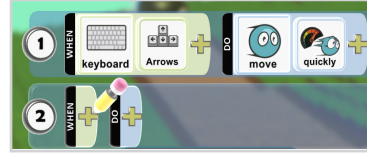
3. الضغط على علامة + الظاهرة بجانب Move Tile لإضافة تفصيل السرعة المطلوبة، وذلك باختيار Quickly Tile.



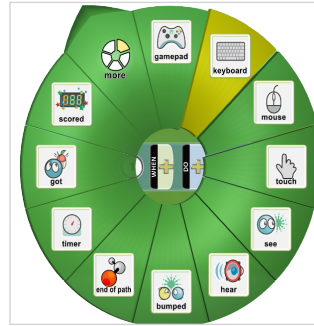
## 4. برمجة الكائن للقفز

■ برمجة (الحدث أو الشرط)

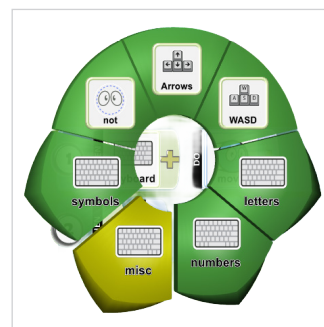
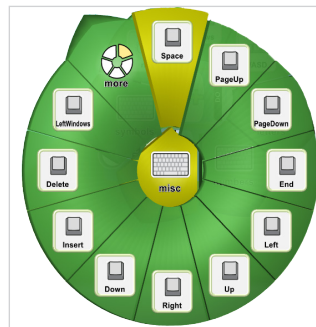
1. الانتقال إلى السطر الثاني لبرمجة الكائن Kodu، والضغط على علامة + الظاهرة بجانب مربع When.



2. اختيار Keyboard Tile من الخيارات الظاهرة.



3. الضغط على علامة + الظاهرة بجانب Keyboard Tile لإضافة تفصيل القفز باستخدام مفتاح المسافة، وذلك باختيار Misc Group، ستظهر مجموعة خيارات إضافية في لوحة المفاتيح يتم اختيار Space Tile.

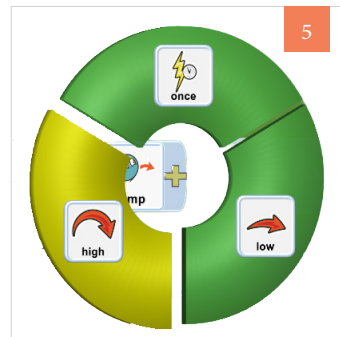
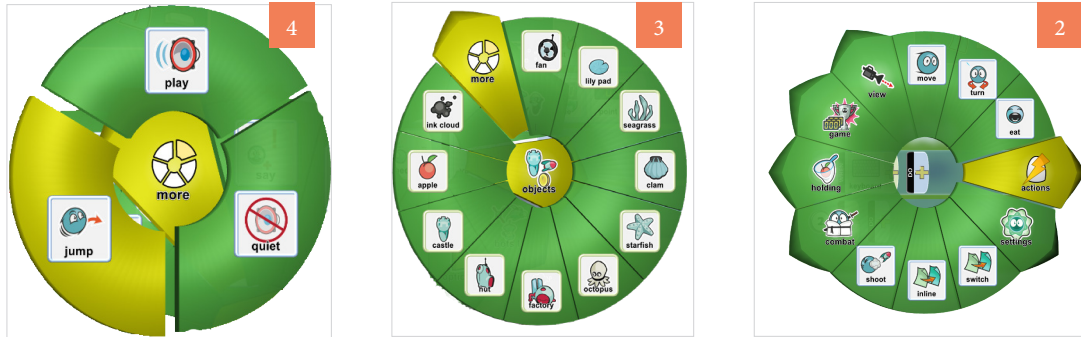


■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار Action Group من الخيارات الظاهرة.
3. اختيار More Group من الخيارات الظاهرة.
4. اختيار Jump Tile من الخيارات الظاهرة.

5. الضغط على علامة + الظاهرة بجانب Jump Tile لإضافة تفصيل الارتفاع المطلوب، وذلك باختيار High Tile.

6. الضغط على علامة + الظاهرة بجانب High Tile لإضافة تفصيل عدد القفزات، وذلك باختيار Once Tile.



7. اختبار العالم

- تشغيل اللعبة وملاحظة سلوك Kodu أثناء اللعب.
- تعديل المقطع البرمجي عند الحاجة.

8. توثيق الإنجاز

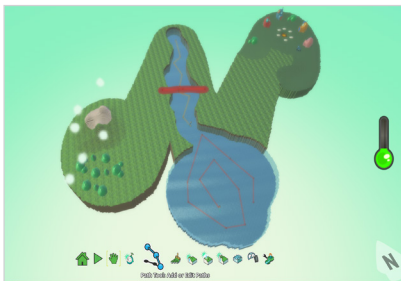
- حفظ اللعبة باسم المتعلم متبوعا بكلمة Adventure.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقا.

## التطبيق



### ورقة عمل 14 : الرحلة

تخيل أنك مستكشف تقود روفر Rover في رحلة ممتعة بين الأرياف والجبال، مهمتك هي إيصال روفر بأمان ليستمتع بمغامرة استكشاف من خلال التحرك بسلسلة باستخدام مفاتيح الأسهم ولا تنسى أن تمكنه من تجاوز العقبات بقفزات سلسلة.



#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر IMPORT .FROM DESKTOP
- استدع مشروع The Journey من مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. برمجة Rover

- أضف البرمجة اللازمة لتحريك Rover من خلال الأسهم في لوحة المفاتيح بأقصى سرعة.
- أضف البرمجة اللازمة لجعل Rover يقفز عاليا في كل مرة يتم الضغط بها على مفتاح X من لوحة المفاتيح.

#### 3. اختبار اللعبة

- شغل اللعبة ولاحظ سلوك Rover أثناء اللعب.
- عدّل المقطع البرمجي عند الحاجة.

#### 4. حفظ وتصدير اللعبة

- احفظ العالم باسمك متبوعاً بكلمة Journey.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أختار Tiles المناسبة لإنشاء سلوك صحيح للكائن داخل اللعبة.

أعدّل الأوامر البرمجية باستخدام خيارات Tiles المختلفة.

أحذف الأوامر البرمجية عند الحاجة.

أعدّل البرمجة عند ظهور خطأ أو استجابة غير صحيحة في سلوك الكائن.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر



### إنشاء وتتبع المسار

#### نواتج التعلم

- التعرف على وظيفة أداة Path Tool واستخدامها داخل اللعبة.
- إنشاء مسارات مخصصة للكائنات وتحديد نقطة البداية والنهاية.
- تعديل المسارات من حيث النوع، الارتفاع، اللون والشكل بما يخدم تصميم اللعبة.
- التحكم بعقد المسار من خلال الإضافة، الحذف والتحرك.
- برمجة الكائنات لتتبع مسار محدد تلقائياً داخل اللعبة.
- تطبيق مهارات التفكير المنطقي والتخطيط المسبق في توجيه حركة الكائنات داخل اللعبة.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ ما أهمية استخدام المسارات لتنظيم حركة الكائنات داخل اللعبة؟

■ أيهما أنسب لحركة الكائن: مسار مفتوح أم مسار حلقي؟ ولماذا؟

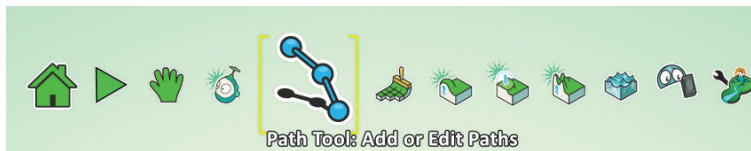
■ كيف يؤثر تغيير نوع أو لون المسار على وضوح حركة الكائن داخل اللعبة؟

## التعلم



تعرفنا حتى الآن على كيفية تحريك الكائنات داخل العالم الرقمي، لكن حركتها لازالت عشوائية، حان الوقت لتتعلم كيف نرسم للكائنات مسارات ونجعلها تتبعها بدقة، لتصبح الحركة منظمة داخل اللعبة.

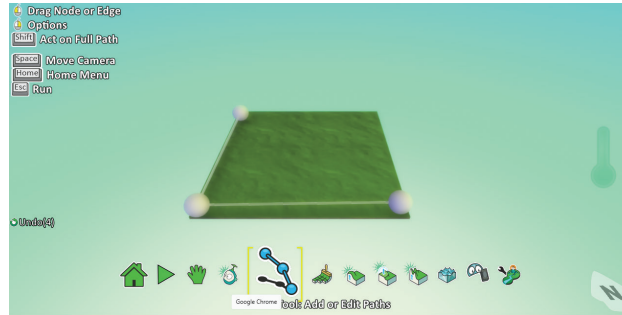
## أداة المسار Path Tool



تستخدم أداة المسار لإنشاء طريق أو خط مخصص داخل العالم قد يستخدم لتوجيه مسار حركة كائن أو للزينة أو لتقسيم المناطق، يتم رسم المسار من خلال اتباع الخطوات التالية:

1. اختيار أداة Path Tool من شريط الأدوات.
  2. الضغط بالزر الأيمن للفأرة على موقع نقطة بداية المسار المطلوب، ستظهر قائمة فرعية بأنواع المسارات.
  3. اختيار نوع المسار المناسب للعبة من القائمة.
  4. الضغط بالزر الأيسر للفأرة للبدء بإضافة العقد (نقاط) وتشكيل المسار المطلوب.
- كل ضغطة زر أيسر على الأرض تضيف عقدة جديدة للمسار ويرسم البرنامج خطاً بينها وبين العقدة السابقة
5. الضغط ضغطتين متتاليتين بالزر الأيسر على آخر عقدة تم إضافتها سيعتبرها البرنامج نهاية المسار.
  6. سيظهر المسار مرسوماً على الأرض فوراً.

إذا ضغطت ضغطاً مزدوجاً في النهاية على أول عقدة رسمتها يغلق المسار ويصبح مساراً حلقياً.



## أنواع المسارات

تختلف المسارات في البرنامج بحسب شكلها وطريقة استخدامها، لتناسب حركة الكائنات داخل العالم الرقمي.

- Add Plain Path
- Add Wall
- Add Road
- Add Flora

### Wall Path

مسار يتحول إلى جدار مرتفع أو حاجز.  
يستخدم لبناء متاهات، وضع حواجز لمنع الكائنات من المرور.



### Plain Path

مسار بسيط بدون زخارف أو جدران.  
يستخدم غالباً لحركة الكائنات (سباق، تجول...الخ).

### Flora Path

مسار مغطى بالنباتات أو الأزهار.  
يستخدم لإضافة طابع جمالي للعالم (حديقة، مسارات طبيعية).

### Road Path

مسار يشبه الطريق أو الشارع.  
يستخدم لإبراز الطرق الرئيسية في العالم أو مسارات الحركة.

## التحكم بالمسارات

بعد رسم المسار يمكن التحكم به وتعديله من حيث الشكل والخصائص، وذلك للحصول على مسار يناسب تصميم العالم وحركة الكائنات.

■ أولاً: تحريك نقاط المسار

■ تحريك عقدة واحدة:

لتعديل موقع عقدة محددة ضمن المسار يجب الضغط عليها بالزر الأيسر للفأرة ثم سحبها وإفلاتها في الموضع الجديد، سيغير شكل المسار مباشرة تبعاً لموضع العقدة الجديد.

■ تحريك المسار بالكامل:

لنقل المسار كاملاً يجب الضغط على مفتاح Shift من لوحة المفاتيح مع الضغط بالزر الأيسر للفأرة على المسار وسحبه ثم إفلاته في المكان الجديد محافظاً على شكل المسار الأصلي.

- Add More Nodes
- Change Height
- Change Type
- Delete

## ■ **ثانياً:** إضافة عقد للمسار.

تحديد العقدة والضغط عليها بالزر الأيمن للفأرة تظهر قائمة فرعية، اختيار الأمر Add More Nodes، سيتم إضافة عقدة بين العقدة الحالية والتي تليها.

## ■ **ثالثاً:** تغيير ارتفاع المسار.

لتغيير ارتفاع المسار (رفع أو خفض العقدة) يتم تحديد العقدة والضغط عليها بالزر الأيمن للفأرة، حيث تظهر قائمة فرعية يتم اختيار الأمر Change Height.

## ■ **رابعاً:** تغيير نوع المسار.

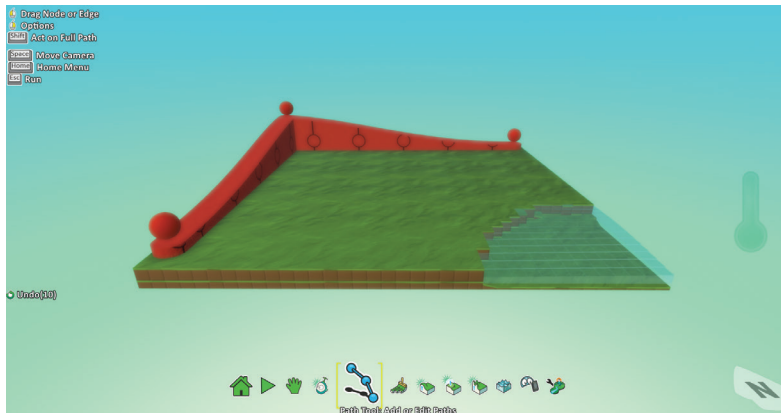
لتغيير نوع المسار (طريق، حائط، منصة.... الخ) يتم تحديد العقدة والضغط عليها بالزر الأيمن للفأرة، حيث تظهر قائمة فرعية يتم اختيار الأمر Change Type.

## ■ **خامساً:** حذف نقطة من المسار.

لحذف العقدة يتم تحديدها والضغط عليها بالزر الأيمن للفأرة، حيث تظهر قائمة فرعية يتم اختيار الأمر Delete.

## ■ **سادساً:** تغيير لون المسار.

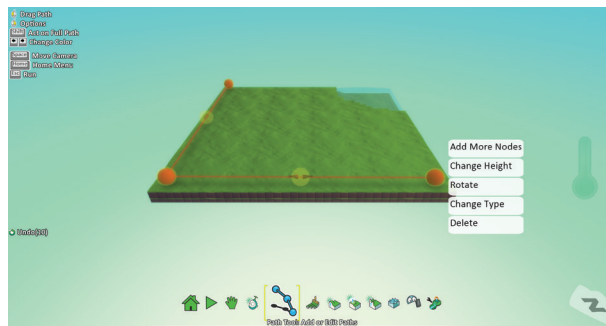
عند المرور على أي من عقد المسار يظهر شريط الألوان يمكن استخدام عجلة الفأرة أو أسهم لوحة المفاتيح للتنقل بين الألوان عند التوقف عند لون محدد سيغير لون عقد المسار كاملاً تبعاً لما تم اختياره.





عند الضغط على مفتاح Shift من لوحة المفاتيح مع المرور بمؤشر الفأرة على أي من عقد المسار ستلاحظ أن البرنامج فَعَلَ خاصية تحرير المسار كاملاً (يتم تحديد المسار كاملاً باللون الأصفر). تظهر قائمة فرعية جديدة خاصة بتحرير المسار كاملاً في حال تحديده بالكامل ثم الضغط بالزر الأيمن على أي من عقد المسار ، حيث ستظهر مجموعة من الخيارات هي:

- Add More Nodes : إضافة نقاط جديدة للمسار.
- Change Height : رفع وخفض المسار بالكامل.
- Rotate : عمل استدارة للمسار حول نقطة ارتكاز دون الحاجة لإعادة رسمه.
- Change Type : تغيير نوع المسار (Plain, Road, Wall, Flora).
- Delete : حذف المسار كاملاً بجميع عقدة وخطوطه.

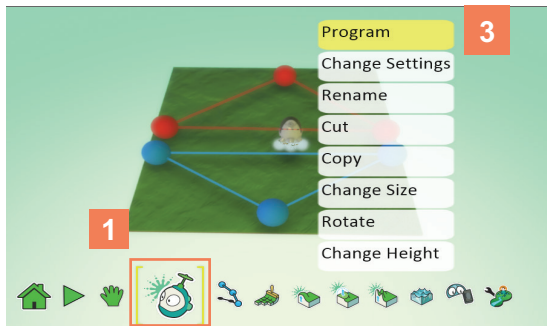


## تتبع المسارات

بعد ضبط المسار بشكل مناسب حان الوقت لبرمجة الكائنات لتتبعه تلقائياً لتصبح الحركة موجهة بتعليمات مناسبة.

لبرمجة الكائن لتتبع الخط باستمرار وبشكل تلقائي.

■ برمجة (الحدث أو الشرط).



1. اختيار أداة Object Tool من شريط الأدوات.
2. الضغط بالزر الأيمن للفأرة على الكائن المطلوب.
3. اختيار الأمر Program من القائمة الفرعية.
4. الضغط على علامة + الظاهرة بجانب مربع When.
5. اختيار More Group من الخيارات الظاهرة.
6. اختيار Always Tile من الخيارات الظاهرة.



■ برمجة الاستجابة.

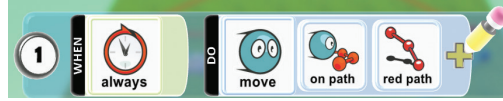
1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار Move Tile من الخيارات الظاهرة.
3. الضغط على علامة + الظاهرة بجانب Move Tile ، لإضافة تفصيل تتبع المسار، وذلك باختيار On path Tile.



## مسارات متعددة وكائنات مختلفة

في حال وجود أكثر من مسار في اللعبة يختار الكائن أقرب مسار له تلقائياً، لجعل الكائن يتتبع مسار محدد:

- تخصيص لون مختلف لكل مسار.
- برمجة الحدث ليتتبع الكائن المسار دائماً باختيار Always Tile.
- برمجة الاستجابة لتتبع المسار مع إضافة تفصيل اللون:
  1. الضغط على علامة + الظاهرة بجانب مربع Do.
  2. اختيار Move Tile من الخيارات الظاهرة.
  3. الضغط على علامة + الظاهرة بجانب Move Tile لإضافة تفصيل تتبع المسار، وذلك باختيار On path Tile.
  4. الضغط على علامة + الظاهرة بجانب On path Tile لإضافة تفصيل لون المسار، وذلك باختيار Path color Group.
  5. اختيار لون المسار المطلوب تتبعه.



## النشاط



صمم طريقاً داخل المتاهة وساعد العربة على الوصول إلى مصنع الطاقة الشمسية باستخدام المنطق، والتخطيط، واتخاذ قرارات ذكية.

### 1. بدء المغامرة



- تشغيل برنامج Kodu Game Lab واختيار الأمر . IMPORT FROM DESKTOP
- استدعاء مشروع Solar Path Mission من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.

### 2. رسم المسار.



- اختيار أداة Path Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة عند موقع بداية المسار ثم اختيار Add Plain Path.
- استكمال رسم المسار بالضغط بالزر الأيسر لإضافة العقد للوصول إلى المصنع، ثم الضغط ضغطاً مزدوجاً على آخر عقدة لإنهاء رسم المسار.
- تغيير لون المسار إلى اللون البرتقالي

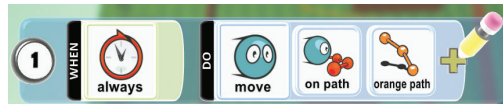
## 3. التعديل على المسار.

- تعديل ارتفاع مسار المتاهة ذو اللون الأزرق.
- اختيار أداة Path Tool من شريط الأدوات.
- تحديد أحد نقاط المسار والضغط على مفتاح Shift من لوحة المفاتيح لتحديد الجزء المطلوب كاملاً.
- الضغط بالزر الأيمن للفأرة واختيار الأمر Change Height وتعديل الارتفاع ليساوي 5.
- تحديد المسار ذو اللون الأحمر بجانب المصنع.
- الضغط بالزر الأيمن للفأرة واختيار الأمر Change Type لتحويله إلى سياج.



