

الوحدة الرابعة: البرمجة باستخدام سكراثش



أهداف التعلم

ستتعلم في هذه الوحدة:

- < استخدام لبنة كرّر حتى (repeat until).
- < المعاملات الحسابية في سكراتش.
- < أنواع المتغيرات المختلفة وكيفية استخدامها لتخزين المعلومات.
- < إجراء العمليات الحسابية في سكراتش.
- < اتخاذ القرارات باستخدام الشروط المركبة.
- < ماهية النظام الإحداثي الديكارتي.
- < استخدام الإحداثيات في البرمجة.
- < التحكم في الكائنات باستخدام لوحة المفاتيح وإحداثياتها.
- < اتخاذ القرارات المركبة باستخدام المُعامِلات المنطقية.
- < استخدام تقنيات الرسوم المتحركة.
- < إنشاء لعبة صغيرة وبرمجتها.

أهلاً بك

في هذه الوحدة، ستجري عمليات حسابية باستخدام اللبنة، وستنشئ لعبة بسيطة من خلال الدمج بين اللبنة الثلاث الجديدة التي ستتعلمها. كما ستصمم وتبرمج لعبة بمواصفات متقدمة.

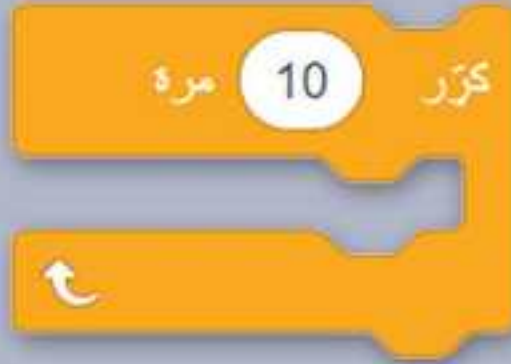
الأدوات

< منصة سكراتش من معهد ماساتشوستس للتقنية (MIT Scratch)

هل تذكر؟

لبنة كَـرّر (Repeat)

يتم استخدام لبنة كَـرّر (Repeat) عندما تريد تنفيذ مجموعة من اللبنات لعدد معين من المرات.



لبنة كَـرّر باستمرار (Repeat Forever)

ستنفذ لبنة كَـرّر باستمرار (Repeat Forever) اللبنات الموجودة بداخلها.



أوقف () () (stop)

لإيقاف لبنة كَـرّر باستمرار، عليك ضغط علامة التوقف (stop sign) أو تنشيط لبنة أوقف () () (stop).



علامة التوقف

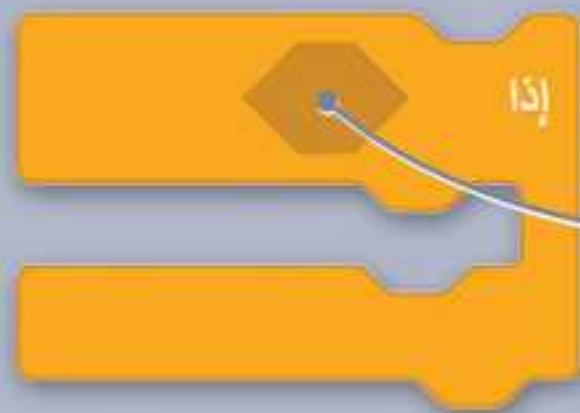


حدد من القائمة المنسدلة المقاطع البرمجية التي تريد إيقافها.

لبنة إذا () (if) (then)

تتحقق لبنة إذا () (if) أولاً من الشرط، فإذا كان الشرط صحيحاً، فيتم تنفيذ اللبنات الموجودة داخلها، وإذا كان الشرط خطأً، فيتم تخطي اللبنات.

إذا كنت بحاجة إلى التحقق من أكثر من شرط، فعليك استخدام المزيد من لبنات إذا () (if) وإلا.



الشرط

ارتد إذا كنت عند الحافة

لبنة ارتد إذا كنت عند الحافة (if on edge, bounce)

تفحص لبنة ارتد إذا كنت عند الحافة إذا كان الكائن يلامس حافة الشاشة أم لا، فإذا قام بلامستها، فذلك يمنع من الاستمرار في حركته. كما أنه يدور 180 درجة.

اتجه نحو الاتجاه 90



لبنة اتجه نحو الاتجاه () () (point in direction)

توجه لبنة اتجه نحو الاتجاه () الكائن في اتجاه معين. لذلك، يدور الكائن في زاوية محددة.

اجعل نمط الدوران يمين - يسار ▼

✓ يمين - يسار

لا دوران

في جميع الاتجاهات

لبنة اجعل نمط الدوران () () (set rotation style)

تتحكم لبنة اجعل نمط الدوران () في اتجاه حركة الكائن.