## 4. برمجة Rover لتتبع المسار.

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن Rover والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- برمجة الكائن لتتبع المسار البرتقالي دائماً.



## 5. اختبار العالم.

- تشغيل اللعبة وملاحظة حركة Rover.
- تعديل المقطع البرمجي عند الحاجة.

## 6. توثيق الإنجاز.

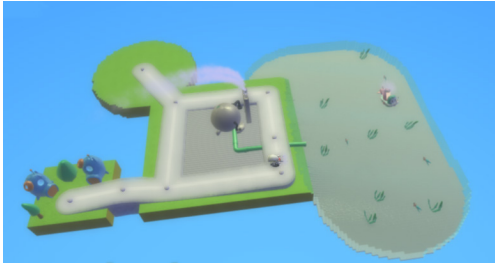
- حفظ اللعبة باسم المتعلم متبوعاً بكلمة Mission.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقاً.

## التطبيق



### ورقة عمل 15: مصنع الأسماك

تخيل أنك مصمم ألعاب محترف، مهمتك اليوم هي التعديل على مسار الكائن Cycle ثم برمجته لتتبع المسار، ورسم مسار للكائن Ship ليتبعه باستمرار داخل البحيرة.



#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر IMPORT FROM DESKTOP.
- استدع مشروع Fishing Factory مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. الكائن Cycle.

- غير لون المسار الذي يقف عليه الكائن إلى اللون الأزرق.
- أضف البرمجة اللازمة لمساعدة الكائن على تتبع المسار باستمرار.

#### 3. الكائن Ship.

- أضف مسار ليتحرك الكائن داخل البحيرة بشكل منتظم دون الاصطدام بالأعشاب.

#### 4. اختبار العالم.

- شغل اللعبة ولاحظ حركة Rover.
- عدّل المقطع البرمجي عند الحاجة.

#### 5. حفظ وتصدير اللعبة.

- احفظ العالم باسمك متبوعاً بكلمة Arena.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



- أستخدم أداة Path Tool لإنشاء مسارات مناسبة داخل اللعبة.
- أنشئ مسارات تحدد نقطة البداية والنهاية للكائنات.
- أعدّل المسار من حيث النوع، الارتفاع، اللون، والشكل بما يخدم تصميم اللعبة.
- أبرمج الكائنات لتتبع مسار محدد داخل اللعبة.
- أخطط حركة الكائنات مسبقاً باستخدام التفكير المنطقي.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

## التفاعل بين الكائنات والحصول على المكافآت

### نواتج التعلم

- التعرف على مفهوم جمع العناصر داخل اللعبة.
- برمجة الكائنات لتستجيب للاصطدام.
- استخدام عدّاد النقاط لاحتساب النتائج.
- التمييز بين كائن يمنح نقاطاً وكائن يُعد عائقاً.
- برمجة الكائنات للاختفاء عند الحاجة.
- ضبط إعدادات الكائنات بما يخدم هدف اللعبة.
- تطبيق مهارات التفكير المنطقي في تصميم لعبة تعتمد على جمع النقاط.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ لماذا تعتمد الكثير من الألعاب على جمع العناصر؟

---

---

■ ماذا تتوقع أن يحدث عند اصطدام كائن متحرك بعائق ثابت؟

---

---

■ اذكر أمثلة لأفكار ألعاب يتم فيها تجميع الكائنات والحصول على مكافآت؟

---

---

## التعلم



تعرفنا سابقاً على كيفية تحريك الكائنات داخل اللعبة، لكن دون تفاعل حقيقي مع البيئة المحيطة، في هذا الدرس سنتعلم كيف نجعل الكائنات تتفاعل عند التصادم واحتساب المكافآت لتكون اللعبة أكثر تشويقاً.

# التفاعل بين الكائنات والحصول على المكافآت

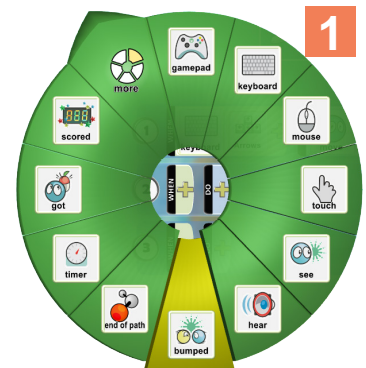
## أولاً: جمع الكائنات

تعتمد العديد من الألعاب الرقمية على مهارة جمع الكائنات حيث يجمع اللاعب عناصر معينة لزيادة النقاط أو التقدم في المراحل.

استكمال برمجة الكائن Kodu لأكل التفاح عند الاصطدام به، من خلال اتباع الخطوات التالية :

### ■ برمجة (الحدث أو الشرط)

1. اختيار أداة object Tool من شريط الأدوات.
2. الضغط بالزر الأيمن للفأرة على الكائن Kodu.
3. اختيار الأمر Program من القائمة الفرعية.
4. الضغط على علامة + الظاهرة بجانب مربع When.
5. اختيار bumped Tile من الخيارات الظاهرة.
6. الضغط على علامة + الظاهرة بجانب bumped Tile وإضافة تفصيل الكائن الذي يجب الاصطدام به، وذلك باختيار object Group ثم اختيار apple Tile.



### ■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار eat Tile من الخيارات الظاهرة.
3. الضغط على علامة + الظاهرة بجانب eat Tile لإضافة تفصيل أكل التفاح، وذلك باختيار .it Tile



## ثانياً: تحصيل النقاط

تسهم مهارة تحصيل النقاط عند جمع العناصر في إضفاء طابع من التشويق والمتعة على اللعبة، حيث يشعر اللاعب بالحماس والتحمدي مع زيادة النقاط كلما نجح في جمع عناصر أكثر

■ برمجة (الحدث أو الشرط)

■ في السطر البرمجي الجديد يتم إضافة Tiles بنفس خطوات برمجة الكائن Kodu للتفاعل مع apple عند الاصطدام به.

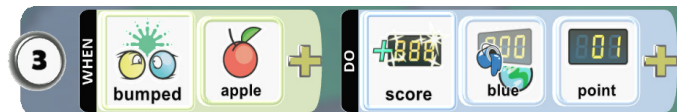
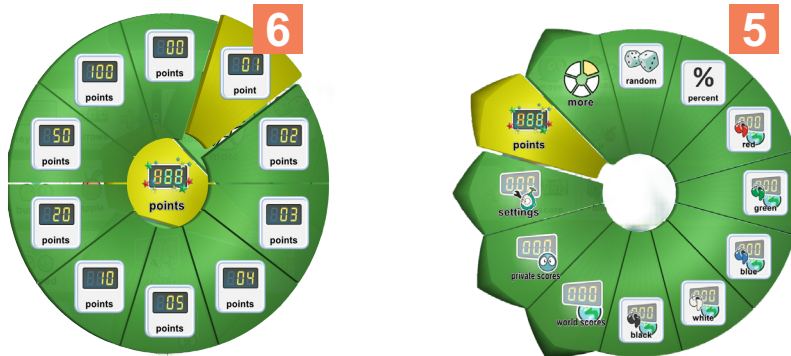
■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار game Group من الخيارات الظاهرة.
3. اختيار Tile score + من الخيارات الظاهرة.
4. الضغط على علامة + الظاهرة بجانب score Tile لإضافة تفصيل لوحة النتائج الخاصة بالكائن حسب لونه، وذلك باختيار blue Tile.

# التفاعل بين الكائنات والحصول على المكافآت



5. الضغط على علامة + الظاهرة بجانب blue Tile لإضافة تفصيل عدد النقاط التي سيكتسبها الكائن عن كل تفاحة، وذلك باختيار points Group ثم تحديد نقطة واحدة لكل تفاحة 01 point.



- في خيارات games Group يوجد لديك خيار لبدء النتيجة برقم محدد، لإضافة نقاط، أو إنقاص نقاط.
- في خيارات points Group يوجد لديك خيارات متعددة لعدد النقاط التي سيكتسبها الكائن في كل مرة.
- بمجرد تفعيل نظام النقاط، تظهر لوحة game Board أعلى يمين الشاشة أثناء تشغيل اللعبة.



## ثالثاً: إخفاء العوائق

تسهم مهارة التحكم في إظهار وإخفاء العوائق في رفع مستوى صعوبة اللعبة، حيث تستخدم العوائق كتحديات تتطلب من اللاعب تركيزاً أكبر للتقدم في اللعبة، ولتنفيذ ذلك يتم اتباع الخطوات التالية:

### ■ برمجة (الحدث أو الشرط)

■ في سطر برمجي جديد يتم إضافة Tiles بنفس خطوات برمجة الكائن kodu للتفاعل مع كائن آخر عند الاصطدام به، بحيث يتم إضافة تفصيل الكائن الذي يجب الاصطدام به من object Group يتم اختيار more Group ثم rock Tile.

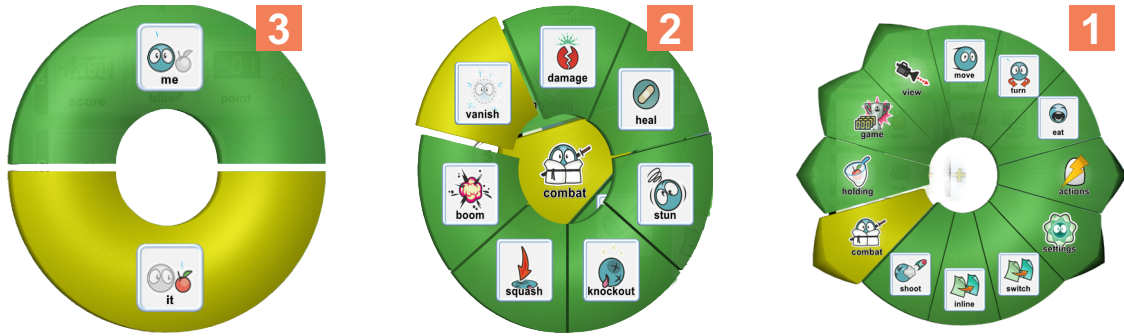


### ■ برمجة الاستجابة

- الضغط على علامة + الظاهرة بجانب مربع Do.
- اختيار combat Group من الخيارات الظاهرة.
- اختيار vanish Tile من الخيارات الظاهرة.
- الضغط على علامة + الظاهرة بجانب vanish Tile لإضافة اختفاء الصخرة وليس اللاعب نفسه، وذلك باختيار Tile it.



# التفاعل بين الكائنات والحصول على المكافآت



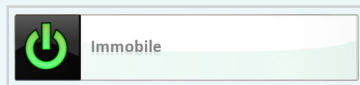
## رابعاً: تثبيت الكائن

تستخدم خاصية تثبيت الكائن في تحويل العنصر إلى عائق لزيادة صعوبة اللعبة دون أن يغير موقعه.



- اختيار أداة object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن apple.
- اختيار الأمر Change Settings من القائمة الفرعية.
- تفعيل خيار Immobile من إعدادات الكائن.

■ الخاصية Immobile تعني تثبيت موقع الكائن في مكانه بحيث لا يستطيع التحرك في أي اتجاه، ولكن يمكنه الاستمرار في الدوران حول نفسه.



## النشاط



ساعد Kodu على جمع كل القطع النقدية من خلال الاصطدام بها لكسب النقاط، انتبه فالصخور تعيق الطريق وتجعل اللعبة أكثر تحدياً، لا تنسى أن تثبت عناصر اللعبة حتى يكون التخطيط للفوز واضح.

### 1. بدء المغامرة

- تشغيل برنامج Kodu Game Lab واختيار الأمر IMPORT FROM DESKTOP.
- استدعاء مشروع Coin Quest من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.

### 2. برمجة kodu

- تحديد أداة object Tool من شريط الأدوات.
- تحديد الكائن kodu والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم ليصطدم الكائن Kodu بالكائن coin وعند حدوث الاصطدام يتم جمع (أكل) القطع النقدية.



- في سطر برمجي جديد يتم إضافة Tiles لبرمجة الكائن kudo للتفاعل مع الكائن coin بحيث أنه عند حدوث الاصطدام يتم احتساب نقطة للاعب.



# التفاعل بين الكائنات والحصول على المكافآت

## 3. برمجة Kodu لتفادي العوائق

- في سطر برمجي جديد يتم إضافة Tiles لبرمجة الكائن Kodu للتفاعل مع الكائن rock بحيث انه عند حدوث الاصطدام تختفي الصخرة ويصبح الطريق مفتوح أمام اللاعب لاستكمال الطريق.



## 4. تثبيت عناصر اللعبة

- اختيار أداة Object Tool من شريط الأدوات.
- تتم هذه الخطوة لكل كائن على حدة : الضغط بالزر الأيمن للفأرة على جميع الكائنات coin و rock.
- اختيار الأمر change Settings من القائمة الفرعية.
- تفعيل خيار Immobile من إعدادات الكائن.

## 5. اختبار العالم

- تشغيل اللعبة وجمع القطع النقدية وإزالة العوائق عن الطريق.
- تعديل المقطع البرمجي عند الحاجة.

## 6. توثيق الإنجاز

- حفظ اللعبة باسم المتعلم متبوعا بكلمة Quest.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقا.



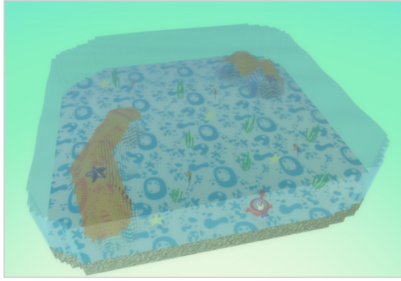
## التطبيق



### ورقة عمل 16: كنز البحار

تخيل أنك مصمم ألعاب محترف، مهمتك اليوم هي مساعدة Sub للتحرك تحت الماء وجمع النجوم المتألئة لكسب النقاط.

انتبه للأسماك المتجولة فعند اصطدامها بالكائن Sub تختفي، مما يفتح الطريق ويضيف لمسة مشوقة للمغامرة. استكشف بحذر واجمع كل النجوم لتحقيق النجاح.



#### 1. بناء العالم

■ شغل برنامج Kodu Game Lab واختر الأمر IMPORT FROM DESKTOP.

■ استدع مشروع Star Dive من مجلد أوراق العمل.

■ ابدأ بتحرير اللعبة.

#### 2. الكائن Sub

■ أضف البرمجة اللازمة للكائن للاصطدام بكائن Star لأكله وتحصيل 5 نقاط عن كل نجمة.

■ أضف البرمجة اللازمة لمساعدة الكائن على إخفاء الكائن Fish في حال الاصطدام به.

#### 3. الكائن 1,2 Starfish

■ فعل خاصية التثبيت Immobile للكائنات.

#### 4. اختبار العالم

■ شغل اللعبة مع الحرص على جمع كل النجوم وتفادي الأسماك.

■ عدّل المقطع البرمجي عند الحاجة.

#### 5. حفظ وتصدير اللعبة

■ احفظ العالم باسمك متبوعاً بكلمة Collection.

■ صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أبرمج الكائنات لتستجيب للاصطدام وأميز بين كائن يمنح نقاطاً وكائن يعد عائقاً.

أستخدم عداد النقاط لاحتساب النتائج بدقة أثناء اللعب.

أبرمج الكائنات لتختفي عند الحاجة وذلك لرفع مستوى التحدي.

أضبط إعدادات الكائنات بما يخدم هدف اللعبة.

أطبق مهارات التفكير المنطقي في تصميم لعبة جمع النقاط وتخطيط خطواتها.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

## برمجة الفوز والخسارة في اللعبة

### نواتج التعلم

- التعرف على خيار تفعيل حدود العالم.
- برمجة الكائن ليتتبع كائن آخر.
- استخدام خيار الفوز والخسارة لإنهاء اللعبة.
- برمجة وحساب النقاط لتحقيق الهدف من اللعبة.
- تطبيق مهارات التفكير المنطقي في تصميم لعبة الفوز والخسارة.
- التمييز بين خيارات المقارنة لحساب نقاط الفوز والخسارة.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ لماذا تُصمَّم بعض الألعاب بحيث لا يمكن الفوز إلا بعد إتمام جميع المهام؟

■ ما الذي يظهر عند تحقيق الفوز بعد إتمام جميع المهام؟

■ ما الذي يظهر عند خسارتك في تحقيق جزء من المهام في اللعبة؟

## التعلم



تعرفنا سابقاً على كيفية التفاعل بين الكائنات عند التصادم واحتساب النقاط لتكون اللعبة أكثر تشويقاً، و خلال هذا الدرس سنتعلم على برمجة ظهور رسالة الفوز Win أو الخسارة Lose في اللعبة.

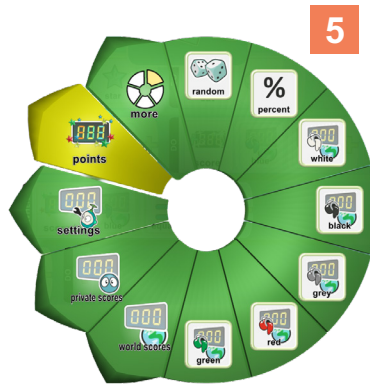
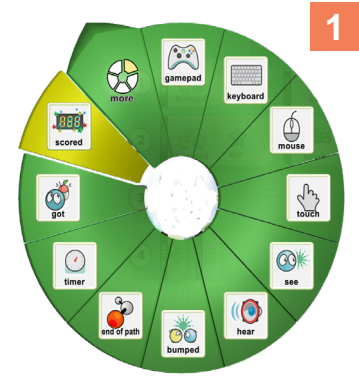
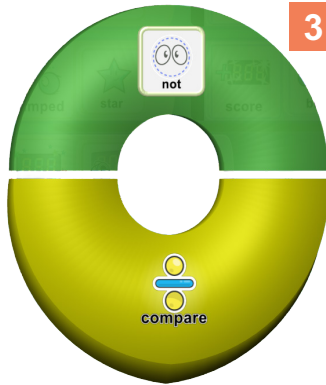
### أولاً: برمجة الكائن للفوز

تعتمد العديد من الألعاب الرقمية على مهارة جمع النقاط حيث يجمع اللاعب عناصر معينة وأداء مهام للوصول لنهاية المرحلة في اللعبة والشعور بفرحة الانتصار عند نهاية اللعبة .

# برمجة الفوز والخسارة في اللعبة

## برمجة (الحدث أو الشرط)

1. اختيار أداة Object Tool من شريط الأدوات.
2. الضغط بالزر الأيمن للفأرة على الكائن Kodu.
3. اختيار الأمر Program من القائمة الفرعية.
4. الضغط على علامة + الظاهرة بجانب مربع When.
5. اختيار scored Tile من الخيارات الظاهرة.
6. الضغط على علامة + الظاهرة بجانب scored Tile وإضافة تفصيل لوحة النتائج الخاصة بالكائن المطلوب، وذلك باختيار blue Tile.
7. الضغط على علامة + الظاهرة بجانب blue Tile وإضافة تفصيل المقارنة المنطقية المطلوبة للحكم على وصول الكائن للنتيجة المطلوبة، وذلك باختيار compare Group ثم اختيار equals Tile.
8. الضغط على علامة + الظاهرة بجانب equals Tile وإضافة تفصيل القيمة المطلوب الوصول إليها، وذلك باختيار points Group ثم اختيار 20points Tile.



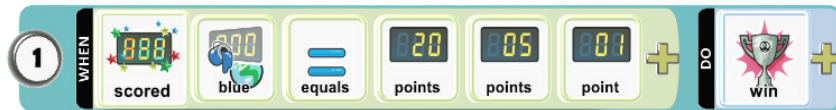


- يوفر البرنامج خيارات مقارنة منطقية متعددة تعلمتها في السابق في مادة الرياضيات يمكنك توظيفها حسب احتياجات اللعبة.
- في حال رغبتك بعدد نقاط غير متاح ضمن الخيارات الجاهزة، تُضاف الأرقام بشكل متتالٍ للحصول على المجموع النهائي للعدد المطلوب.



### برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار game Group من الخيارات الظاهرة.
3. اختيار win Tile من الخيارات الظاهرة.



## ثانياً: برمجة كائن لتتبع كائن آخر

تساعد عملية تتبع الكائن لكائن آخر في اللعبة على إضافة عنصر التشويق أثناء اللعب، حيث تضيف متعة وتحدي للاعب للوصول إلى الفوز وتفادي الخسارة أثناء اللعب.

# برمجة الفوز والخسارة في اللعبة

## برمجة (الحدث أو الشرط)

1. اختيار أداة Object Tool من شريط الأدوات.
2. الضغط بالزر الأيمن للفأرة على الكائن Puck.
3. اختيار الأمر Program من القائمة الفرعية.
4. الضغط على علامة + الظاهرة بجانب مربع When.
5. اختيار see Tile من الخيارات الظاهرة.
6. الضغط على علامة + الظاهرة بجانب see Tile وإضافة تفصيل الكائن الذي يجب رؤيته، وذلك باختيار Kudo Tile.
7. الضغط على علامة + الظاهرة بجانب Kudo Tile وإضافة تفصيل مدى الرؤية ليكون عن قرب، وذلك باختيار Options Group ثم اختيار close by Tile.



يمكنك الاستغناء عن تحديد مدى الرؤية، ولكن ذلك سيُسَهِّل الوصول للكائن بشكل سريع مما يُفقد اللعبة عنصر الصعوبة. يمنحك البرنامج خيارات متعددة لمجال الرؤية ليعطيك مجال لإطلاق العنان لخيالك في تصميم اللعبة وفي رفع مدى صعوبة تجاوز التحديات



1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار move Tile من الخيارات الظاهرة.
3. الضغط على علامة + الظاهرة بجانب move Tile لإضافة تفصيل اتجاه حركة الكائن Puck نحو الكائن Kudo عند تشغيل اللعبة، وذلك باختيار toward Tile.
4. الضغط على علامة + الظاهرة بجانب toward Tile وإضافة تفصيل سرعة التتبع، وذلك باختيار slowly Tile.



خيارات الاتجاه متنوعة (مثل : نحو الكائن، بعيداً عن الكائن، تجنب الكائن ...) تساعدك في بناء تحديات مختلفة للعبة أكثر تشويقاً.



## ثالثاً: برمجة الكائن للخسارة

غالباً ما تحفز رسالة الخسارة في اللعبة على إعادة المحاولة للوصول لتحقيق الفوز، وتحقيق نتائج أفضل في المحاولات القادمة  
استكمل برمجة الكائن Puck للاصطدام بالكائن Kodu ليخسر اللعبة.

### برمجة (الحدث أو الشرط)

- في سطر برمجي جديد يتم الضغط على علامة + الظاهرة بجانب مربع When.
- اختيار bumped Tile من الخيارات الظاهرة.
- الضغط على علامة + الظاهرة بجانب bumped Tile وإضافة تفصيل الكائن الذي يجب الاصطدام به، وذلك باختيار Kodu Tile.

## برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار game Group من الخيارات الظاهرة.
3. اختيار end Tile من الخيارات الظاهرة.



## رابعاً: إعدادات متقدمة للعبة أكثر تشويقاً

### إعدادات الكائن

يساعد التحكم في إعدادات الكائن على تصميم حركة وسلوك ديناميكي يضيف واقعية وتفاعل أكبر داخل اللعبة، وهي خطوة مهمة للعبة أكثر احترافية.



■ للتحكم في سرعة وتسارع الكائن أثناء تحركه داخل اللعبة:

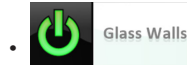
- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Puck.
- اختيار الأمر Change Settings من القائمة الفرعية.
- تظهر خيارات متعددة سنتعرف في هذا الدرس على اثنين منها:

العنصر	الوظيفة	تأثير القيم
	تحديد السرعة النهائية لحركة الكائن للأمام (التحكم في السرعة النهائية للكائن).	<ul style="list-style-type: none"> <li>قيمة صغيرة تؤدي إلى حركة بطيئة جداً.</li> <li>قيمة كبيرة تؤدي إلى حركة أسرع.</li> </ul>
	التحكم في سرعة زيادة الحركة عند بدء التحرك (التحكم في كيفية الوصول لهذه السرعة - ببطء أم بسرعة).	<ul style="list-style-type: none"> <li>قيمة صغيرة تؤدي إلى تسارع بطيء وتدرجي.</li> <li>قيمة كبيرة تؤدي إلى انطلاق سريع نحو السرعة القصوى.</li> </ul>

## إعدادات العالم

لإضافة تنوع وتحدٍ أكبر على تصميم اللعبة يمكن التحكم في كيفية تفاعل الكائنات مع الجدران داخل اللعبة، هل تكون حاجزاً صلباً يمنع الوقوع خارج العالم أم جدران شفافة يمكن العبور من خلالها وخسارة اللعبة.

■ اختيار أداة Change World Settings Tool من شريط الأدوات.



■ الذهاب إلى خيار Glass Walls

العنصر	الوظيفة	تأثير القيم
 مفعّل	يُحوّل الجدران إلى حواجز شفافة بالنسبة للكائن.	<ul style="list-style-type: none"> <li>يمكن للكائن المرور عبر الجدران دون عائق.</li> </ul>
 غير مفعّل	يُحوّل الجدران إلى جدران صلبة وطبيعية.	<ul style="list-style-type: none"> <li>تمنع الجدران الكائن من المرور وتعمل كحاجز.</li> </ul>

## النشاط



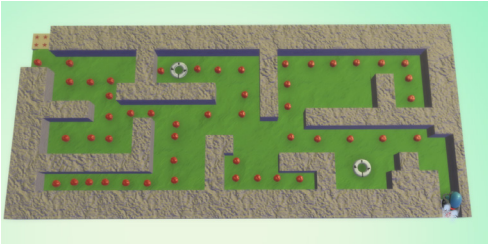
ساعد Cycle على جمع التفاح والحصول على 25 نقطة للفوز باللعبة، ابتعد عن الكائن Puck الذي سيلاحقك و يتتبع حركتك لتفادي الخسارة.

### 1. بدء المغامرة.

■ تشغيل برنامج Kodu Game Lab واختيار الأمر .IMPORT FROM DESKTOP

■ استدعاء مشروع win & lose من مجلد أوراق العمل.

■ الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.



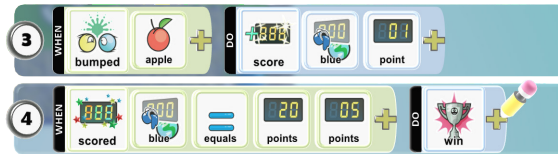
### 2. برمجة الكائن Cycle لجمع التفاح وكسب النقاط.

■ تحديد أداة Object Tool من شريط الأدوات.

■ تحديد الكائن Cycle والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.

■ في سطر برمجي جديد إضافة اللازم لبرمجة الكائن Cycle عند حدوث اصطدام بالكائن Apple يتم احتساب نقطة واحدة لكل تفاحة، تُضاف إلى لوحة النتيجة النهائية الخاصة بالكائن.

■ في سطر برمجي جديد إضافة اللازم لتتبع الكائن تحصيل النقاط وعند الوصول إلى 25 نقطة تظهر له رسالة الفوز باللعبة.



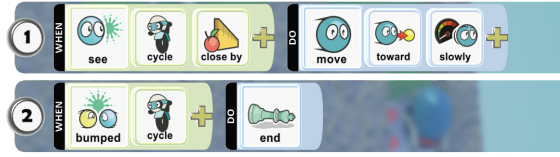
### 3. برمجة الكائن Puck لتتبع اللاعب.

■ تحديد أداة Object Tool من شريط الأدوات.

■ تحديد الكائن Puck والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.

■ في سطر برمجي جديد إضافة اللازم لبرمجة الكائن Puck لتتبع الكائن Cycle عند رؤيته قريباً منه، بحيث يتجه نحوه ببطء.

- في سطر برمجي جديد إضافة اللازم لبرمجة الكائن Puck ليصطدم بالكائن Cycle وعند حدوث الاصطدام يخسر اللعبة.



#### 4. إعدادات ذكية للعب.

- إعدادات العالم:
- إلغاء تفعيل خيار Glass Walls من خيارات العالم لرفع صعوبة اللعبة وجعلها أكثر حماساً.
- إعدادات الكائن:
- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Puck.
- اختيار الأمر Change Settings من القائمة الفرعية.
- تغيير إعداد Forward Speed Multiplier ليصبح 0.1
- تغيير إعداد Forward Acceleration Multiplier ليصبح 0.1

#### 5. اختبار العالم.

- تشغيل اللعبة وجمع التفاح في محاولة للفوز باللعبة.
- تعديل المقطع البرمجي عند الحاجة.

#### 6. توثيق الإنجاز.

- حفظ اللعبة باسم المتعلم متبوعاً بكلمة Apples.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقاً.



## التطبيق



### ورقة عمل 17 : سباق الكنز

تخيل أنك مصمم ألعاب محترف، مهمتك اليوم هي مساعدة Cycle للبحث عن القطع النقدية المبعثرة حول العالم، للفوز بالمهمة عليك أن تجمع 50 نقطة حيث يحصل اللاعب على 5 نقاط لكل قطعة نقدية، انتبه من التحديات حيث أن الكائنات Wisp و Octopus و Fish يسعون لخسارتك.

#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر IMPORT FROM DESKTOP.
- استدع مشروع Quest Menia من مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. الكائن Cycle.

- أضف السطر البرمجي اللازم للكائن لمقارنة مجموع النقاط ( النتيجة )، في حال الحصول على 50 نقطة يُعلن فوز اللاعب.
- أضف سطر برمجي آخر للكائن لمقارنة مجموع النقاط ( النتيجة )، في حال الحصول على أقل من صفر تُنهي اللعبة بالخسارة.

#### 3. الكائن Octopus.

- أضف السطر البرمجي اللازم للكائن ليتتبع الكائن Cycle عند رؤيته قريباً منه، بحيث يتجه نحوه ببطء.
- أضف سطر برمجي آخر للكائن لينقص نقطة واحدة من النتيجة النهائية في كل مرة يصطدم بها بالكائن Cycle.
- اضبط إعدادات السرعة والتسارع الخاصة بالكائن لتصبح 0.3.

#### 4. اختبار العالم.

- شغل اللعبة واحرص على جمع القطع النقدية للفوز باللعبة وعدم خسارة أي من النقاط.
- عدل المقطع البرمجي عند الحاجة.

#### 5. حفظ وتصدير اللعبة.

- احفظ العالم باسمك متبوعاً بكلمة Coins.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أميز بين كائن يمنح نقاطاً وكائن يعد عائقاً.

أستخدم عداد النقاط لاحتساب النتائج بدقة أثناء اللعب.

أجعل الكائنات تختفي عند الحاجة لرفع مستوى التحدي.

أضبط إعدادات الكائنات بما يخدم هدف اللعبة.

أطبق مهارات التفكير المنطقي في تصميم لعبة جمع النقاط وتخطيط خطواتها.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

# برمجة الأهداف والتتبع الذكي

### نواتج التعلم

- برمجة الأهداف داخل اللعبة.
- تحديد اتجاه عنصر الإطلاق لتحقيق تتبع ذكي ودقيق للهدف.
- إضافة النقاط عند إصابة الهدف لزيادة التحدي والحماس في اللعبة.
- استخدام المحاذاة Indentation لربط أكثر من استجابة بنفس الحدث دون تكرار البرمجة.
- التحكم في إعدادات صحة الكائن وعدد الضربات اللازمة لتدميره.
- تفعيل خيار إعداد الكائن ليكون غير قابل للتدمير عند الحاجة في تصميم اللعبة.
- تطبيق مهارات التفكير المنطقي في برمجة الأهداف وتنظيم قواعد اللعبة.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ كيف يساعد تنظيم القواعد البرمجية بأن يجعل اللعبة أكثر ذكاءً ووضوحاً؟

■ ما الفائدة من تحديد اتجاه الهدف؟

■ ماذا سيضيف اللاعب الثاني إلى اللعبة؟

## التعلم



تعرفنا سابقاً على كيفية احتساب النقاط، وسنتعلم اليوم على برمجة الأهداف وتحديد اتجاه حركتها، والتحكم بعدد الضربات لتدمير الكائنات، وتنظيم القواعد البرمجية للعبة أكثر وضوحاً وذكاءً.

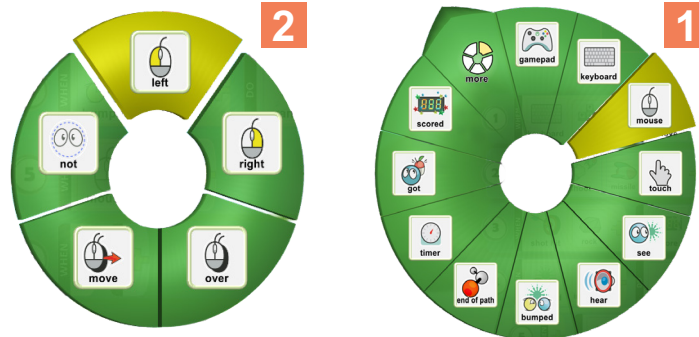
## أولاً: برمجة الأهداف

تعتمد الكثير من الألعاب الرقمية على وجود هدف واضح يتحرك نحوه اللاعب، ومن خلال برمجة الأهداف تمنح اللاعب سلوكاً ذكياً مما يزيد من عامل التشويق والحماس في اللعبة.

إزالة العوائق الظاهرة في اللعبة من خلال استهدافها:

### ■ برمجة (الحدث أو الشرط)

1. اختيار أداة Object Tool من شريط الأدوات.
2. الضغط بالزر الأيمن للفأرة على الكائن Kodu.
3. اختيار الأمر Program من القائمة الفرعية.
4. الضغط على علامة + الظاهرة بجانب مربع When.
5. اختيار mouse Tile من الخيارات الظاهرة.
6. الضغط على علامة + الظاهرة بجانب mouse Tile وإضافة تفصيل الضغط على الزر الأيسر للفأرة، وذلك باختيار left Tile.



### ■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار shoot Tile من الخيارات الظاهرة.
3. الضغط على علامة + الظاهرة بجانب shoot Tile وإضافة تفصيل الأداة المستخدمة لإصابة الهدف، وذلك باختيار missile Tile.
4. الضغط على علامة + الظاهرة بجانب missile Tile وإضافة تفصيل أنه مع كل ضغطة يتم الإطلاق مرة واحدة فقط، وذلك باختيار once Tile.



## ثانياً: التتبع الذكي

في حال عدم تحديد اتجاه الإطلاق، سيتحرك العنصر المقذوف بشكل عشوائي نحو مؤشر الفأرة وقد لا يصل للهدف بدقة. لذلك سنتعلم كيفية ضبطه بطريقة تجعله يتحرك بدقة ويتتبع الهدف بشكل أفضل

استكمال السطر البرمجي السابق:

■ برمجة الاستجابة

■ الضغط على علامة + الظاهرة بجانب once Tile وإضافة تفصيل اتجاه العنصر المقذوف للأمام، وذلك باختيار direction Group ثم اختيار forward Tile.



# برمجة الأهداف والتتبع الذكي

يوفر البرنامج خيارات اتجاه متعددة، وهي:

ماذا يعني؟	الخيار
يتحرك العنصر المقذوف بخط مستقيم دون صعود أو هبوط ودون التأثير بتضاريس الأرض.	Level
يتبع العنصر المقذوف تضاريس الأرض ويعلو فوق المرتفعات ويحاول تجاوز العوائق.	Cruise
تحديد اتجاه حركة العنصر المقذوف: للأمام - للأعلى - للأسفل - اتجاهات العالم الأساسية.	Direction

## ثالثاً: إضافة النقاط عند إصابة الهدف

تزداد اللعبة حماساً عندما يحصل اللاعب على نقاط عند إصابة الأهداف، حيث تشجعه هذه النقاط على الاستمرار وتحقيق نتائج أفضل

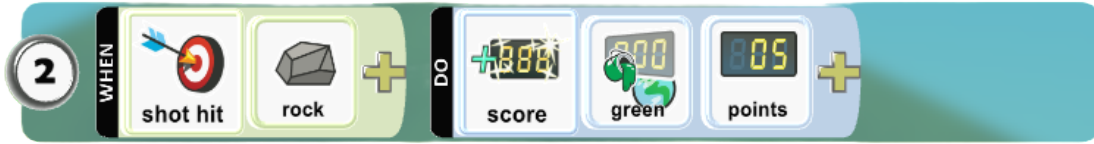
■ برمجة (الحدث أو الشرط)

1. الانتقال إلى سطر برمجي جديد للكائن.
2. الضغط على علامة + الظاهرة بجانب مربع When.
3. اختيار more Group من الخيارات الظاهرة.
4. اختيار shot hit Tile من الخيارات الظاهرة.
5. الضغط على علامة + الظاهرة بجانب shot hit Tile وإضافة تفصيل الكائن المحدد كهدف للإصابة، وذلك باختيار object Group ثم اختيار more Group ثم اختيار الكائن المطلوب rock Tile.



## ■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار game Group من الخيارات الظاهرة.
3. اختيار score Tile + من الخيارات الظاهرة.
4. الضغط على علامة + الظاهرة بجانب score Tile + وإضافة تفصيل لون لوحة النتائج الخاصة بالكائن، وذلك باختيار green Tile.
5. الضغط على علامة + الظاهرة بجانب green Tile وإضافة تفصيل الحصول على 5 نقاط عن كل إصابة، وذلك باختيار points Group ثم اختيار 5 points.



## رابعاً: تنسيق الأوامر البرمجية باستخدام المحاذاة (Indentation)

تساعد المحاذاة في تنظيم الأوامر البرمجية لتكون واضحة وسهلة الفهم، ومن خلالها يمكن تمييز الأوامر التابعة لبعضها لتظهر بنية البرنامج بشكل مرتب وذكي في حال وجود أكثر من استجابة مرتبطة بنفس الحدث بدلاً من تكرار كتابة الحدث يتم كتابته مرة واحدة فقط ثم عمل محاذاة لبقية الأحداث فتكون تابعاً لذلك الحدث كالتالي:

## ■ برمجة (الحدث أو الشرط)

- الانتقال إلى سطر برمجي جديد للكائن.
- الضغط على علامة + الظاهرة بجانب مربع When.
- اختيار bumped Tile من الخيارات الظاهرة.
- الضغط على علامة + الظاهرة بجانب bumped Tile وإضافة تفصيل الكائن المحدد للاستخدام به، وذلك باختيار object Group ثم اختيار الكائن المطلوب ink cloud Tile.

# برمجة الأهداف والتتبع الذكي

## ■ برمجة الاستجابة

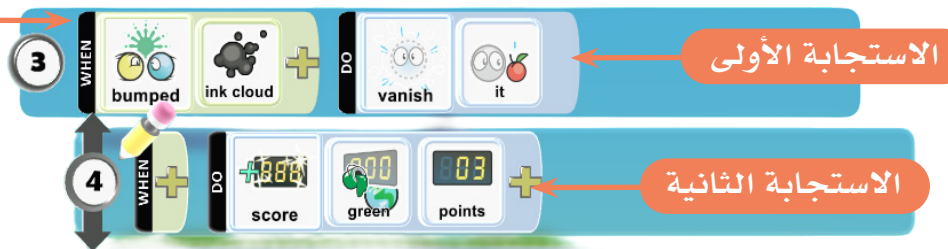
- الضغط على علامة + الظاهرة بجانب مربع Do.
- اختيار combat Group من الخيارات الظاهرة.
- اختيار vanish Tile من الخيارات الظاهرة.
- الضغط على علامة + الظاهرة بجانب vanish Tile وإضافة تفصيل اختفاء الكائن الذي يتم الاصطدام به ، وذلك باختيار it Tile.