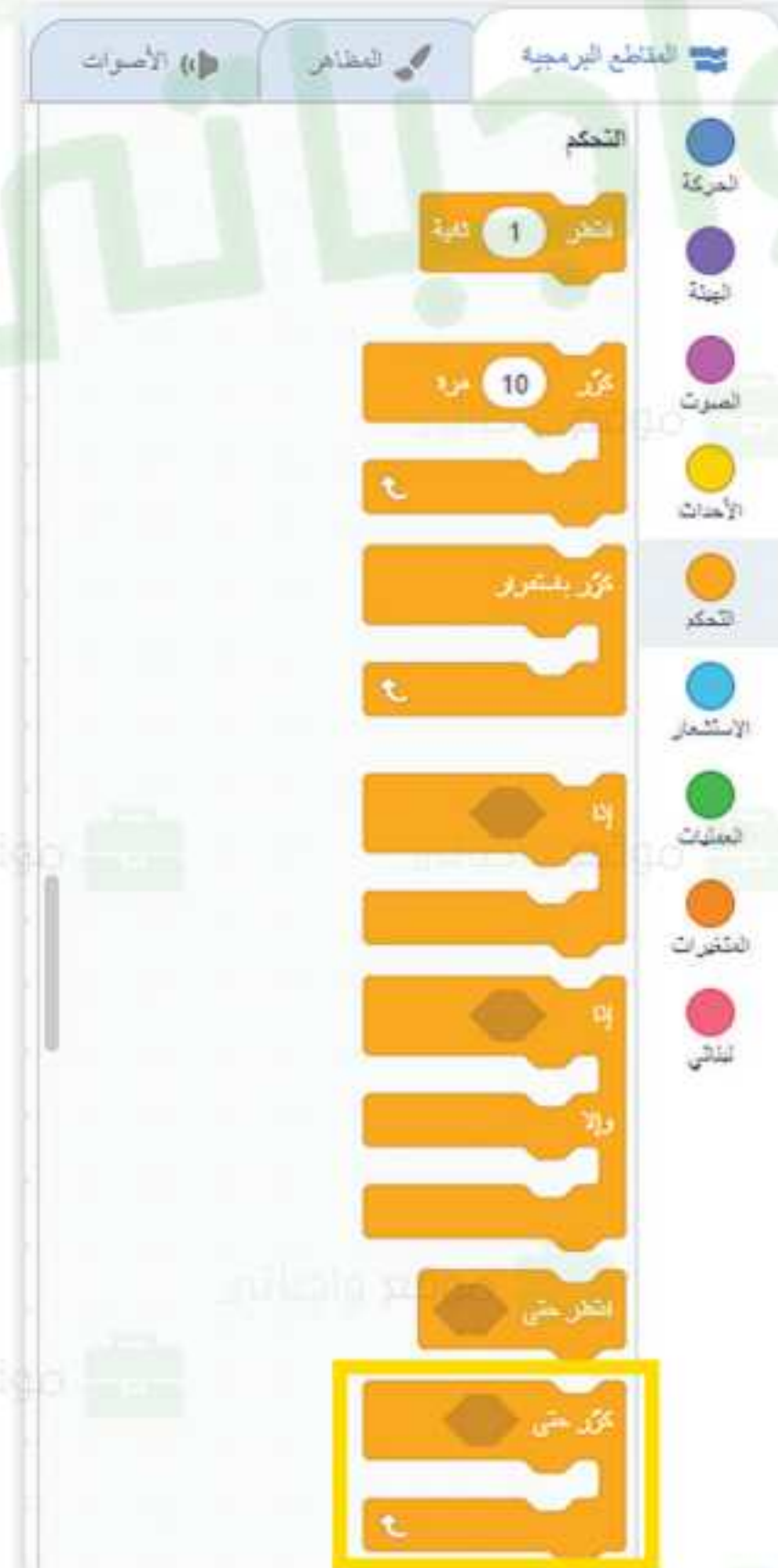


الدرس الأول: التكرار في سكراتش

كما تعلمت سابقاً، يمكنك جعل جهاز الحاسب ينفذ اللبنة عدة مرات، باستخدام ثلاثة أنواع من الحلقات: كُرّر (repeat)، وكُرّر باستمرار (repeat forever)، وكُرّر حتى (repeat until) يسمح لك بتكرار نفس الأوامر مراراً وتكراراً. في هذا الدرس، ستستخدم لبنة كُرّر حتى (.)

كُرّر حتى

لبنة كُرّر حتى (repeat until) هي إحدى لبنات التحكم (control) تسمح لك بتكرار مجموعة من الإجراءات حتى يتم استيفاء شرط معين. بعد سحب اللبنة إلى منطقة البرمجة النصية، تحتاج إلى تحديد الشرط الذي سيوقف الحلقة. ستستمر الحلقة في تكرار مجموعة الإجراءات داخل اللبنة حتى يتحقق الشرط.