## ■ لإضافة الاستجابة الثانية والمرتبطة بنفس الحدث الأول وهو الاصطدام بكائن Ink Cloud :

- الانتقال للسطر البرمجي التالي والضغط بالزر الأيسر للفأرة على رقم السطر البرمجي لإظهار المقابض حول رقم السطر ثم باستخدام السحب والإفلات اسحب السطر لجهة اليمين حتى يتم إزاحته ويصبح تابع للحدث الرئيسي.
- الضغط على علامة + الظاهر بجانب مربع Do.
- اختيار score Tile + من الخيارات الظاهرة.
- الضغط على علامة + الظاهر بجانب score Tile وإضافة تفصيل لون لوحة النتائج الخاصة بالكائن، وذلك باختيار green Tile.
- الضغط على علامة + الظاهر بجانب green Tile وإضافة تفصيل الحصول على 3 نقاط عن كل اصطدام، وذلك باختيار points Group ثم اختيار 3 points.

الحدث أو الشريط الرئيسي



## خامساً: إعدادات الكائن

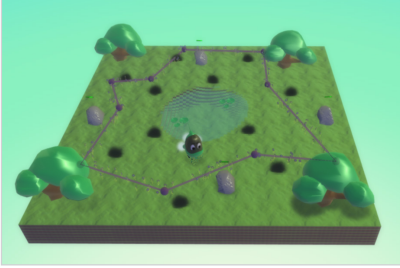
لإضافة تنوع وتحدي أكبر على تصميم اللعبة، يمكن التحكم في إعدادات صحة الكائن وقوته إما بجعله منيع لا يمكن استهدافه أو بالتحكم في الحد الأقصى لعدد الإصابات التي يمكنه تلقيها، وذلك كالتالي:

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Rock.
- اختيار الأمر Change Settings من القائمة الفرعية.

تظهر خيارات عديدة، سنتعرف على بعضها وهي:

متى يُستخدم؟	الوظيفة	الخيار
عند الرغبة بأن يرى اللاعب قوة الكائن وتأثره بالضربات	عرض شريط صحة الكائن على الشاشة ليعرف اللاعب كمية الصحة المتبقية	Show Hit Points
عند ضبط قوة الكائن ليحتاج عدة ضربات لتدميره	تحديد عدد النقاط التي يمتلكها الكائن قبل أن يتم تدميره	Max Hit Points
عند الرغبة بأن يكون الكائن قوي جداً أو هدف تدريبي	جعل الكائن لا يتأثر بالضربات ولا يفقد الصحة	Invulnerable
عند ضبط قوة الهجوم بحيث تصبح الضربة أقوى أو أضعف	تحديد مقدار الضرر الذي يسببه المقذوف للكائن عند الإصابة	Missile Damage

## النشاط



ساعد Kodu في المحافظة على نظافة الحديقة حيث تجمعت السحب الملوثة فيها، Kodu يحصل على ثلاث نقاط عند إخفاء كل غيمة، ويحصل على خمس نقاط في كل مره يتخلص فيها من الصخور التي تعرقل الطريق من خلال إصابتها، يكافأ اللاعب بالفوز عند حصوله على 25 نقطة.

### 1. بدء المغامرة

- تشغيل برنامج Kodu Game Lab واختيار الأمر IMPORT FROM DESKTOP.
- استدعاء مشروع Clean The Garden من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الامر Edit.

### 2. برمجة الكائن kodu

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن Kodu والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم ليتم إطلاق قذيفة نحو الهدف مرة واحدة، وذلك في كل مرة يتم فيها الضغط على زر الفأرة الأيسر.
- إضافة السطر البرمجي اللازم للكائن Kodu ليحصل على 5 نقاط في لوح النتائج الأخضر، وذلك عن كل إصابة يتلقاها الكائن Rock.
- إضافة السطر البرمجي اللازم ليتم إخفاء الكائن ink Cloud عند الاصطدام به.
- إضافة الاستجابة الثانية لشرط الاصطدام للكائن Kodu من خلال استخدام الإزاحة indenting ليكسب الكائن 3 نقاط عند الاصطدام بالكائن ink Cloud.



### 3. إعدادات الكائن

- تحديد أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن.
- اختيار الأمر Change Settings من القائمة الفرعية.
- الكائن Tree
- ضبط الكائن ليكون غير قابل للتدمير من خلال تفعيل الخيار Invulnerable.
- الكائن Rock
- عرض شريط صحة الكائن على الشاشة من خلال تفعيل خيار Show Hit Points.
- تحديد الحد الأقصى للنقاط ليكون 10 والتي يمتلكها الكائن قبل أن يتم تدميره من خلال خيار Max Hit Points.
- الكائن Kodu
- تحديد مقدار الضرر الذي تسببه قذيفة الكائن عند الإصابة لتكون 5 من خلال خيار Missile Damage.

### 4. اختبار العالم

- تشغيل اللعبة وتنظيف الحديقة من خلال إخفاء الغيوم السوداء وتدمير الصخور التي شوهدت الحديقة في محاولة للفوز باللعبة.
- تعديل المقطع البرمجي عند الحاجة.

### 5. توثيق الإنجاز

- حفظ اللعبة باسم المتعلم متبوعاً بكلمة Garden.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقاً.



## التطبيق



### ورقة عمل 18: جمع النقود

تخيل أنك مصمم ألعاب محترف، ومهمتك اليوم هي مساعدة Kodu للبحث عن القطع النقدية المبعثرة على الطريق، للفوز بالمهمة يجب على Kodu أن يجمع أكثر من 30 نقطة، حيث سيحصل على 5 نقاط لكل قطعة نقدية، و نقطة واحدة إضافية في كل مره يحطم الصخرة لإزاحتها عن الطريق انتبه من التحديات حيث أن الكائن Rock يعوق حركة الكائن Kodu.



#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر IMPORT .FROM DESKTOP
- استدع مشروع Coin Collection مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. برمجة الكائن Kodu

- أضف السطر البرمجي اللازم ليطلق الكائن قذيفة نحو الهدف مرة واحدة وذلك في كل مرة يتم فيها الضغط على زر الفأرة الأيسر.
- أضف السطر البرمجي اللازم ليحصل الكائن على نقطة واحدة في لوح النتائج الأحمر، وذلك عن كل إصابة يتلقاها الكائن Rock.
- أضف السطر البرمجي اللازم ليتم إخفاء الكائن Coin عند الاصطدام به.
- أضف الاستجابة الثانية لشرط اصطدام الكائن Kodu بالكائن Coin من خلال استخدام الإزاحة Indenting ليكسب الكائن خمس نقاط.

### 3. إعدادات كائنات اللعبة

- الكائن Tree: عدم قابلية الكائن للتدمير من خلال تفعيل خيار Invulnerable.
- الكائن Rock:
- عرض شريط صحة الكائن على الشاشة.
- تحديد عدد 10 نقاط يمتلكها الكائن قبل أن يتم تدميره.
- الكائن Kodu: تحديد مقدار الضرر الذي تسببه قذيفة الكائن عند الإصابة لتكون 5.

### 4. اختبار العالم

- شغل اللعبة واحرص على جمع القطع النقدية للفوز باللعبة وإزالة العوائق عن الطريق لتحقيق الهدف.
- عدّل المقطع البرمجي عند الحاجة.

### 5. حفظ وتصدير اللعبة

- احفظ العالم باسمك متبوعاً بكلمة Coins.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أبرمج عنصر الإطلاق لاستهداف الأهداف داخل اللعبة.

أحدد اتجاه عنصر الإطلاق ليتتبع الهدف بدقة.

أضيف النقاط عند إصابة الأهداف لزيادة التحدي.

أستخدم المحاذاة لتنظيم الأوامر البرمجية وربطها بالحدث المناسب.

أتحكم في صحة الكائنات وعدد الضربات المطلوبة لتدميرها.

أطبق مهارات التفكير المنطقي في تصميم منطق اللعبة.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

# الحوارات التفاعلية ورفع مستوى اللعبة

### نواتج التعلم

- توظيف الحوارات التفاعلية بين الكائنات لزيادة الواقعية والتشويق داخل اللعبة.
- التعبير عن مشاعر الكائنات باستخدام الرموز التعبيرية المناسبة لرفع مستوى التفاعل البصري.
- استخدام السمع والرؤية كأحداث محفزة لاستجابات الكائنات داخل اللعبة.
- توليد الكائنات أثناء اللعب باستخدام Creatable لتعزيز حيوية اللعبة وتفاعلها.
- التحكم في التوقيت الزمني للتوليد باستخدام المدد الثابتة والعشوائية لزيادة التحدي.
- توظيف إعدادات التخفي لرفع مستوى الصعوبة داخل اللعبة.
- ربط الأحداث بنتائج اللعب مثل فتح الممرات المشروطة بعد تحقيق عدد معين من النقاط.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ ماذا تضيف الحوارات التفاعلية إلى اللعبة؟

■ ما فائدة ظهور أحداث عشوائية داخل اللعبة بدلاً من أحداث ثابتة؟

■ كيف يمكن لإعدادات الكائن أن تجعل اللعبة أكثر تشويقاً أو أكثر صعوبة؟

## التعلم



تعرفنا سابقاً على كيفية برمجة الأهداف، وسنتعلم اليوم على كيفية إنشاء حوارات تفاعلية بين الكائنات، واستخدام الرؤية والسمع والتحكم بالوقت والعشوائية، إضافة إلى إعدادات متقدمة ترفع مستوى اللعبة وتزيد من تحدياتها.

### أولاً: الحوارات التفاعلية والتعبير عن المشاعر داخل اللعبة

تعتمد الكثير من الألعاب على التفاعل بين الشخصيات لزيادة الواقعية والتشويق، ومن خلال استخدام Say Tile و Express Tile يمكن للكائنات التحدث والتعبير عن المشاعر أثناء اللعب.

# الحوارات التفاعلية ورفع مستوى اللعبة

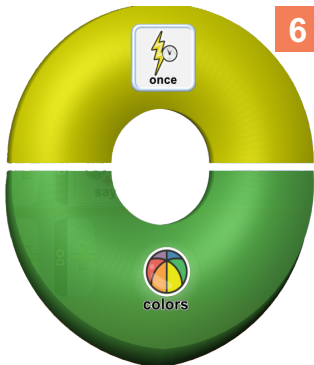
برمجة الكائن لبدء حوار عند سماعه لكائن آخر، كالتالي:  
 ■ برمجة (الحدث أو الشرط).

1. اختيار أداة Object Tool من شريط الأدوات.
2. الضغط بالزر الأيمن للفأرة على الكائن Cycle.
3. اختيار الأمر Program من القائمة الفرعية.
4. الضغط على علامة + الظاهرة بجانب مربع When.
5. اختيار Tile hear من الخيارات الظاهرة.
6. الضغط على علامة + الظاهرة بجانب Tile hear وإضافة تفصيل الكائن المطلوب سماعه، وذلك باختيار bots1 Group.
7. اختيار Tile saucer.



■ برمجة الاستجابة.

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار actions Group من الخيارات الظاهرة.
3. اختيار Tile say من الخيارات الظاهرة.
4. تظهر نافذة الكتابة، لكتابة النص المطلوب من الكائن إظهاره يتم كتابة الجملة أو الفقرة المطلوبة ثم الضغط على حفظ.
5. الضغط على علامة + الظاهرة بجانب Tile say وإضافة تفصيل ظهور النص مرة واحدة فقط وإلا يظهر النص كلما تحقق الحدث، وذلك باختيار Tile once.





يمكن إضافة لون للنص الظاهر من خلال إضافة Tile الخاصة باللون المطلوب من Colors Group.

توجد عدة خيارات لطريقة ظهور النص أسفل نافذة النص:



### الاستخدام

### الخيار

عرض النص على كامل شاشة اللعبة.

Full Screen

عرض النص في فقاعة مع عرض الأسطر بالترتيب.

Thought Balloon, pick line sequentially

عرض النص في فقاعة مع عرض الأسطر عشوائياً في كل مرة.

Thought Balloon, pick line randomly

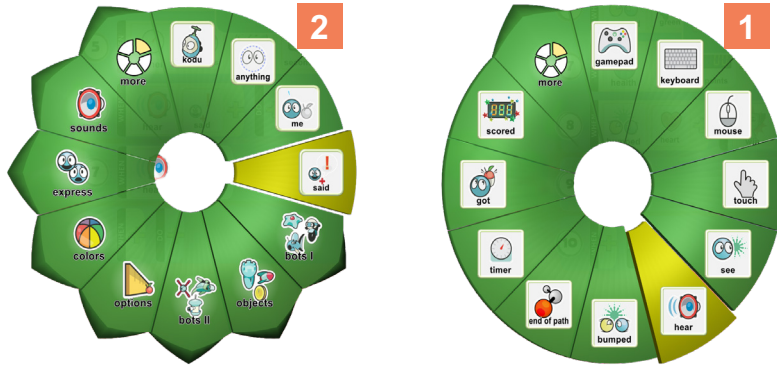
### الرد على الحوار بين الكائنات

عند تحدّث الكائن الأول نتيجة رؤية أو سماع كائن محدّد، والرغبة بتأخير ردّ الكائن الثاني ليأتي لاحقاً لإضفاء طابع واقعي على التفاعل، بحيث يُقدّم الرد بعد مرور زمنٍ قصير على المحادثة :

#### ■ برمجة (الحدث أو الشرط).

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Saucer.
- اختيار الأمر Program من القائمة الفرعية.
- الضغط على علامة + الظاهرة بجانب مربع When.
- اختيار Tile hear من الخيارات الظاهرة.
- الضغط على علامة + الظاهرة بجانب Tile hear وإضافة تفصيل النص الذي تم قوله، وذلك باختيار said Tile.
- تظهر نافذه لكتابة النص المطلوب من الكائن قوله، ويجب الانتباه للرد عليه، وذلك بكتابة الجملة أو الفقرة التي تم قولها من قبل الكائن Cycle حرفياً دون أي تغيير.

# الحوارات التفاعلية ورفع مستوى اللعبة



■ برمجة الاستجابة.

- الضغط على علامة + الظاهرة بجانب مربع Do.
- اختيار actions Group من الخيارات الظاهرة.
- اختيار say Tile من الخيارات الظاهرة.
- تظهر نافذة الكتابة، لكتابة النص المطلوب من الكائن إظهاره ( للرد على حوار الكائن Cycle)، بعد كتابة النص يتم الضغط على حفظ.
- الضغط على علامة + الظاهرة بجانب say Tile وإضافة تفصيل ظهور النص مرة واحدة فقط، وإلا سيظهر النص كلما تحقق الحدث، وذلك باختيار Tile once.



خيارات طريقة ظهور النص أسفل نافذة النص:



## نوع التفاعل

## الخيار

يستجيب الكائن مباشرة فور سماع الصوت بدون انتظار أي شيء آخر.

Match as soon as I hear

ينتظر الكائن اكتمال الصوت ثم يستجيب بعده.

Match after I've heard

## التعبير عن المشاعر

يمكن إضافة تفاعلاً بصرياً وحيوياً داخل اللعبة من خلال إظهار الكائن لحالات تعبيرية تمثل مشاعر مختلفة مثل: السعادة، الخوف، الغضب أو الدهشة تظهر هذه الحالات على شكل رموز تعبيرية (مثل: وجوه أو قلوب أو علامات) .  
ليكن الحدث هو رؤية أو سماع كائن محدد، عندها ستكون:

■ برمجة الاستجابة.

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار actions Group من الخيارات الظاهرة.
3. اختيار Tile express من الخيارات الظاهرة.
4. الضغط على علامة + الظاهرة بجانب express Tile وإضافة تفصيل الحالة التعبيرية المطلوبة مثل السعادة، وذلك باختيار happy Tile.
5. الضغط على علامة + الظاهرة بجانب happy Tile وإضافة تفصيل رمز السعادة المطلوب إظهاره مثل النجوم، وذلك باختيار stars Tile.



تختلف الرموز التعبيرية التي ستظهر لك في الخيارات حسب نوع المشاعر التي أخترت إظهارها.

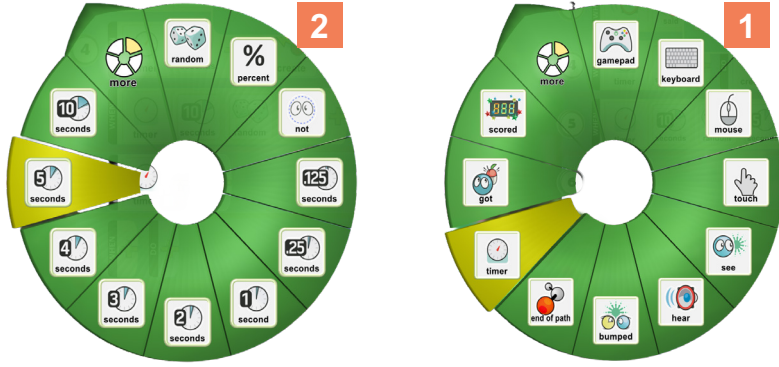


## ثانياً: توليد الكائنات داخل اللعبة

تتيح خاصية Creatable توليد كائنات جديدة أثناء اللعب بشكل تلقائي أو عند وقوع حدث معين، مما يجعل اللعبة أكثر حيوية، ويمنح اللاعب تجربة تفاعلية.

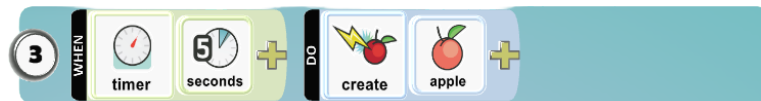
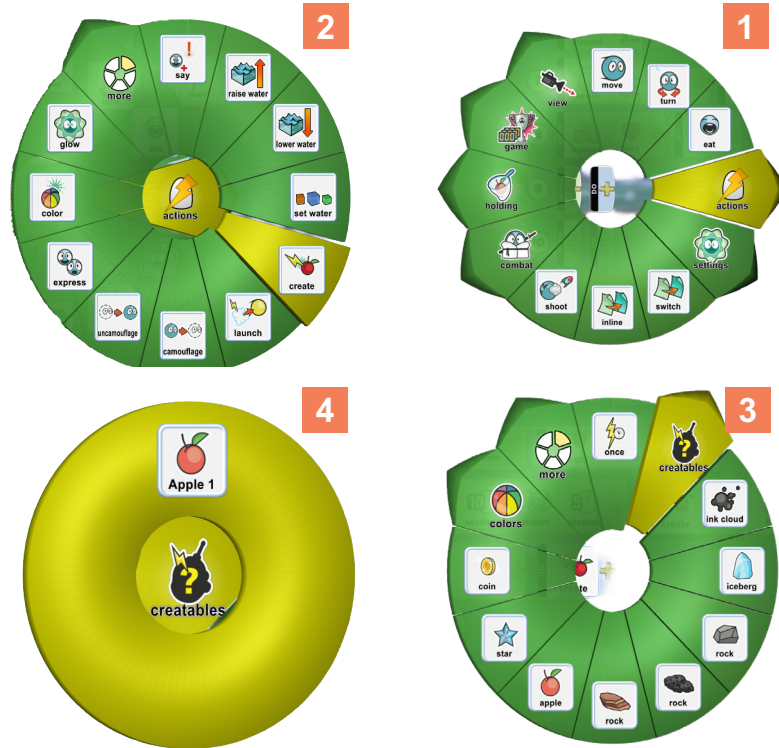
### خطوات توليد كائنات جديدة:

- **أولاً:** تحديد الكائن المطلوب توليده أثناء اللعب، وكذلك تحديد الكائن الذي سينفذ عملية التوليد بعد الاستجابة لحدث معين.
- **ثانياً:** ضبط إعدادات الكائن ليكون قابلاً للتوليد، من خلال تفعيل خاصية Creatable، عند تفعيل هذه الخاصية سيختفي عند تشغيل اللعبة وسيبقى ظاهراً فقط أثناء وضع تحرير اللعبة.
- **ثالثاً:** برمجة الكائن لتنفيذ عملية توليد الكائن المطلوب عند الاستجابة لحدث معين مثل مرور مدة زمنية.
- برمجة (الحدث أو الشرط).
  1. اختيار أداة Object Tool من شريط الأدوات.
  2. الضغط بالزر الأيمن للفأرة على الكائن Saucer.
  3. اختيار الأمر Program من القائمة الفرعية.
  4. الضغط على علامة + الظاهرة بجانب مربع When.
  5. اختيار timer Tile من الخيارات الظاهرة.
  6. الضغط على علامة + الظاهرة بجانب timer Tile وإضافة تفصيل المدة الزمنية التي يتم توليد الكائن بعدها، وذلك باختيار 5 seconds Tile.



## ■ برمجة الاستجابة.

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار Group actions من الخيارات الظاهرة.
3. اختيار Tile create من الخيارات الظاهرة.
4. الضغط على علامة + الظاهرة بجانب create Tile وإضافة تفصيل الكائن المطلوب توليده، وذلك باختيار creatables Group ستظهر الكائنات التي تم تفعيل خاصية التوليد لها داخل اللعبة.
5. اختيار apple Tile من الخيارات الظاهرة.



## ثالثاً: مهارات التخفي داخل اللعبة

تزداد اللعبة حماساً عند إنشاء تحديات متنوعة داخلها، فمن خلال خصائص بعض الكائنات قد تكون غير مرئية، أو قادرة على المرور عبر الجدران، أو تكون مموّهة يصعب ملاحظتها، مما يرفع مستوى التحدي داخل اللعبة.

### لعرض إعدادات الكائن:

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Rock.
- اختيار الأمر Change Settings من القائمة الفرعية.

تظهر خيارات التخفي داخل إعدادات الكائن، داخل اللعبة وهي:

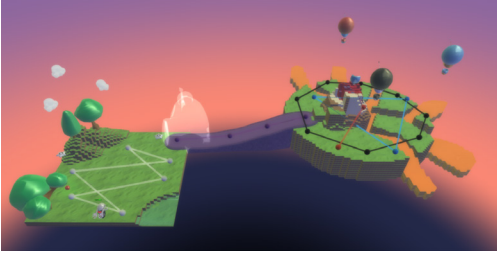
تأثيره أثناء اللعب	الوظيفة	الخيار
يختفي شكل الكائن من الشاشة، لكنه يبقى موجوداً ويمكنه أن يتحرك ويؤثر ويتأثر (حسب البرمجة).	يجعل الكائن غير مرئي بالنسبة للاعب.	Invisible
لا يصطدم بالجدران أو الكائنات الأخرى، يتحرك بحرية عبرها	يجعل الكائن قادر على المرور عبر الأشياء.	Ghost
يصبح الكائن شبه شفاف أو صعب الملاحظة، ولكن يمكن رؤيته بشكل خافت	يجعل الكائن يندمج مع البيئة أو يقلل وضوحه.	Camouflage

■ أمثلة لتحديات يمكنك تنفيذها بالاستعانة بخيارات الإخفاء:

1. باب غير مرئي يُفتح عند تحقق شرط معين، مما يضيف عنصر المفاجأة.
2. كنز مخفي لا يمكن رؤيته (لا يظهر على الشاشة) وعلى اللاعب اكتشاف موقعه من خلال تلميحات أو أحداث.
3. كائن يلاحق اللاعب ويمكنه المرور عبر الجدران، مما يزيد صعوبة الاحتماء داخل اللعبة.
4. فخ مموه ينتظر اللاعب في الطريق ليختبر انتباهه وتركيزه.

الخيارات متعددة ولا حدود لها، فبالاستعانة بخيالك لتصميم لعبة مشوقة. ومن خلال الأدوات والخيارات المقدمة لك في البرنامج أصبح تحويل الخيال إلى واقع سهل جداً

## النشاط



ساعد Cycle في خوض مغامرته عبر جمع التفاح الذي يصنعه Saucer، حيث يحصل اللاعب على نقطة واحدة مقابل التفاح الأحمر بينما يخسر نقطة عند التقاط التفاح الأسود، وعند وصوله إلى 5 نقاط، يُفتح الباب السري المؤدي إلى الجزيرة الثانية ليستكمل رحلته نحو قمة الجبل للحصول على القلب والفوز باللعبة. احذر من البالونات التي تستهدفك حيث إصابة واحدة تعني خسارة اللعبة.

### 1. بدء مغامرتك

- تشغيل برنامج Kodu Game Lab واختيار الأمر IMPORT FROM DESKTOP.
- استدعاء مشروع Islands Rush من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.

### 2. برمجة الكائن Cycle.

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن Cycle والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم للكائن Cycle ليقول: « I love apples can I get 5 please » مرة واحدة فقط، على أن تظهر الرسالة على كامل شاشة اللعبة، وذلك عند سماعه للكائن Saucer.



### 3. برمجة Saucer 1,2.

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن Saucer والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم للتعبير عن مشاعر الفرح عند رؤيته للكائن Cycle وذلك بإظهار النجوم.

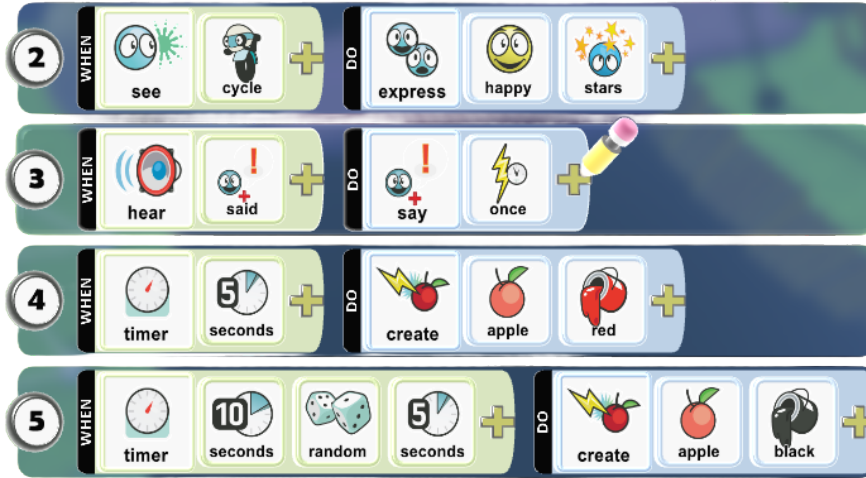
# الحوارات التفاعلية ورفع مستوى اللعبة

■ إضافة السطر البرمجي اللازم للرد على الكائن Cycle بعد سماع جملة:

«I love apples can I get 5 please» ، حيث ينتظر اكتمال الصوت ثم يستجيب بعده بعبارة: «I will share some of my apples with you» على أن تظهر الرسالة مرة واحدة فقط داخل فقاعة وبأسطر مرتبة .

■ إضافة السطر البرمجي اللازم ليؤد الكائن تفاحة حمراء كل 5 ثوان.

■ إضافة السطر البرمجي اللازم ليؤد الكائن تفاحة سوداء بعد مدة عشوائية بين 5 إلى 15 ثانية في كل مرة.



## 4. برمجة الكائن Hut.

■ تحديد أداة Object Tool من شريط الأدوات.

■ تحديد الكائن Hut والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.

■ إضافة السطر البرمجي اللازم لتتبع لوح النتائج ذو اللون الأحمر، عند حصول اللاعب على 5 نقاط يختفي الكائن Hut لإزالة الحاجز بين الجزيرتين.



## 5. إعدادات الكائنات.

■ تحديد أداة Object Tool من شريط الأدوات.

■ الضغط بالزر الأيمن للفأرة على الكائن.

■ اختيار الأمر Change Settings من القائمة الفرعية.

■ الكائن Hut : تفعيل خيار Invisible ليصبح الكائن غير مرئي.

■ الكائن Apple : تفعيل خيار Creatable ليكون الكائن قابلاً للتوليد.

## 6. اختبار العالم.

- تشغيل اللعبة وجمع التفاح الأحمر في محاولة لفتح بوابة العبور للوصول إلى القلب للفوز باللعبة.
- تعديل المقطع البرمجي عند الحاجة.

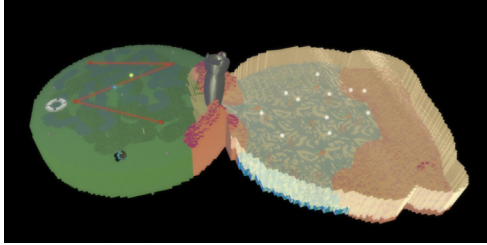
## 7. توثيق الانجاز.

- حفظ اللعبة باسمك متبوعاً بكلمة Rush.
- تصدير اللعبة إلى سطح المكتب حتى تتمكن من مشاركتها لاحقاً.

## التطبيق



### ورقة عمل 19: مهمة القلب الغامض



تخيل أنك مصمم ألعاب محترف، ومهمتك اليوم هي مساعدة Kodu في طلب المساعدة من Puck ليبدأ Puck بإنتاج نجوم متألئة داخل اللعبة. يحصل Kodu على 5 نقاط عند جمع كل نجمة، لذا واصل البحث وجمع أكبر عدد منها. وعندما يصل إلى 30 نقطة تُفتح البوابة المخفية المؤدية إلى الجزيرة الثانية، حيث يكون القلب مخفياً في أعماق البحيرة، وعلى Kodu أن يكون حذراً فهناك أسماك خطيرة تنتظر للهجوم، وأي إصابة تعني خسارة اللعبة.

#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر `.IMPORT FROM DESKTOP`.
- استدع مشروع Gate to the hidden heart مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. برمجة الكائن Kodu.

- إضافة السطر البرمجي اللازم للكائن ليقول:

«I need some stars to get over the gate»

مرة واحدة فقط على أن تظهر الرسالة على كامل شاشة اللعبة، وذلك عند سماعه للكائن Puck.

#### 3. برمجة الكائن Puck.

- إضافة السطر البرمجي اللازم لعبّر الكائن عن مشاعر السعادة عند رؤيته للكائن Kodu وذلك بإظهار القلوب.

- إضافة السطر البرمجي اللازم للرد على الكائن Kodu وذلك بعد سماعه لجملة :

«I need some stars to get over the gate» ، حيث ينتظر الكائن اكتمال الصوت ثم يستجيب بعده بقول العبارة: «Sure I have some stars for you» مرة واحدة فقط على

أن تظهر الرسالة في فقاعة والاسطر مرتبة.

- إضافة السطر البرمجي اللازم للكائن Puck ليقوم بتوليد نجمه بلون عشوائي كل 5 ثوان .
- بعد الانتهاء من البرمجة قم بعمل نسخة ثانية من الكائن ولصقها بقرب المسار.

#### 4. عدادات كائنات اللعبة.

- الكائن Castle جعله غير مرئي من خلال تفعيل خيار Invisible.
- الكائن Star جعله كائن قابل للتوليد من خلال تفعيل خيار Creatable.

#### 5. اختبر العالم.

- شغل اللعبة واجمع النجوم اللازمة لفتح بوابة العبور للوصول إلى القلب للفوز باللعبة.
- عدل المقطع البرمجي عند الحاجة.

#### 6. حفظ وتصدير اللعبة.

- احفظ العالم باسمك متبوعاً بكلمة Gate.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أستخدم الحوارات التفاعلية بين الكائنات لزيادة الواقعية والتشويق داخل اللعبة.

أفعل التعبير عن مشاعر الكائنات باستخدام الرموز التعبيرية المناسبة لرفع التفاعل البصري.

أوظف السمع والرؤية كأحداث محفزة لاستجابة الكائنات داخل اللعبة.

أبرمج توليد الكائنات أثناء اللعب باستخدام Creaable لتعزيز حيوية اللعبة.

أتحكم في توقيت ظهور الكائنات باستخدام المدد الثابتة والعشوائية لزيادة التحدي.

أستخدم إعدادات التخفي لرفع مستوى الصعوبة وإضافة عنصر المفاجأة داخل اللعبة.

أربط الأحداث بنتائج اللعب مثل فتح الممرات المشروطة بعد اكتساب عدد معين من النقاط.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

## منظور الكاميرا والمؤثرات الصوتية

### نواتج التعلم

- التحكم بمنظور الكاميرا داخل اللعبة لرفع جودة العرض والتجربة البصرية.
- اختيار وضع الكاميرا المناسب لطبيعة اللعبة (ثابت - ثابت بإزاحة - حر).
- توظيف الكاميرا في توجيه اللاعب داخل البيئة وربطها بعناصر التحدي.
- إضافة مؤثرات صوتية تفاعلية داخل اللعبة.
- توظيف الموسيقى والبيئات الصوتية لتعزيز التشويق والواقعية.
- ربط الصوت بالأحداث داخل اللعبة باستخدام أوامر WHEN - DO.
- ضبط مستوى الصوت والتحكم بين المؤثرات والموسيقى لخلق توازن سمعي داخل اللعبة.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ ما أهمية زاوية التصوير ومنظور الكاميرا في الألعاب؟

---

---

■ كيف يمكن للصوت أن يجعل اللعبة أكثر واقعية وتشويقاً؟

---

---

■ ما الفرق بين الموسيقى المستمرة والمؤثرات القصيرة داخل اللعبة؟

---

---

## التعلم



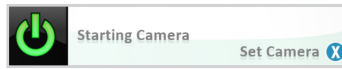
تعرفنا سابقاً على كيفية إضافة تفاعل نصي بين الكائنات، اليوم سنتعلم كيف نجعل لعبتنا أكثر احترافية من خلال التحكم في منظور الكاميرا وإعداداتها، وإضافة المؤثرات الصوتية والموسيقى وربطها بالأحداث داخل اللعبة لتصبح أكثر تشويقاً وتفاعلاً.

## أولاً: التحكم بمنظور الكاميرا داخل اللعبة

تعتمد الألعاب على الكاميرا بشكل كبير في توجيه اللاعب والتحكم بطريقة رؤية العالم، يوفر البرنامج ثلاث وضعيات رئيسية للكاميرا يمكن ضبطها باستخدام أداة Change World Settings من شريط الأدوات:

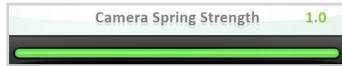
### Starting Camera

■ تحديد زاوية البداية عند تشغيل العالم، أي المنظر الذي يراه اللاعب منذ نقطة البداية.



### Camera Spring Strength

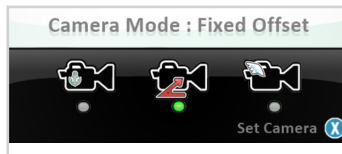
■ تحديد سرعة عودة الكاميرا إلى وضعها الطبيعي بعد تحريكها، قيمة أكبر تعني رجوع أسرع وأكثر ثباتاً.



### Camera Mode

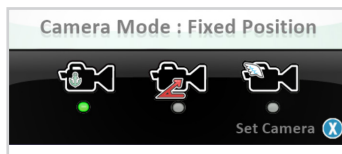
#### 1. وضع الكاميرا الثابتة Fixed Offset

- الكاميرا ثابتة في مكان واحد ولا تتبع اللاعب، مناسبة لعرض المشهد من زاوية ثابتة.
- مناسبة للألعاب ذات المجال المحدود أو المشاهد الثابتة.



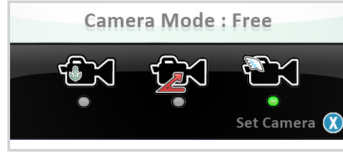
#### 2. وضع الكاميرا بإزاحة ثابتة Fixed Position

- الكاميرا تتبع اللاعب من زاوية ثابتة نسبياً، كأنها ملتصقة به من الخلف أو الأعلى.
- مناسبة للألعاب التي تتطلب متابعة اللاعب من زاوية محددة.



### 3. وضع الكاميرا الحرة Free

- الكاميرا حرة الحركة، يمكن للاعب تدويرها وتحريكها بحرية حول العالم.
- مناسبة للألعاب المفتوحة ثلاثية الأبعاد والحركة الواسعة.



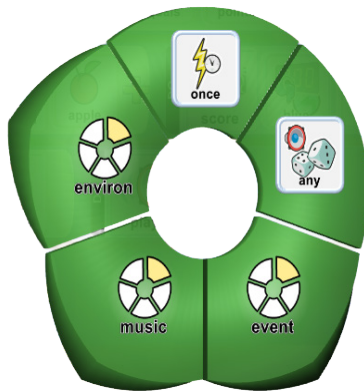
- خيارات الكاميرا بعد التعديل عليها تُمكنك من أن ترى تأثير خياراتك في وضع تشغيل اللعبة.
- عند اختيار أي وضع من أوضاع الكاميرا، تظهر لك علامة **Set Camera X** أسفل الإعداد، وعند الضغط عليها ينتقل البرنامج إلى وضع خاص يسمح لك بتحديد موضع الكاميرا يدوياً وتوجيهها بحرية إلى الزاوية المناسبة، ثم يتم حفظ هذا الموضع ليكون منظور الكاميرا أثناء اللعب، ويمكنك تعديلها لاحقاً بالطريقة نفسها وقتما تشاء.



## ثانياً: المؤثرات الصوتية داخل اللعبة

تعد المؤثرات الصوتية عنصراً مهماً في الألعاب لأنها تضيف حياةً وتشويقاً للمشهد وتساعد اللاعب على التفاعل مع الأحداث، ومن خلال ربط الصوت بالمواقف داخل اللعبة تصبح التجربة أكثر واقعية ومتعة.

يوفر البرنامج مجموعة من المؤثرات الصوتية يمكن تقسيمها إلى:



#### 1. مؤثرات صوتية مرتبطة بحدث event

- مثل: القفز - الانفجار - الفوز - الخسارة.

#### 2. موسيقى خلفية للعبة music

- مثل: موسيقى درامية - حماسية - غامضة.

#### 3. أصوات بيئية environ

- مثل: صوت المحيط - الغابة - المدينة.

# منظور الكاميرا والمؤثرات الصوتية

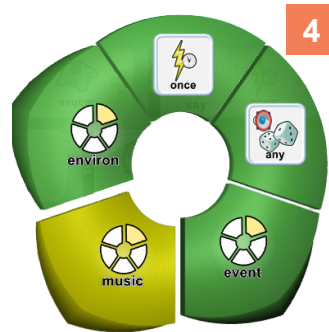
برمجة تشغيل مؤثر صوتي عند حدوث حدث معين

■ برمجة (الحدث أو الشرط)

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Cycle.
- اختيار الأمر Program من القائمة الفرعية.
- الضغط على علامة + الظاهرة بجانب مربع When.
- تحديد الحدث المطلوب مثل always أو bumped أو..... إلخ.

■ برمجة الاستجابة

1. الضغط على علامة + الظاهرة بجانب مربع Do.
2. اختيار actions Group من الخيارات الظاهرة.
3. اختيار more Group من الخيارات الظاهرة.
4. اختيار play Tile من الخيارات الظاهرة.
5. الضغط على علامة + الظاهر بجانب play Tile وإضافة تفصيل نوع المؤثر الصوتي المطلوب.





## التحكم في إعدادات الصوت

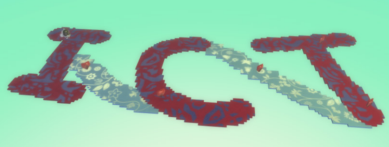
من خلال أداة تغيير إعدادات العالم Change world Settings Tool من شريط الأدوات يمكنك التحكم بإعدادات الصوت للعبة :

الوظيفة	الخيار
التحكم في مستوى صوت المؤثرات (الاصطدام، القفز، الانفجارات، أصوات الأزهار ... إلخ).	
التحكم في مستوى صوت الموسيقى الخلفية للعالم أو المرحلة.	

## النشاط



ساعد Kodu في التعرف على الوضعيات المختلفة لمنظور الكاميرا وإضافة مؤثرات صوتية غامضة تناسب رحلة الاستكشاف.



### 1. بدء المغامرة.

- تشغيل برنامج Kodu Game Lab واختيار الأمر IMPORT FROM DESKTOP.
- استدعاء مشروع ICT Land من مجلد أوراق العمل.
- الضغط بالزر الأيسر للفأرة على المشروع بعد التحميل واختيار الأمر Edit.