يتم استخدام لبنة كُرّر حتى عندما لا تعرف عدد التكرارات، ويتوقف تكرار اللبنة الموجودة بداخلها حين يصبح الشرط صحيحاً.



يجب وضع اللبنة التي تريد تكرارها داخل لبنة كُرّر حتى.

المقطع البرمجي الآتي يجعل القطة تتحرك 10 خطوات في كل تكرار حتى تصل إلى الحافة، وعندما تصل إلى الحافة تتوقف وتقول "هذه هي الحافة".



لإنشاء مقطع برمجي باستخدام لبنة كرّر حتى (repeat until):

< أضف لبنة عند نقر العلم الأخضر (when flag clicked) من فئة لبنات الأحداث (Events). ①

< اسحب وأفلت لبنة كرّر حتى (repeat until) من فئة التحكم (Control) في منطقة المقطع البرمجي. ②

< أضف لبنة ملامس لـ () () touching من فئة لبنات الاستشعار (Sensing) واختر الحافة (edge). ③

< ضع لبنة تحرك (10) خطوة (move (10) steps) من فئة لبنات الحركة (Motion) داخل تكرار لبنة كرّر حتى (repeat until). ④

< أضف لبنة قل () لمدة (2) ثانية (say () for 2 seconds) من فئة لبنات الهيئة (Looks) واكتب داخلها "هذه هي الحافة". ⑤



شاهد مثالاً مختلفاً. تريد إنشاء مقطع برمجي يتحقق من إدخال المستخدم لكلمة المرور الصحيحة، على سبيل المثال "Scratch".



طريقة عمل المقطع البرمجي:

- 1 يسأل المستخدم عن كلمة المرور الخاصة به. *****
 - 2 ثم يتحقق من الشرط، فإذا كان الشرط خطأ، يبدأ التكرار، ويتم تنفيذ اللبنة الموجودة داخل لبنة كُرّر حتى؛ لذلك يطلب المقطع البرمجي كلمة المرور مرة أخرى من المستخدم.
 - 3 طالما تم تقييم الشرط على أنه خطأ، يُكرر تنفيذ اللبنة الموجودة داخل التكرار.
- عندما يصبح الشرط صحيحاً، يتوقف التكرار ويتم تنفيذ اللبنة الموجودة بعد لبنة كُرّر حتى.

شغل المقطع البرمجي لترى كيف يعمل.

أدخل 1234
كلمة مرور
وستشاهد هذه
الرسالة:



كلمة مرور خاطئة، حاول مرة
أخرى ..

أدخل كلمة مرور جديدة.
واكتب Scratch اضغط
على مفتاح Enter أو
اضغط على ✓ وستشاهد
هذه الرسالة على الشاشة:



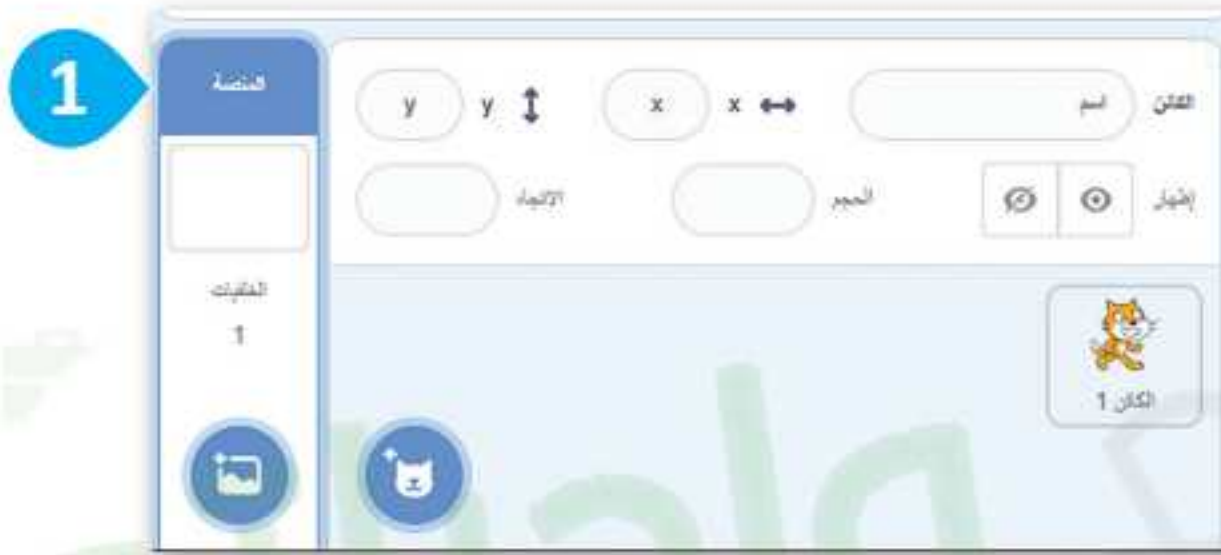
لقد تمت بتسجيل الدخول

استخدام لبنة كرر حتى في لعبة المتاهة

لتشاهد كيف يمكن استخدام لبنة كرر حتى في الألعاب.