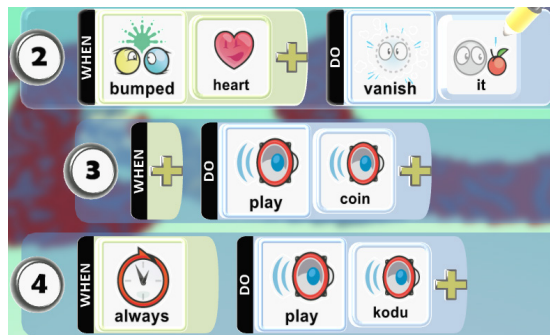
### 2. برمجة الكائن kodu.

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن Kodu والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.

الصوت	النوع	مجموعة المؤثر الصوتي
Coin	Arcade	Event

- استكمال الاستجابة الثانية عند الاصطدام بالكائن Heart وذلك بإضافة السطر البرمجي اللازم لتشغيل صوت (الفوز) :

الصوت	النوع	مجموعة المؤثر الصوتي
Kudo	New	Music



### 3. التحكم في منظور الكاميرا وإعدادات الصوت.

- تحديد أداة Change world Settings Tool من شريط الأدوات.
- تعديل إعداد Music Volume لتصبح 50.
- تعديل إعداد Effects Volume لتصبح 90.
- ضبط وضع الكاميرا في بداية تشغيل اللعبة من خلال خيار Starting Camera.
- جرب أوضاع الكاميرا الثلاثة، ثم تحرك داخل اللعبة واستخدم خاصيتي التكبير Zoom in والتصغير Zoom out، وسجل ملاحظتك لكل وضع:

ماذا تلاحظ	وضع الكاميرا
	Fixed
	Fixed Offset
	Free

### 4. اختبار العالم.

- تشغيل اللعبة وملاحظة أهمية تحديد وضع الكاميرا بالنسبة لمنظور اللاعب وتأثير المؤثرات الصوتية على مدى استمتاع اللاعب باللعبة.

### 5. توثيق الانجاز.

- حفظ اللعبة باسم المتعلم متبوعا بكلمة Camera.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقا.

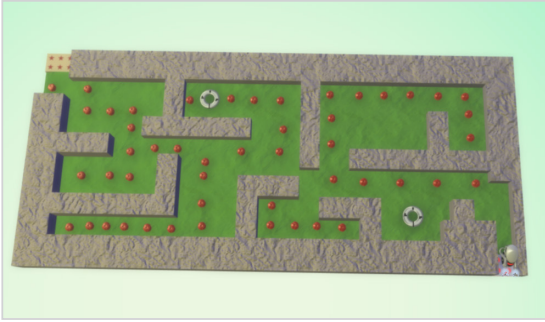


## التطبيق



### ورقة عمل 20: جامع التفاح

تخيل أنك مصمم ألعاب محترف، مهمتك اليوم هي اختيار أفضل وضع للكاميرا من منظور اللاعب، وإضافة الحيوية والتشويق للعبة من خلال إضافة المؤثرات الصوتية المناسبة للعبة، لا تنسى أن تضبط الموضوع الأولي للكاميرا.



#### 1. بناء العالم.

- شغل برنامج Kodu Game Lab واختر الأمر  
IMPORT FROM DESKTOP
- استدع مشروع Pack Cycle مجلد أوراق العمل.
- ابدأ بتحرير اللعبة.

#### 2. برمجة الكائن Cycle.

- أضف السطر البرمجي اللازم للكائن لتشغيل مؤثر صوتي لموسيقى خلفية مناسبة طوال فترة تشغيل اللعبة.
- أضف الاستجابة الثانية عند اصطدامه بالكائن Apple، وذلك بتشغيل صوت مرتبط بحدث (كسب النقاط).

#### 3. برمجة الكائن Puck.

- أضف السطر البرمجي اللازم لتشغيل صوت مرتبط بحدث (خسارة اللعبة) عند اصطدامه بالكائن Cycle.

#### 4. إعدادات الكاميرا والمؤثرات الصوتية.

- اضبط وضع الكاميرا عند بدء تشغيل اللعبة.
- اختر أفضل وضع للكاميرا أثناء اللعب.
- اضبط إعدادات الصوت بما يناسبك.

## 5. اختيار العالم.

- شغل اللعبة واستمتع بالواقعية.
- عدّل المقطع البرمجي عند الحاجة.

## 6. حفظ وتصدير اللعبة.

- احفظ العالم باسمك متبوعاً بكلمة Sound.
- صدّر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك

أستخدم منظور الكاميرا المناسب لطبيعة اللعبة.

أوظف الكاميرا لدعم التحدي داخل اللعبة.

أضيف مؤثرات صوتية مناسبة داخل اللعبة.

أضيف موسيقى خلفية مناسبة للعبة.

أربط الصوت بالأحداث داخل اللعبة.

أتحكم في مستوى الصوت بشكل متوازن.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

# خصائص متقدمة للعبة أكثر احترافية

## نواتج التعلم

- إنشاء بوابات تنقل للانتقال إلى مواقع مختلفة داخل نفس العالم.
- تطبيق مفهوم Teleporting للانتقال من مكان إلى آخر داخل نفس العالم.
- اكتساب قوة من كائن آخر وتوظيفها في تطوير أداء اللاعب داخل اللعبة.
- ربط مراحل اللعبة من خلال تفعيل خاصية Next Level للانتقال بين العوالم.
- تصميم لعبة أكثر احترافية تحتوي على مناطق لعب متعددة ومتراصة بطريقة منطقية.



ملفات أوراق العمل  
ملفات مصادر التعلم





## الاستكشاف



■ لماذا تحتاج الألعاب الاحترافية إلى أكثر من منطقة لعب؟

---

---

■ ماذا تضيف خاصية الانتقال المفاجئ (Teleport) إلى متعة اللعبة؟

---

---

■ لماذا تربط بعض الألعاب نهاية المرحلة بفتح مرحلة جديدة؟

---

---

## التعلم



تعرفنا سابقاً على كيفية ربط الأحداث بالنتائج بهدف رفع مستوى التحدي والمتعة، في هذا الدرس سنتعلم مجموعة من المهارات المتقدمة التي تضيف طابعاً احترافياً للعبة، من خلال إضافة بوابات تنقل تُمكن اللاعب من الانتقال إلى مناطق جديدة داخل العالم نفسه، وبرمجة اكتساب القوة من كائن آخر لرفع مستوى التحدي والمتعة، كما سنتعلم ربط مراحل اللعبة ببعضها من خلال تفعيل خاصية الانتقال إلى مرحلة جديدة عند تحقيق شرط محدد.

## أولاً: بوابات التنقل

لا يوفر البرنامج أمراً مباشراً لنقل اللاعب بين المواقع، لذلك يتم استخدام أسلوب برمجي ذكي لمحاكاة الانتقال (Teleport) حيث يعتمد هذا الأسلوب على مبدئين أساسيين هما: اختفاء اللاعب من موقعه الحالي، ثم ظهوره فوراً في موقع جديد ليُشعر اللاعب بأنه انتقل لحظياً داخل اللعبة. ويتم ذلك:

ثم برمجة بوابة الوصول للتحقق من هذه القيمة، فإذا كانت  $Score A = 1$  يتم توليد اللاعب مرة واحدة في الموقع الجديد



باستخدام المتغير  $Score A$  كإشارة تحكم، فعند اصطدام اللاعب ببوابة النقل، يتم إخفاؤه وإسناد القيمة  $Score A = 1$  كرسالة بدء عملية الانتقال

بهذه الطريقة يتحقق تأثير الانتقال الفوري بطريقة منظمة وذكية، مما يضيف تشويقاً ويشري التجربة التفاعلية داخل اللعبة

### 1 تفعيل خاصية التوليد لدى اللاعب:

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Saucer.
- اختيار الأمر Change Settings من القائمة الفرعية.
- تفعيل خيار Creatable.

### 2 تحديد توقيت توليد اللاعب في الأماكن المطلوبة:

- بعد تفعيل خاصية التوليد Creatable للكائن، سيختفي عند تشغيل اللعبة وسيبقى ظاهراً فقط في وضع تحرير اللعبة. لذلك يجب برمجة كائن آخر يكون مسؤولاً عن توليد اللاعب عند نقطة البداية باللعبة.
- يجب ربط حدث توليد اللاعب بشرط حدث معين مثل ربطه بلوحة نتائج غير مستخدمة ظاهرياً في اللعبة، فيتم توليده عند نقطة البداية عندما تكون النتيجة 0 ثم يتم توليده في النقطة الثانية عند وصول النتيجة إلى 1 ويتم تغيير النتيجة وربطها بكل كائن يولد اللاعب في المنطقة الجديدة للإيحاء بانتقاله.

### 3 برمجة الكائنات لخلق مفهوم التنقل داخل العالم:

الكائن المسؤول عن توليد اللاعب ، يتم وضعه بالقرب من نقطة البداية في كل جزيرة داخل العالم:

#### ■ برمجة (الحدث أو الشرط).

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Rock ثم اختيار الأمر Program.
- الضغط على علامة + في مربع When.
- اختيار scored Tile ثم إضافة تفصيل لوحة النتائج التي سيتم ربطها بعملية التوليد:

اختيار world scores ← score A ← equals ← Points ← Points 00

#### ■ برمجة الاستجابة

- الضغط على علامة + في مربع Do.
- اختيار actions Group.
- إضافة تفصيل الكائن المطلوب توليده وهو اللاعب:

اختيار create ← creatables ← cycle ← once (التوليد لمرة واحدة فقط)



يتم نسخ الكائن Rock وتوزيعه في النقاط المطلوب توليد اللاعب فيها لخلق مفهوم التنقل مع الحرص على تغيير قيمة النتيجة لكل بوابة، لضمان انتقال اللاعب بين المواقع.

#### ■ كائن بوابة التنقل

اختيار كائن مناسب ليكون بوابة تنقل من نقطة النهاية في المرحلة الأولى و المسؤول عن تغيير نتيجة لوحة Score A والتي تحدد متى يتم توليد اللاعب مرة أخرى، قبل تغيير نتيجة اللوحة يجب اخفاء اللاعب.

# خصائص متقدمة للعبة أكثر احترافية

## ■ برمجة (الحدث أو الشرط).

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Star ثم اختيار الأمر Program.
- الضغط على علامة + في مربع When.
- إضافة تفصيل الاصطدام باللاعب:



## ■ برمجة الاستجابة.

- الضغط على علامة + في مربع Do.

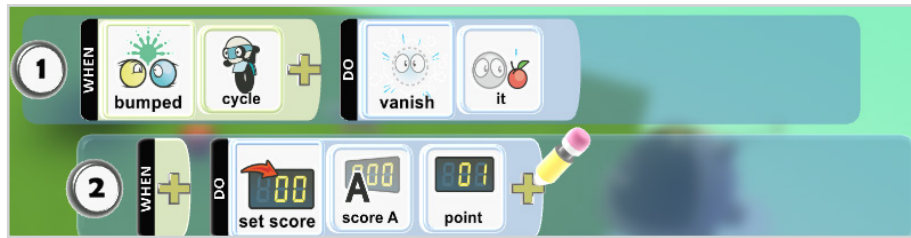


- محاذاة Indenting للأمر التالي:

- الضغط على علامة + في مربع Do ، لإضافة تفصيل لوحة النتيجة التي تم ربطها بعملية التوليد لتوضيح مفهوم التنقل كالتالي:



( 01 هي قيمة النتيجة المطلوبة للتوليد في المستوى الآخر )



## ثانياً: اكتساب القوة من كائن آخر

يمكن إضافة المزيد من التشويق للعبة من خلال تمكين اللاعب من اكتساب قوة إضافية عند التفاعل مع كائن معين داخل العالم، فتزداد قدرته على التقدم ومواجهة التحديات، ومساعدته في تطوير منطق اللعبة لتكون أكثر واقعية وممتعة:

### ■ برمجة (الحدث أو الشرط).

- اختيار أداة Object Tool من شريط الأدوات.
- الضغط بالزر الأيمن للفأرة على الكائن Heart ثم اختيار الأمر Program.
- الضغط على علامة + في مربع When.
- برمجة حدث اصطدام اللاعب بالكائن كشرط لاكتساب القوة:



### ■ برمجة الاستجابة.

- الضغط على علامة + في مربع Do.
- إضافة تفصيل بأن المطلوب اكسابه القوة هو اللاعب:



- ثم استكمال إضافة تفصيل قيمة القوة المطلوب اكتسابها (25 نقطة):



ثم استكمال قيمة النقاط من خلال اختيار points = 05.

- محاذاة Indenting للأمر التالي:

- الضغط على علامة + في مربع Do، لإضافة تفصيل إخفاء الكائن كالتالي:



## ثالثاً: الانتقال بين مراحل اللعبة باستخدام Next Level

يمكن تنظيم اللعبة بصورة أكثر احترافية من خلال ربط مراحلها ببعضها باستخدام خاصية Next Level، حيث ينتقل اللاعب إلى مرحلة جديدة عند تحقيق شرط معين، وتساعد هذه المهارة في بناء لعبة متعددة المراحل تزيد من عنصر التشويق والتحدي

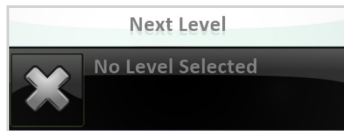
1. التأكد من تصميم وبرمجة جميع العوالم المطلوب ربطها كل عالم على حدة.
2. تحديد المرحلة التالية من خلال أداة إعدادات العالم World Settings: تفعيل الخيار .Next Level
3. ربط شرط أو حدث (مثل الحصول على عدد معين من النقاط) بالفوز والانتقال إلى المرحلة التالية، وذلك من خلال برمجة أحد الكائنات من خلال الأوامر التالية:

- إضافة البرمجة اللازمة (الشرط) الذي يحدد وصول لوحة النتائج الظاهرة للكائن إلى القيمة المطلوبة، وتشغيل صوت مناسب عند تحقيق النتيجة.
- محاذاة Indenting للأمر التالي لربط الفوز بتحقيق النتيجة النهائية.
- محاذاة Indenting للأمر التالي لتفعيل الانتقال إلى المرحلة التالية من خلال:

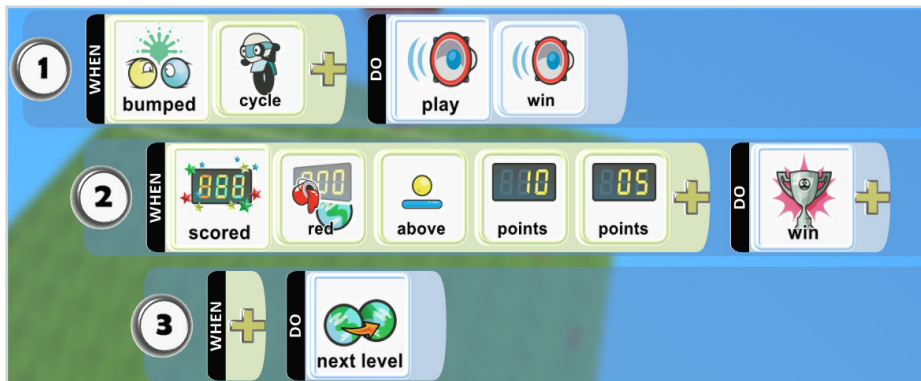
الضغط على علامة X

next level

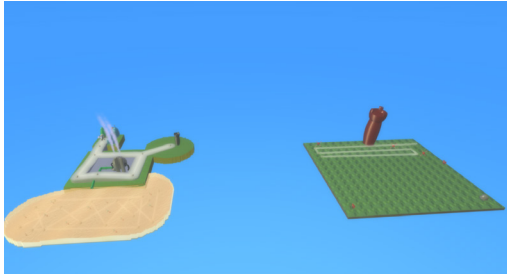
اختيار game



سيتم الانتقال إلى المشاريع المحفوظة على الجهاز، بعد اختيار (المشروع) كمستوى ثاني لربطه باللعبة يتم الضغط على Attach.



## النشاط



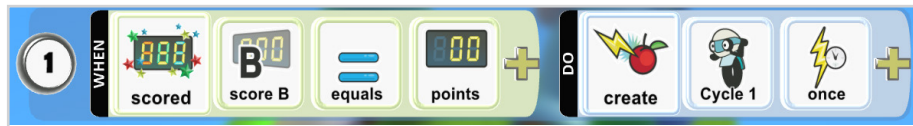
ساعد Cycle في خوض مغامرته داخل العالم، في المرحلة الأولى عليه جمع الأسماك للحصول على النقاط اللازمة للتقدم واستخدام بوابة التنقل للعبور والوصول إلى القلعة الحمراء ( المرحلة الثانية ) من اللعبة. ولكن انتبه من الأعداء الذين يحاولون منعك من الفوز، لا تقلق فالمساعدة موجودة حولك عليك البحث عنها.

### 1. بدء المغامرة

- تشغيل برنامج Kodu Game Lab واختيار الأمر IMPORT FROM DESKTOP.
- من مجلد أوراق العمل:
- تحميل مشروع المرحلة الثانية (Level Two)، ليصبح جاهزاً للاستخدام داخل البرنامج.
- استدعاء مشروع المرحلة الأولى (First Level Game).
- الضغط بالزر الأيسر للفأرة على المشروع (First Level Game) واختيار الأمر Edit.

### 2. برمجة الكائن Tree

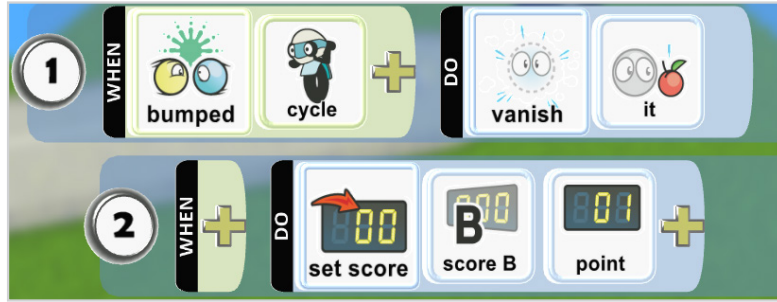
- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم لتوليد اللاعب (Cycle) في بداية اللعبة.



# خصائص متقدمة للعبة أكثر احترافية

## 3. برمجة الكائن Castle 1

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم لإخفاء اللاعب Cycle عند الاصطدام به، وإعطاء إشارة لتوليد اللاعب في المرحلة الثانية من خلال تغيير لوح النتائج الخاص بالبوابات.



## 4. برمجة الكائن Castle 2

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم للانتقال إلى المرحلة التالية في حال تحقق الشرط: (عدد النقاط الحمراء أكبر من 15) وفوز اللاعب.
- اربط المشروع Level Two كمرحلة ثانية للعبة



## 5. برمجة الكائنين: Heart 1, Heart 2

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن Heart 1 والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم لمعالجة إصابات اللاعب بنسبة 50 وذلك عند الاصطدام به، ثم الاختفاء بعد تحقق التشافي.



تكرار نفس الخطوات السابقة للكائن Heart 2.

## 6. برمجة الكائن Rock

- تحديد أداة Object Tool من شريط الأدوات.
- تحديد الكائن والضغط بالزر الأيمن للفأرة عليه واختيار الأمر Program.
- إضافة السطر البرمجي اللازم لتوليد اللاعب Cycle عند اكتماله لمستوى الجزيرة الأول بالعالم نفسه.



## 7. اختبار العالم

- تشغيل اللعبة وتحقيق الفوز للانتقال إلى المرحلة التالية.
- تعديل المقطع البرمجي عند الحاجة.

## 8. توثيق إنجازك

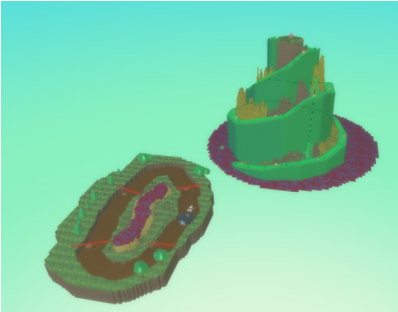
- حفظ اللعبة باسم المتعلم متبوعا بكلمة Marathon.
- تصدير اللعبة إلى سطح المكتب لمشاركتها لاحقا.