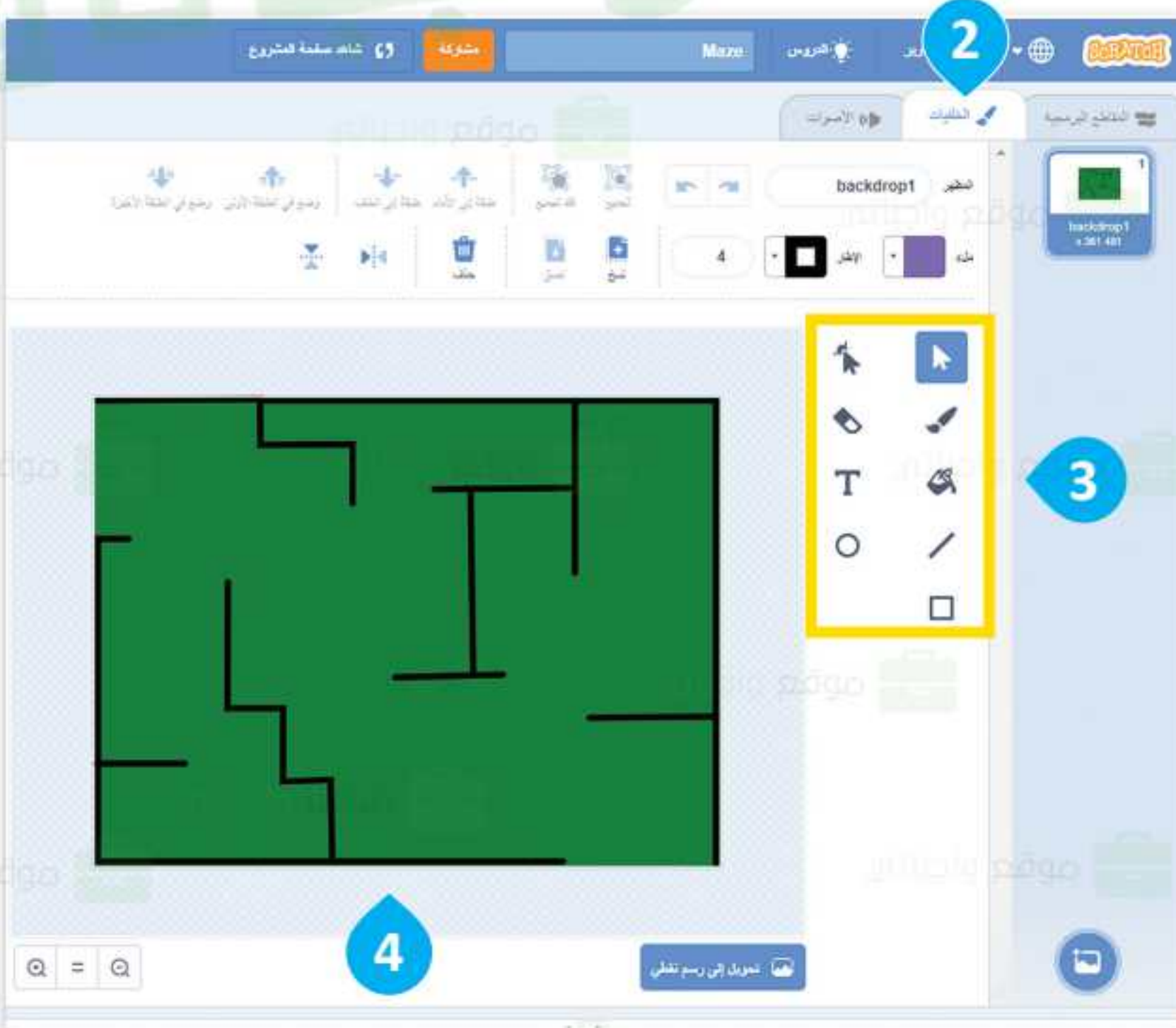
مهمتك هي تصميم لعبة حيث تتحرك الدجاجة عبر المتاهة، وتجمع البيض الموجود في طريقها حتى تصل إلى مخرج المتاهة، حيث تضع البيض داخل الوعاء.

1. أنشئ خلفية المتاهة



لإنشاء خلفية للمتاهة
اتبع الخطوات الآتية:

- < اضغط على قسم المنصة (Stage)، وانتقل إلى علامة تبويب الخلفيات (Backdrops).
- < استخدم أدوات الرسم.
- < ارسم المتاهة في الصورة خطوة بخطوة.