## التطبيق



### ورقة عمل 21: ماراثون الألعاب



تخيل أنك مصمم ألعاب محترف، مهمتك اليوم بناء لعبة أكثر احترافية ذات أكثر من مستوى في المستوى الأول عليك برمجة فكرة بوابات التنقل لنقل اللاعب من سباق السيارات إلى الجبل، لجمع النقاط للوصول للقمة، واستكمال النقاط المطلوبة للانتقال إلى المرحلة الثانية، والتي يجب على اللاعب فيها الوصول إلى الهدف من خلال التجوّل في المدينة، وجمع النقاط، وتفادي الأعداء. لكن لا تقلق فالمساعدة موجودة للتأكد من مقدرة اللاعب على تحقيق الفوز في كل مرحلة

#### 1. بناء العالم

- شغل برنامج Kodu Game Lab واختر الأمر IMPORT .FROM DESKTOP
- من مجلد أوراق العمل:
- حمل مشروع المرحلة الثانية (Marathon Adventure level 2)، ليصبح جاهزاً للاستخدام داخل البرنامج.
- استدع مشروع المرحلة الأولى (The Marathon Adventure).
- ابدأ بتحرير اللعبة.

#### 2. برمجة الكائن Rock1

- أضف السطر البرمجي اللازم لتوليد اللاعب Cycle لمرّة واحدة فقط عند نقطة بداية السباق (score A= 0).

### 3. برمجة الكائن Rock2

- أضف السطر البرمجي اللازم لتوليد اللاعب Cycle لمرة واحدة فقط في الجزيرة الثانية بالعالم نفسه، وذلك عندما تكون قيمة لوح النتائج المرتبطة بالتنقل (score A= 1).

### 4. برمجة الكائن Star1

- أضف السطر البرمجي اللازم لإخفاء اللاعب عند الاصطدام به، وإعطاء إشارة لتوليد اللاعب في النقطة الثانية من خلال تغيير لوح النتائج الخاص بالبوابات (score A= 1).

### 5. برمجة الكائن Star2

- أضف السطر البرمجي اللازم للانتقال إلى المرحلة التالية Marathon Adventure level 2 في حال تحقق الشرط عدد النقاط البيضاء أكبر من 20 وفوز اللاعب.

### 6. اختبار العالم

- شغل اللعبة واسع نحو الفوز بالماراثون للوصول إلى النهاية.
- عدّل المقطع البرمجي عند الحاجة.

### 7. حفظ وتصدير اللعبة

- احفظ العالم باسمك متبوعاً بكلمة Grand.
- صدر اللعبة واحفظها على سطح المكتب بنفس الاسم.



## عبر عن رأيك



أستخدم مفهوم بوابات التنقل داخل العالم نفسه.

أوظف فكرة اكتساب اللاعب قوة من كائن آخر.

أربط مرحلة بأخرى باستخدام Next Level.

أنظم قواعد اللعبة بحيث يشعر اللاعب أنه ينتقل بين مراحل متصلة ومتدرجة في الصعوبة.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

## وحدة المشروعات Projects Unit

1 مشروع برمجي متكامل .

2 مشروع لعبة The Kuwaiti Mission

3 إجراءات تنفيذ المشروع.

4 ألعاب الكترونية مقترحة.



# مشروع برمجي متكامل

### نواتج التعلم

- التخطيط للعبة رقمية تُدمج فيها الأشكال و الشخصيات بطريقة منطقية لتنفيذ فكرة.
- تصميم لعبة رقمية متكاملة باستخدام برنامج Kodu Game Lab وفق خطوات مدروسة.
- اختبار اللعبة وتتبع أخطاء البرمجة و التصميم لتطويرها.
- توظيف مكونات متعددة في اللعبة مثل: الشخصيات، الحركة، التفاعل، المتغيرات، تحديد نقاط الفوز و الخسارة.
- عرض المشروع على الزملاء و معلم الصف باستخدام وسيلة عرض مناسبة.
- تقبل التغذية الراجعة من الآخرين، واقتراح تحسينات لتطوير المشروع.
- المشاركة الفاعلة والعمل بروح الفريق في تنفيذ المشروع.



ملفات أوراق العمل  
ملفات مصادر التعلم





## مشروع لعبة The Kuwaiti Mission



### فكرة اللعبة

في إطار سعي دولة الكويت نحو مستقبل علمي وتقني متقدم، وتماشياً مع رؤيتها في دعم الابتكار واستكشاف الفضاء؛ تأتي هذه اللعبة لتأخذ اللاعب في رحلة استكشافية طموحة خارج كوكب الأرض.

يقود اللاعب مركبة فضائية كويتية في مهمة وطنية تهدف إلى إرسال أقمار صناعية إلى سطح القمر وكوكب المريخ وكوكب نبتون، متجاوزاً التحديات والعوائق في الفضاء، ومعتمداً على التخطيط والدقة واتخاذ القرار الصحيح لضمان نجاح المهمة.

### الهدف من اللعبة

يهدف اللاعب إلى بدء المهمة بإطلاق المركبة نحو القمر، ثم البحث عن العملات النقدية التي تساعد على الانتقال إلى المرحلة الثانية. بالطبع رحلته من الأرض انطلاقاً إلى القمر لا تخلو من التحديات ولكن الفوز حليف اللاعب، في المرحلة الثانية يصل اللاعب إلى سطح المريخ ويواجه العقبات لجمع الموارد اللازمة للوصول إلى كوكب نبتون حيث تقبع مركبته أعلى الجبال الجليدية للعودة إلى أرض الوطن سالماً.

1. تصميم لعبة رقمية متكاملة باستخدام برنامج Kodu Game Lab.
2. بناء لعبة تتكون من مرحلتين، وكل مرحلة تحتوي على مستويين.
3. استخدام الشروط، العدادات، البوابات، الحوار، والمساعدات.
4. تنمية مهارات التفكير المنطقي، حل المشكلات، والتصميم الإبداعي.

## مراحل اللعبة

### المرحلة الأولى: من الأرض إلى القمر:



في هذه المغامرة، يرافق اللاعبون Kodu في مهمة فضائية كويتية تبدأ تحت الماء وتنتهي على القمر، يبدأ Kodu باستكشاف قاع البحر وجمع العملات، حيث تمنح كل عملة نقطة واحدة، بينما يؤدي الاصطدام بالأعشاب البحرية إلى فقدان جزء من الصحة، وتنتهي اللعبة عند نفاذها. عند جمع 10 عملات، تُفتح بوابة مخفية ويظهر صاروخ ينقل Kodu إلى الفضاء. على سطح القمر، يجمع Kodu النجوم المتوهجة، وتمنح كل نجمة 5 نقاط. وعند الوصول إلى 30 نقطة، تنجح المهمة ويفوز اللاعب بالمستوى، تعتمد اللعبة على الاستكشاف والحذر واتخاذ القرار الذكي عبر بيئتين مختلفتين: تحت الماء وعلى القمر.

#### ■ تحميل الملف:

■ ملف المشروع The Kuwaiti Mission - Level 1 من مجلد أوراق العمل.

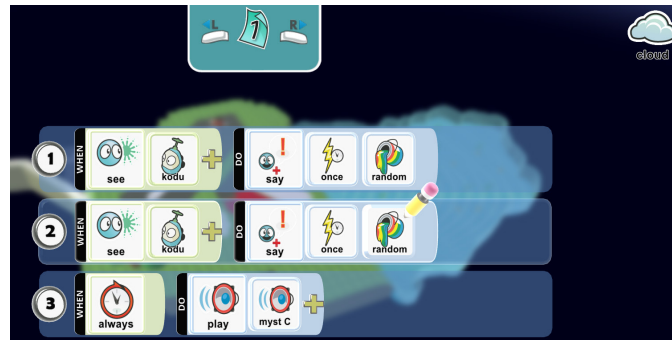
#### ■ المقاطع البرمجية:

■ المقطع البرمجي الخاص بالكائن Kodu.

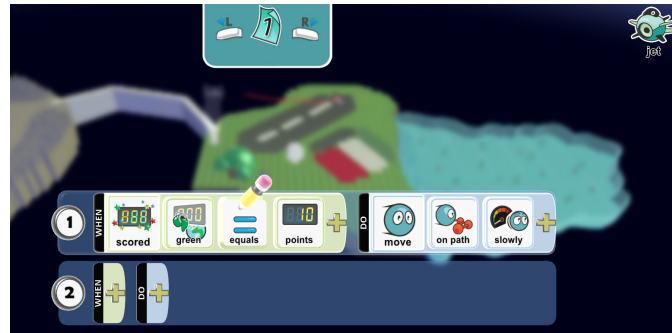




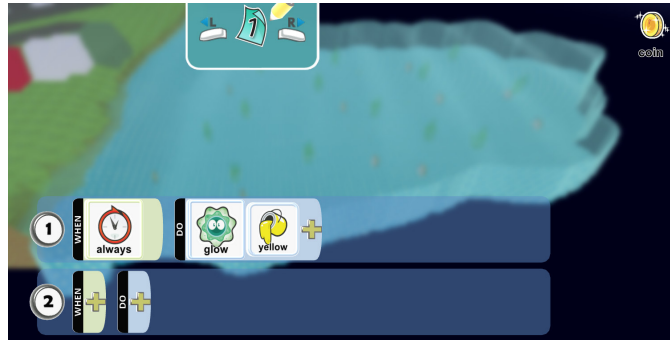
■ المقطع البرمجي الخاص بالكائن Cloud.



■ المقطع البرمجي الخاص بالكائن Jet.



## ■ المقطع البرمجي الخاص بالكائن Coin.



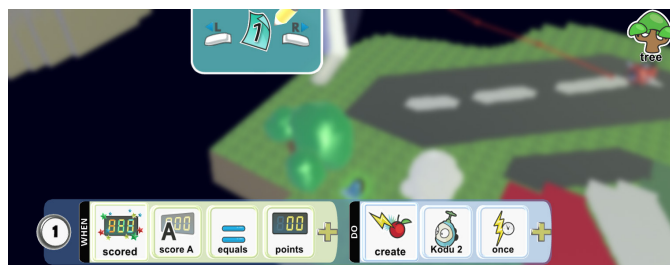
## ■ المقطع البرمجي الخاص بالكائن Stick.



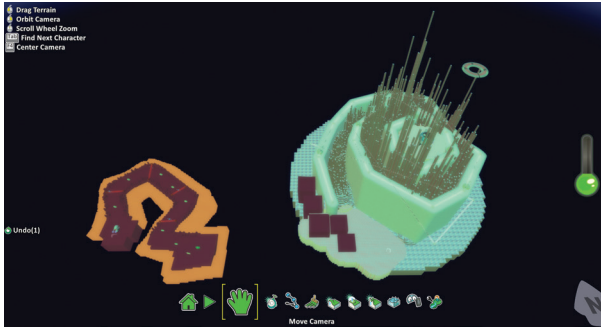
## ■ المقطع البرمجي الخاص بالكائن Star.



## ■ المقطع البرمجي الخاص بالكائن Tree.



## المرحلة الثانية: من المريخ إلى نبتون:



في البداية يخوض اللاعب سباقاً على كوكب المريخ حيث يجمع العملات ويصل إلى القلب (Heart) مع تجنّب الأقمار الصناعية الخطرة (Sputniks) في كل مرة يلمس فيها اللاعب أحد هذه الأقمار يفقد 5% من صحته، وعند وصول الصحة إلى الصفر تنتهي اللعبة بالخسارة، ولكن كائن النجمة يساعد اللاعب على التشافي.

بعد الوصول إلى القلب، يتم نقله إلى كوكب نبتون يتتبع اللاعب مساره عبر طريق جبلي جليدي، ويتسلق القمة حيث تنتظره مركبته الفضائية وعند الوصول إلى المركبة، يعود اللاعب إلى الوطن ويفوز باللعبة بالكامل.

### ■ تحميل الملف:

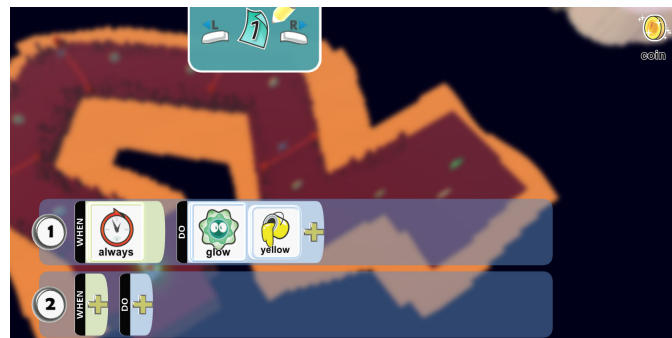
■ ملف المشروع The Kuwaiti Mission - Level 2 من مجلد أوراق العمل.

### ■ المقاطع البرمجية:

■ المقطع البرمجي الخاص بالكائن Cycle.



■ المقطع البرمجي الخاص بالكائن Coin.



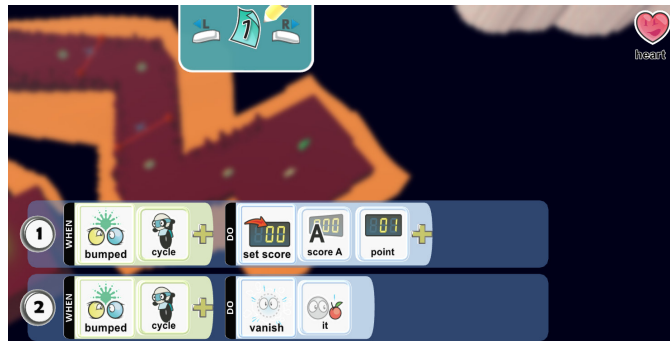
## ■ المقطع البرمجي الخاص بالكائن Sputnik.



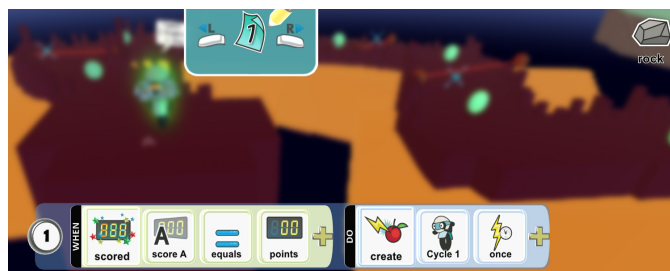
## ■ المقطع البرمجي الخاص بالكائن Star.



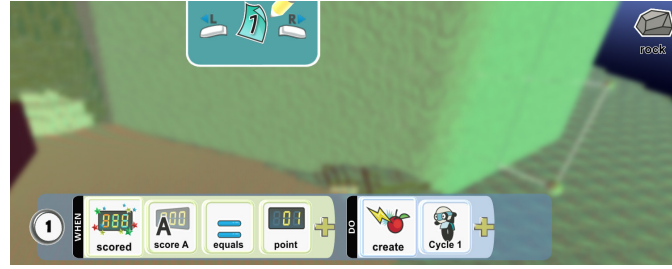
## ■ المقطع البرمجي الخاص بالكائن Heart.



## ■ المقطع البرمجي الخاص بالكائن Rock1.



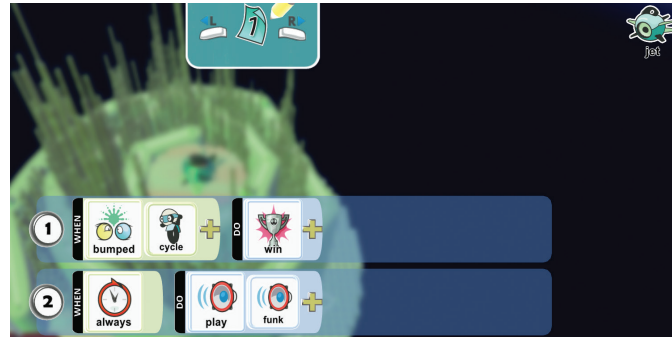
## ■ المقطع البرمجي الخاص بالكائن Rock2.



## ■ المقطع البرمجي الخاص بالكائن Puck.



## ■ المقطع البرمجي الخاص بالكائن Jet.



## تطوير اللعبة

يمكنك زيادة عدد المراحل و واستكشاف كوكب آخر من المجموعة الشمسية، أو يمكنك زيادة العوائق في كل مرحلة لتكون اللعبة أكثر صعوبة و تشويقاً أثناء اللعب.



## إجراءات تنفيذ المشروع

### الهدف الأساسي للمشروع

يهدف تنفيذ المشروع إلى توظيف المهارات التي تم اكتسابها وتطبيقها من خلال أوراق العمل، إلى جانب تنمية مهارات العمل الجماعي التعاوني، وتعزيز القدرة على العمل معاً كفريق واحد، وتصميم مشروع مميز من خلال التخطيط وتنفيذ الأفكار بطريقة منظمة ومبتكرة.

### فكرة المشروع

تصميم لعبة رقمية تقنية بسيطة باستخدام برنامج Kodu Game Lab ، تتكون من مرحلتين على الأقل، تهدف إلى حل مشكلة أو تنفيذ فكرة مفيدة بطريقة إبداعية قابلة للتطبيق، مع توظيف مهارات البرمجة والتحكم التي تم تعلمها.

سجل اسم اللعبة والفكرة بعد مناقشة أعضاء الفريق.

### مهارات المشروع

أولاً : المهارات الرقمية والتقنية:

- تصميم بيئة اللعبة ومراحلها التي تشمل: الأرض، العوائق، والأهداف.
- برمجة الأوامر الأساسية في (WHEN / DO) للتحكم في حركة الكائنات وتنفيذ الأحداث.
- استخدام الوسائط الرقمية داخل اللعبة مثل: الأصوات، المؤثرات، وتغيير المظاهر لدعم فكرة اللعبة.

## ثانياً: التفكير الناقد وحل المشكلات:

- تحليل فكرة اللعبة وتحديد المشكلة أو الهدف الذي تسعى اللعبة إلى تحقيقه.
- تجربة الأوامر البرمجية واختيار الأنسب لتحقيق شروط الفوز والخسارة.
- اختبار اللعبة واكتشاف الأخطاء والعمل على تصحيحها وتحسين الأداء.

## ثالثاً: مهارات التوثيق والبحث:

- توثيق مراحل تصميم اللعبة خطوة بخطوة (الفكرة - التصميم - التنفيذ).
- جمع معلومات داعمة لفكرة اللعبة (مثل معلومات علمية أو تعليمية مرتبطة بالمحتوى) من مصادر موثوقة.
- عرض فكرة اللعبة وشرح آلية عملها باستخدام وصف مبسط أو عرض تقديمي.

## رابعاً: مهارات العمل الجماعي:

- التعاون وتقسيم الأدوار داخل فريق العمل (مصمم - مبرمج - مشغل اللعبة).
- تحمل المسؤولية والالتزام بمهام المشروع ضمن الوقت المحدد.
- المشاركة في اتخاذ القرارات المتعلقة بتطوير اللعبة وتحسينها.

## مهام فريق العمل

تقسيم أعضاء ومهام فريق العمل على النحو التالي:

الوصف التفصيلي للمهمة	المهمة	اسم المتعلم
إدارة العمل واتخاذ القرار النهائي وتقديم المشروع للمعلم.	قائد الفريق	
	المصمم	
	مساعد المصمم	
	المبرمج	
	مساعد المبرمج	

## مراحل تنفيذ المشروع (مخطط زمني مقترح)

المهام التفصيلية لتنفيذ المشروع	المرحلة	الأسبوع
<ul style="list-style-type: none"> <li>■ اختيار فكرة لعبة مناسبة وهادفة بمراحل متعددة (مرحلتين على الأقل).</li> <li>■ تحديد المشكلة أو الهدف التعليمي للعبة وشروط الفوز والخسارة.</li> <li>■ إعداد وثيقة المتطلبات الأولية (عدد المراحل، نوع العوائق، الكائنات الأساسية).</li> </ul>	التخطيط والتحليل	الأسبوع 1
<ul style="list-style-type: none"> <li>■ توزيع الأدوار داخل فريق العمل (مصمم - مبرمج - مخبر (مشغل)).</li> <li>■ إعداد خطة تنفيذ أسبوعية لتصميم اللعبة في Kodu .</li> <li>■ تحديد الأدوات والمهارات التي سيتم استخدامها (الحركة، الاصطدام، المطاردة ..).</li> </ul>	خطة المشروع وتوزيع المهام	الأسبوع 2
<ul style="list-style-type: none"> <li>■ تصميم واجهة اللعبة وبيئة المراحل (الأرض، العوائق، الأهداف).</li> <li>■ رسم مخطط مبسط لمراحل اللعبة يوضح مسار اللاعب.</li> <li>■ توثيق التصميم الأولي بالصور أو الرسومات التوضيحية.</li> </ul>	التصميم والتوثيق الأولي	الأسبوع 3
<ul style="list-style-type: none"> <li>■ البدء في برمجة اللعبة باستخدام أوامر Kodu الأساسية (WHEN / DO).</li> <li>■ تنفيذ المرحلة الأولى عملياً واختبار التحكم بلوحة المفاتيح أو حركة ذاتية .</li> <li>■ تجربة الأدوات البرمجية والتقنية داخل Kodu.</li> </ul>	البرمجة والتنفيذ الأولي	الأسبوع 4
<ul style="list-style-type: none"> <li>■ اختبار مراحل اللعبة واكتشاف الأخطاء البرمجية.</li> <li>■ تعديل الأوامر وتحسين الأداء (السرعة، الاستجابة، الصعوبة).</li> <li>■ التأكد من تحقق شروط الفوز والخسارة بشكل صحيح.</li> </ul>	الاختبار والتحسين	الأسبوع 5
<ul style="list-style-type: none"> <li>■ إعداد التقرير النهائي للمشروع (أعضاء الفريق - اسم اللعبة - فكرة اللعبة - عدد المراحل).</li> <li>■ إعداد عرض تقديمي يشرح تصميم اللعبة وتنفيذها في Kodu.</li> <li>■ التدرب على عرض اللعبة وشرحها أمام المتعلمين والمعلم في الفصل.</li> </ul>	التوثيق النهائي والعرض	الأسبوع 6

يختلف عدد أسابيع المشروع والمراحل المختلفة وفق خطة توزيع المنهج.

## التوثيق العلمي للمشروع

### 1. المراجع:

الالتزام بتوثيق جميع المراجع العلمية المستخدمة في المشروع وفق أسلوب علمي معتمد، مثل نظام APA.

### 2. عرض المشروع ومشاركته:

يعرض فريق العمل مشروعه ومناقش فكرته ومراحله مع المعلم وزملاء الصف، مع إتاحة مشاركته عبر منصات التعلم عن بُعد مثل Teams.

### 3. الاستمرار في التعلم:

الحرص على متابعة المستجدات التقنية وتطوير المهارات البرمجية لدى المتعلم، من خلال الالتحاق بالدورات التدريبية والمشاركة في المنتديات والمجتمعات التقنية المتخصصة.

## متطلبات العرض التقديمي

يتضمن العرض التقديمي للمشروع العناصر التالية:

1. عرض بيانات المدرسة وأسماء أعضاء فريق العمل.
2. الترحيب بالمعلم والزملاء المتعلمين.
3. تقديم نبذة عن فكرة المشروع، مع توضيح الهدف منه، والمشكلة التي يعالجها، والفئة المستفيدة.
4. استعراض النموذج البرمجي للمشروع وشرح آلية عمله.
5. عرض المراجع التي تم الاعتماد عليها في تنفيذ المشروع.
6. فتح المجال لطرح الأسئلة ومناقشة المشروع.



## ألعاب إلكترونية مقترحة

### 1. لعبة حارس الحديقة:

لعبة بيئية يؤدي فيها اللاعب دور حارس يحمي الحديقة من التلوث والكائنات المزعجة باستخدام التخطيط واتخاذ القرار. الكائنات المقترحة:

Kodu - Cloud Ink - Rock - Tree - Wall - Cloud - Apple

### ■ المرحلة الأولى: تنظيف الحديقة

#### المستوى الأول:

يبدأ الكائن Cloud حوار يوضح فيه أن الحديقة ملوثة ويجب تنظيفها، ليرد عليه الكائن Kodu بأنه سيؤدي المهمة بنجاح. الهدف هو إزالة عدد محدد من Cloud Ink ، وعند الوصول للعدد المطلوب تفتح البوابة المخفية للانتقال للجزيرة الثانية في نفس المرحلة.

الخسارة في المستوى الأول تكون عند الاصطدام بالصخور المبعثرة على الطريق أكثر من ثلاث مرات

#### المستوى الثاني:

الهدف إزالة التفاح الأسود من المتاهة ذات الجدران والمسارات الضيقة وجمع أكبر عدد من الملوثات خلال وقت زمني محدد، عند انتهاء الوقت دون جمع كل الملوثات يخسر اللاعب في اللعبة، وفي حال انتهاء المهمة بنجاح سينتقل إلى المستوى الثاني من اللعبة.

### ■ المرحلة الثانية: حماية الحديقة

الهدف حماية الأشجار من الكائن Rover المطارد والذي يحاول تدميرها عند رؤيتها، وللغوز يجب حماية الأشجار من الدمار والصمود لمدة زمنية محددة. الخسارة عندما يلامس الكائن Rover اللاعب. هنالك شجرة سحرية تمنح اللاعب قوة أو سرعة أكبر.

## 2. لعبة مهمة جمع الطاقة:

لعبة مغامرات يجمع فيها اللاعب مصادر الطاقة ويتجنب العوائق المختلفة.  
الكائنات المقترحة:

Rover - Coin - Rock - Star - Water - Puck

### ■ المرحلة الأولى: جمع الطاقة

#### المستوى الأول:

يبدأ الكائن Rover حواراً يوضح فيه ضرورة جمع الطاقة Coin دون السقوط في الماء. الهدف هو جمع عدد محدد من Coin وعند الوصول للعدد المطلوب تُفتح البوابة المخفية للانتقال للمستوى الثاني. الخسارة عند السقوط في الماء المتواجد حول الجزيرة وداخلها.

#### المستوى الثاني:

الهدف جمع القطع النقدية داخل المتاهة ذات الجدران والمسارات الضيقة ، كذلك جمع أكبر عدد من الملوثات خلال وقت زمني محدد، عند انتهاء الوقت دون جمع كل الملوثات يخسر اللاعب ، وفي حال الانتهاء من المهمة بنجاح سينتقل للمرحلة الثانية من اللعبة.

### ■ المرحلة الثانية: الطاقة تحت الخطر

الهدف جمع القطع النقدية الموزعة عبر الجزيرة، وتجاوز العوائق من الصخور والحوائط المبعثرة على الطريق.  
الحذر من الكائن Puck الذي يسعى لخسارة اللاعب من خلال لمسها.  
يجد اللاعب في طريقه النجوم التي تمنحه السرعة لإنهاء المهمة.

## 3. لعبة سباق عبر الوادي:

لعبة سباق تعتمد على السرعة والتحكم داخل مسار طبيعي.  
الكائنات المقترحة:

Cycle - Kodu - Path - Coin - Rock - Wisp

### ■ المرحلة الأولى: السباق الأساسي

#### المستوى الأول:

يبدأ الكائن Cycle حواراً يوضح فيه أنه يجب عليه الوصول إلى بوابة التنقل في نهاية الطريق.

الهدف هو الوصول إلى بوابة التنقل وتفاذي الكائن Kodu الذي يُنقص من صحة اللاعب عند الاصطدام به، الخسارة عندما يصبح  $Health = 0$  بسبب الاصطدام بالكائن.

## المستوى الثاني:

الهدف جمع القطع النقدية من مسار سباق ضيق دون وجود حواجز جانبية، مع الحرص على ألا يقع اللاعب. عند انتهاء الوقت دون جمع كل القطع النقدية يخسر اللاعب، وفي حال انتهاء المهمة بنجاح سينتقل للمرحلة الثانية من اللعبة.

## ■ المرحلة الثانية: السباق النهائي

الهدف جمع القطع النقدية الموزعة عبر المتاهة، وتجاوز العوائق من الصخور. الحذر من الكائن Rock الذي قد يتسبب بخسارة اللاعب من خلال استهدافه. سيجد اللاعب في طريقه النجوم التي تمنحه قوة شفائية تساعده لإنهاء المهمة بنجاح.

## 4. لعبة متاهة الصخور الذكية:

للعبة متاهة تعتمد على التخطيط واتخاذ القرار.  
الكائنات المقترحة:

Kodu - Wall - Rock - Rover - Star

## ■ المرحلة الأولى: المتاهة الأولى

### المستوى الأول:

يبدأ الكائن Rover حواراً يوضح فيه أنه يجب عليه الوصول إلى بوابة التنقل في نهاية المتاهة، وتجنب الاصطدام بالصخور المبعثرة على الطريق.

الهدف هو الخروج من المتاهة، الكائن Star يساعد اللاعب على إنارة الطريق الصحيح، الخسارة عند يصبح  $Health = 0$  بسبب الاصطدام بالصخور.

### المستوى الثاني:

الهدف جمع القطع النقدية من متاهة أكثر تعقيداً. تحقيق الفوز يكون عند جمع كل القطع النقدية والوصول لآخر قطعة عند خط النهاية.

## ■ المرحلة الثانية: متاهة المطاردة

الهدف جمع القطع النقدية من متاهة أكثر تعقيداً بها كائنات خيالية خطيرة. الكائن Kodu يحاول أن يتسبب بخسارة اللاعب في كل مرة يصطدم به. فالخسارة تكون عندما يصبح  $Health = 0$ ، والفوز عند جمع عدد معين من القطع النقدية.

## 5. لعبة حماية محطة الطاقة:

لعبة دفاعية يحمي فيها اللاعب موقعاً ما ( مثل: المصنع Factory ) من الهجوم. الكائنات المقترحة:

Kodu - Rover - Rock - Tree - Factory - Heart

## ■ المرحلة الأولى: الدفاع الأولي

### المستوى الأول:

يبدأ الكائن Rover حواراً يوضح فيه أنه يجب منع الكائن Kodu من الوصول إلى المصنع Factory. الهدف هو الصمود لفترة زمنية بحيث تُفتح بوابة الانتقال. الكائن Heart يزيد من صحة اللاعب، والخسارة عندما يلامس الكائن Kodu المصنع.

### المستوى الثاني:

الدفاع عن المصنع في تضاريس وموقع مختلف، والصمود لفترة زمنية أطول مع زيادة أعداد Kodu الذي يحاول التسبب بخسارة اللاعب، حيث تكون الخسارة عندما يلامس الكائن Kodu المصنع.

## ■ المرحلة الثانية: الهجوم الأخير

الدفاع عن المصنع في تضاريس وموقع أكثر تعقيداً، والصمود لفترة زمنية أطول مع زيادة Kodu الذي يحاول التسبب بخسارة اللاعب، حيث تكون الخسارة عندما يلامس Kodu المصنع.

## 6. لعبة مغامرة تحت الماء:

لعبة مغامرات تعليمية في أعماق البيئة البحرية لتنظيفه من المخلفات الضارة بالبيئة. الكائنات المقترحة:

Octopus - Starfish - Clam - Seagrass - Coin - Lily Pad - Heart - Light

## ■ المرحلة الأولى: الاستكشاف تحت الماء

### المستوى الأول:

يبدأ الكائن Starfish الحوار مع الكائن Octopus يوضح فيه أنه يجب عليه جمع عدد معين من القطع النقدية لفتح بوابة التنقل الكائن Light يساعد اللاعب عن طريق تقديم معلومات عن موقع القطع النقدية، والخسارة عند ملامسة الكائن Seagrass ووصول صحة اللاعب إلى 0.

### المستوى الثاني:

جمع عدد معين من القطع النقدية في موقع ذي تضاريس مختلفة وعوائق، حيث تتسبب بعض الكائنات مثل Clam و Seagrass بفقدان اللاعب جزء من صحته، وتكون الخسارة عند وصول الصحة إلى  $Health = 0$ .

## ■ المرحلة الثانية: الكنز الكبير

جمع عدد معين من القطع النقدية في موقع ذي تضاريس مختلفة وعوائق، حيث أن الكائن Clam يحاول اللحاق باللاعب للتأكد من خسارته في اللعبة عند الملامسة.



## عبر عن رأيك



أشارك أفكارى مع أعضاء الفريق.

أوظف مهارات التصميم و البرمجة لإنتاخ لعبة.

أختبر اللعبة.

أحفظ اللعبة.

أصدر اللعبة.

أعرض المشروع على المعلم والزملاء باستخدام وسيلة عرض مناسبة.

أقدر أهمية العمل الجماعى فى إنجاز المشروع.



## ملاحظات المعلم

الملاحظات:

التاريخ:

اليوم:

## ملاحظات ولي الأمر

- Hinton, G. (2017). The deep learning revolution. [Interview]. BBC News.
- OpenAI. (2025). Generative images of Hamad, Sarah, and Jalbot in scenes from Smart Future Kuwait [AI-generated digital illustrations using GPT-5 (DALL,E)]. ChatGPT. <https://chat.openai.com/>
- U.S. Department of Education. (2023). Artificial intelligence and the future of teaching and learning: Insights and recommendations. Office of Educational Technology. <https://www.ed.gov/sites/ed/files/documents/ai-report/ai-report.pdf>
- Alalaq, A. S. (2025). AI-Powered Search Engines. ShodhAI: Journal of Artificial Intelligence, 2(1), 49—62.
- PMC. (2023). Search Engines and Generative Artificial Intelligence Integration. In PMC
- Sahoo, P., Singh, A. K., Saha, S., Jain, V., Mondal, S., & Chadha, A. (2024). A systematic survey of prompt engineering in large language models: Techniques and applications. arXiv. <https://arxiv.org/abs/2402.07927>
- Learn Prompting. (2024). Learn prompting: An open-source course on prompt engineering. Retrieved from <https://learnprompting.org/>
- Microsoft. (2024). Prompt engineering techniques — Azure OpenAI Service documentation. Microsoft Learn. Retrieved from <https://learn.microsoft.com/en-us/azure/ai-foundry/openai/concepts/prompt-engineering>
- Zhou, Y., Chen, H., Liu, X., & Zhao, J. (2024). A systematic survey of prompting techniques for large language models. arXiv preprint arXiv:2406.06608. Retrieved from <https://arxiv.org/abs/2406.06608>
- The Prompt Report. (2024). Comprehensive overview of prompt engineering research and trends. arXiv. Retrieved from <https://arxiv.org/abs/2402.07934>
- Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., & Neubig, G. (2023). Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. ACM Computing Surveys, 55(9), 1—35. <https://doi.org/10.1145/3560815>
- Freepik. (n.d.). Programmer Phatplus lineal color icon [Icon]. Freepik. [https://www.freepik.com/icon/programmer\\_1681499](https://www.freepik.com/icon/programmer_1681499)
- McCarthy, J. (2007). What is artificial intelligence? Stanford University. Retrieved from <http://jmc.stanford.edu/articles/whatisai/whatisai.pdf>

- Li, F.-F. (2023). *The world I see: Curiosity, exploration, and the power of AI*. Flatiron Books.
- Ji, Z., Lee, N., Frieske, R., Yu, T., et al. (2023). Hallucinations in large language models: A survey. arXiv preprint, arXiv:2309.01219. <https://arxiv.org/abs/2309.01219>
- Maynez, J., Narayan, S., Bohnet, B., & McDonald, R. (2020). On faithfulness and factuality in abstractive summarization. In *Proceedings of ACL 2020* (pp. 1906—1919). <https://doi.org/10.18653/v1/2020.acl-main.173>
- Vaughan, L., & Thelwall, M. (2021). Web search: Multidisciplinary perspectives. *Annual Review of Information Science and Technology*, 55(1), 1—30. <https://doi.org/10.1002/aris.1440550101>
- Pan, B., Hembrooke, H., Joachims, T., et al. (2007). In Google we trust: Users' decisions on search results. *Journal of Computer-Mediated Communication*, 12(3), 801—823. <https://doi.org/10.1111/j.1083-6101.2007.00351.x>
- Floridi, L. (2018). *The philosophy of information* (2nd ed.). Oxford University Press.
- Wardle, C., & Derakhshan, H. (2017). Information disorder: Toward an interdisciplinary framework. Council of Europe Report. <https://rm.coe.int/information-disorder-report/168076277c>
- IEEE Digital Privacy Initiative. (a.n.d.). What is digital privacy and its importance? IEEE. <https://digitalprivacy.ieee.org/publications/topics/what-is-digital-privacy-and-its-importance/>
- Quach, S., Thaichon, P., Martin, K. D., Weaven, S., & Palmatier, R. W. (2022). Digital technologies: tensions in privacy and data. *Journal of the Academy of Marketing Science*, 50(6), 1299-1323. <https://doi.org/10.1007/s11747-022-00845-y>
- IEEE Digital Privacy Initiative. (n.d.). The role of standards in digital privacy. IEEE. <https://digitalprivacy.ieee.org/publications/topics/the-role-of-standards-in-digital-privacy/>
- Belenguer, L. (2022). AI bias: Exploring discriminatory algorithmic decision-making. *Frontiers in Artificial Intelligence*, 5, 8830968. <https://doi.org/10.3389/frai.2022.8830968>
- Paul, J. (2024, November). Privacy and data security concerns in AI-driven data processing. ResearchGate. [https://www.researchgate.net/publication/385781993\\_Privacy\\_and\\_data\\_security\\_concerns\\_in\\_AI](https://www.researchgate.net/publication/385781993_Privacy_and_data_security_concerns_in_AI)
- Ferrara, E. (2023). Fairness and bias in artificial intelligence: A brief survey of sources, impacts and mitigation strategies. *Information*, 6(1), 3. <https://doi.org/10.3390/info6010003>
- Kunle-Lawanson .O. (2022). The role of AI in information security risk management. *World Journal of Advanced Engineering Technology and Sciences*, 7(2), 308-319. <https://doi.org/10.3390/wjates7020308>

org/10.30574/wjaets.2022.7.2.0128

- Arrible. (2023). The game development & its main stages. <https://arrible.com/blog/the-game-development-its-main-stages/>
- CG Spectrum. (2021). How video games are made: The game development process. <https://www.cgspectrum.com/blog/game-development-process>
- GDKeys. (2023). Game development process: Game production pipeline. <https://gdkeys.com/game-development-process/>
- Jiang, Y., & Zheng, L. (2020). Deep learning for video game genre classification. arXiv. <https://arxiv.org/abs/2011.12143>
- KoduGameLab. (n.d.). Kodu Game Lab — 3D game programming for kids. <https://www.kodugamelab.com/>
- Qaffas, A. A. (2020). An operational study of video games÷ genres. *iJIM*, 14(15), 185—200. <https://doi.org/10.3991/ijim.v14i15.16691>
- Mario Kart World [Video game screenshot]. (2025, June 11). In Game review: Mario Kart World (Nintendo Switch 2) “ FutureFive. Retrieved from <https://futurefive.co.nz/story/game-review-mario-kart-world-nintendo-switch-2>
- Rocket League Sideswipe [Video game screenshot]. (2021, December 21). In J. Minor, Rocket League Sideswipe (for iOS). PCMag UK. <https://uk.pcmag.com/mobile-games/137822/rocket-league-sideswipe-for-ios>
- Minecraft [Video game screenshot]. (2025, July 16). In J. Skrebels, New content, new looks, and movie tie-ins: Minecraft is more lively than ever this summer. Xbox Wire. <https://news.xbox.com/en-us/2025/07/16/minecraft-updates-summer-2025/>





7