2. إضافة الكائنات

الآن بعد أن أصبحت الخلفية جاهزة، عليك حذف كائن القطة وإضافة الكائنات: دجاجة (Hen)، وعاء (Bowl)، بيضة (Egg)، وضعها في أماكنها الصحيحة.



Egg

بيضة



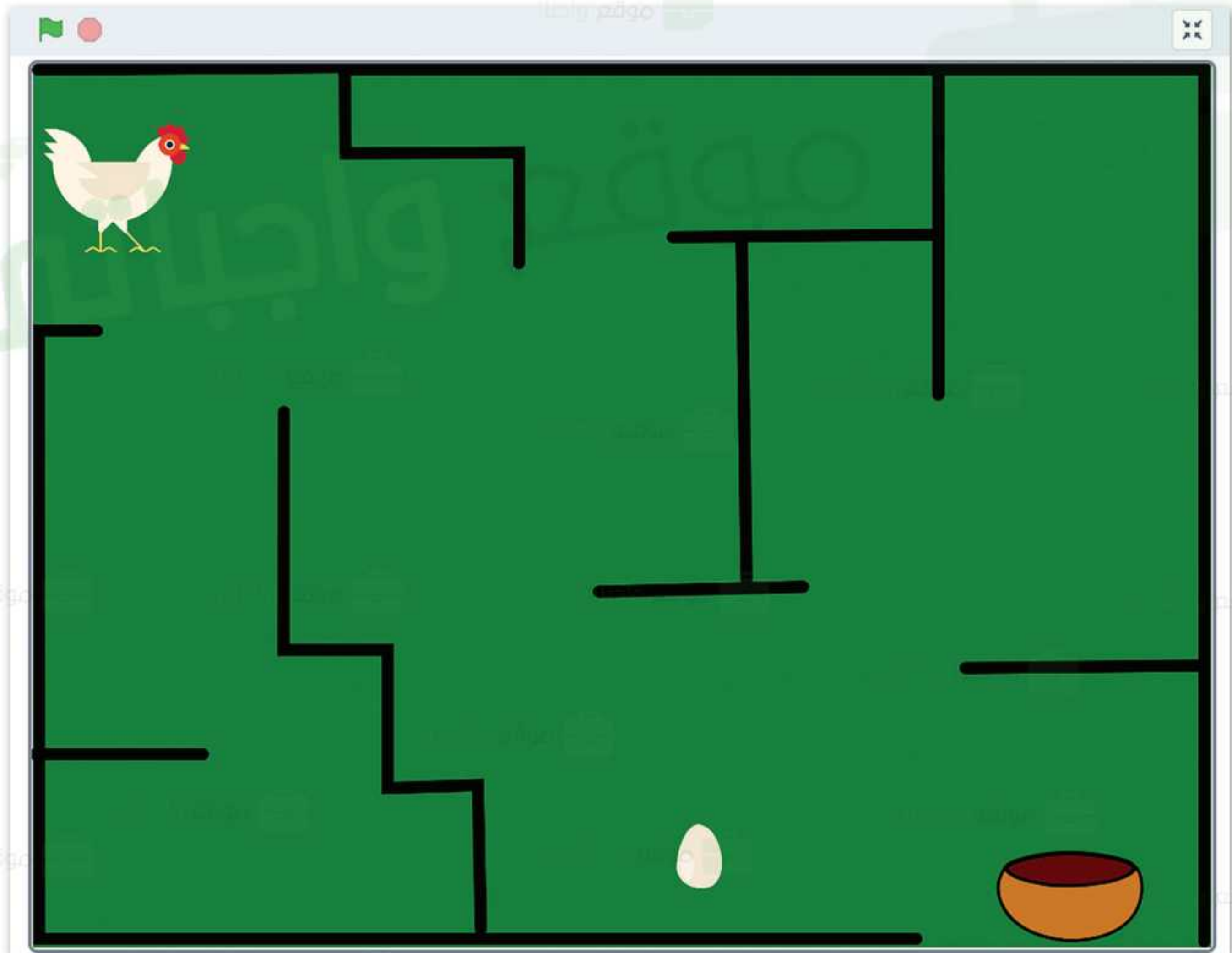
Bowl

وعاء



Hen

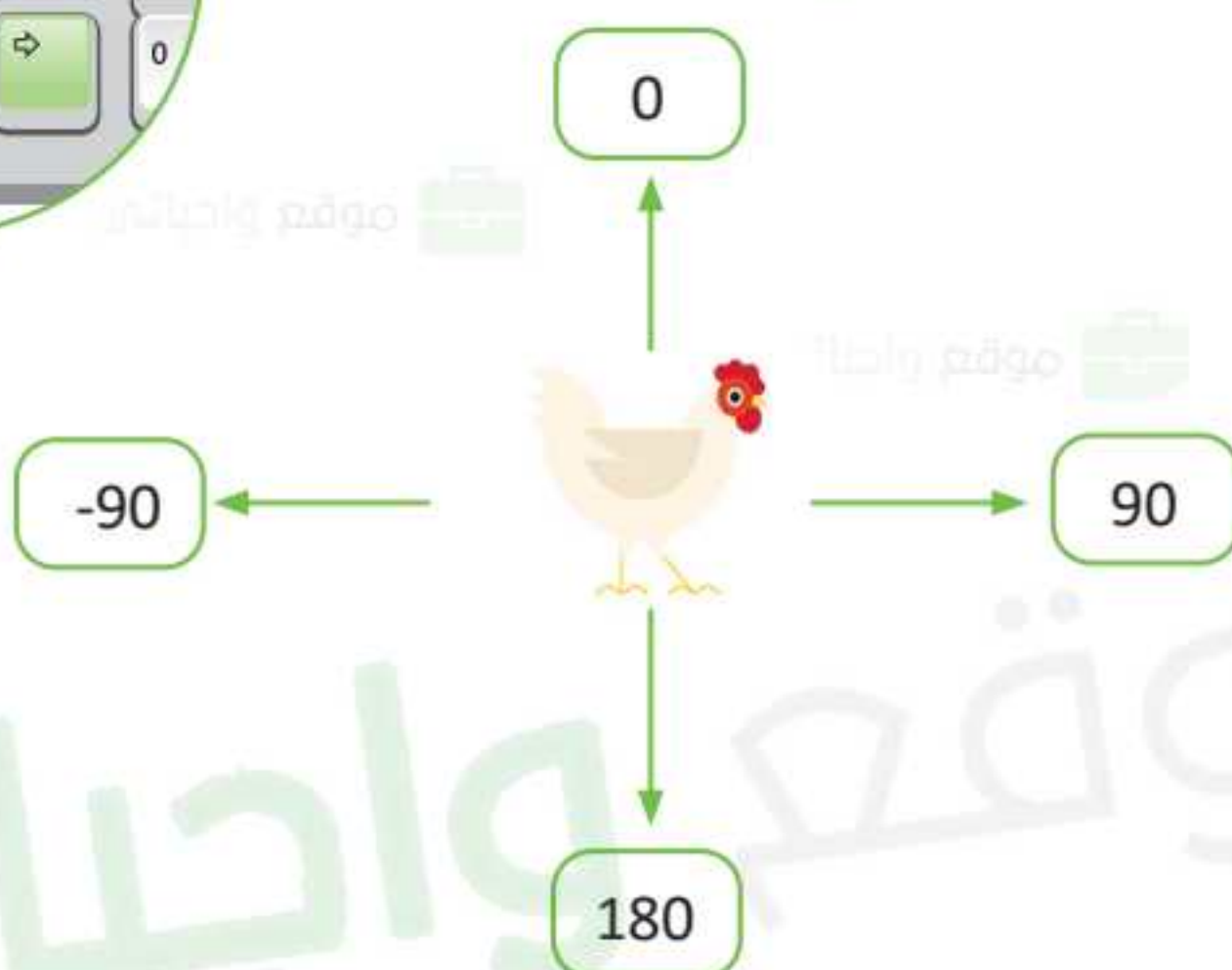
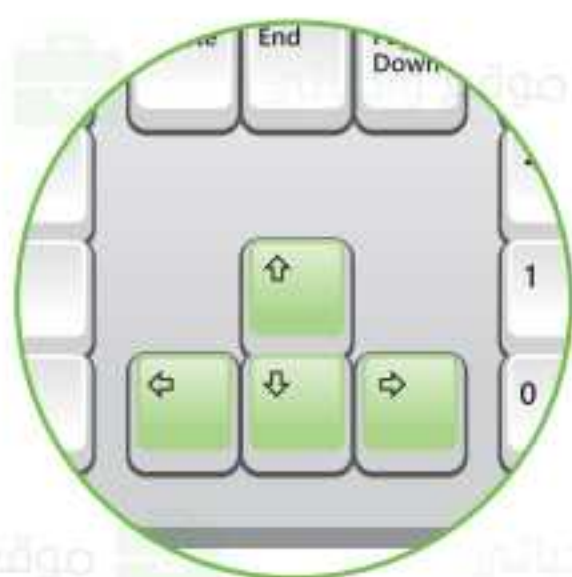
دجاجة



3. كتابة المقطع البرمجي للعبة

لتحريك الدجاجة، ستستخدم لبنة كَرَر حتى ولبنة ملامس لـ () () touching مع اختيار الكائن Bowl كشرط.

ستتحرك الدجاجة داخل المتاهة باستخدام مفاتيح الأسهم.



يجب على الدجاجة تجنب الحوائط، فعندما تلامس الحائط فإنها تتحرك 5 خطوات للخلف. ولجعل الدجاجة تنفذ هذا، عليك استخدام لبنة ملامس اللون () () touching color).



معلومة

أداة انتقاء اللون (Color Picker) هي أداة تُستخدم لتحديد لون على أي صورة مفتوحة على شاشتك.

لنطبق معًا

تدريب 1

إيقاف حركة الكائن باستخدام لوحة المفاتيح

أنشئ المقطع البرمجي الآتي ثم حاول إيقاف حركة الكائن. ما المفتاح الذي ضغطت عليه؟



مفتاح المسافة

الحل



تدريب 2

إيقاف حركة الكائن باستخدام لوحة المفاتيح

غيّر المقطع البرمجي الآتي ثم أجر التغييرات المناسبة لإيقاف حركة الكائن عند الضغط على الحرف "s".



تدريب 3

تحريك الكائن بشكل متكرر

أنشئ خوارزمية ولبنة من التعليمات البرمجية لجعل الكائن يستدير بمقدار 30 درجة حتى تضغط على أي مفتاح من لوحة المفاتيح. **خطوات الخوارزمية**



1. يبدأ البرنامج عند نقر العلم.
2. يتحقق البرنامج من شرط الضغط على أحد مفاتيح لوحة المفاتيح.
3. إذا لم يتحقق الشرط يستدير الكائن بمقدار 30 درجة.
4. تكرر الخطوات الثلاثة حتى يتحقق الشرط.

تدريب 4

إنشاء خوارزمية وكتابة مقطع برمجي

أنشئ خوارزمية ومقطعًا برمجيًا يجعل الكائن يتوقف عند الضغط على زر الفأرة بحيث:

- يجعل الكائن يقول "مرحبًا" لمدة ثانيتين.
- يسأل المُستخدم إذا كان يريد أن يمشي الكائن.
- إذا كانت الإجابة بنعم، فسوف يتحرك الكائن خطوتين باستمرار حتى يتم الضغط على زر الفأرة.



خطوات الخوارزمية:

1. يبدأ البرنامج عند نقر العلم.
2. يقول الكائن "مرحبًا" لمدة ثانيتين.
3. يسأل المستخدم إذا كان يريد أن يمشي الكائن.
4. إذا كانت الإجابة بنعم يتحرك الكائن خطوتين باستمرار.
5. يتحقق البرنامج من شرط الضغط على زر الفأرة.
6. يكرر الكائن التحرك خطوتين حتى يتحقق الشرط وهو الضغط على زر الفأرة.

تدريب 5

كتابة مقطع برمجي

أنشئ المنصة الآتية بحيث:

- يجعل الكائن يتحرك حتى يلمس كرة القدم.
- عندما يلمس الكائن كرة القدم، فإنها ستتحرك حتى تلمس حافة المنصة.



Ben



Soccer Ball



الدرس الثاني: برمجة العمليات الحسابية

العمليات الحسابية

لقد تعلمت سابقًا العمليات المنطقية ($=$, $<$, $>$).

لبنة () أكبر من () () more than () .



لبنة () أصغر من () () less than () .



ولبنة () يساوي () () equal to () .



الآن، سوف تتعلم العمليات الحسابية. يمكنك استخدام سكراتش لتنفيذ أي نوع من العمليات الحسابية مثل: الجمع، والطرح، والضرب، والقسمة، وغير ذلك. تُستخدم المُعاملات (Operators) في البرمجة لإجراء الحسابات، فالمُعامل هو رمز يمثل إجراءً محددًا، على سبيل المثال: علامة الجمع (+) هي المُعامل الذي يمثل الجمع. تسمى المُعاملات التي تُستخدم لإجراء العمليات الحسابية المُعاملات الرياضية (Mathematical Operators). يمكنك العثور على المُعاملات الرياضية في فئة لبنات العمليات (Operators).



تُستخدم الرموز الآتية في البرمجة لتمثيل العمليات الحسابية:



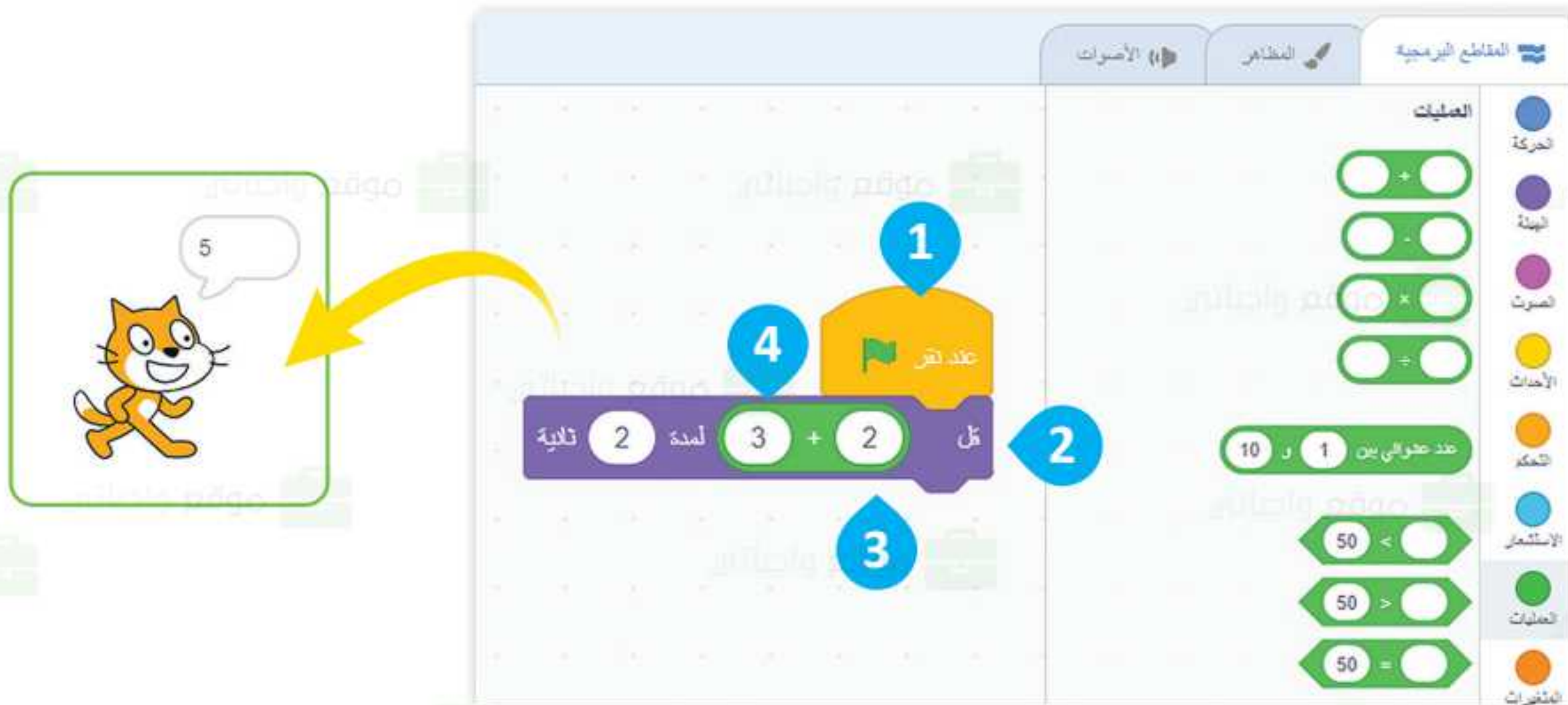
تُكتب العمليات الحسابية في البرمجة بطريقة مختلفة عن كتابتها الرياضيات.

المُعامَلات في البرمجة

البرمجة	الرياضيات
$2 + 4$	$2 + 4$
$2 - 4$	$2 - 4$
$2 * 4$	2×4
$2 / 4$	$2 \div 4$

لإجراء عملية الجمع:

- 1 < أضف لبنة عند نقر العلم الأخضر (when flag clicked) من فئة لبنات الأحداث (Events).
- 2 < أضف لبنة قل () لمدة (2) ثانية (say () for 2 seconds) من فئة لبنات الهيئة (Looks).
- 3 < اسحب وأفلت لبنة الجمع (addition) وضعها داخل لبنة قل (say).
- 4 < اكتب الأرقام التي تريد جمعها.



المتغيرات في سكراتش

يشير اسم المتغير إلى مكان محدد في ذاكرة جهاز الحاسب، ويستخدم لتخزين البيانات أثناء تنفيذ المقطع البرمجي.

المتغيرات في سكراتش.



القيمة

الاسم

يحتوي سكراتش على متغير مُعد سابقًا، يسمى متغيري (My Variable) وهو جاهز للاستخدام.

يمكنك العثور على جميع اللبئات الخاصة بالمتغيرات في فئة لبنات المتغيرات (Variables).



كل متغير له اسم فريد وقيمة.



من المهم ملاحظة أن المتغيرات في سكراتش تنتمي إلى الكائن الذي تم إنشاؤها فيه، مما يعني أن كل كائن يمكن أن يكون له مجموعة المتغيرات الخاصة به. بالإضافة إلى ذلك، يمكن للمتغيرات في سكراتش تخزين أي نوع من البيانات، مثل الأرقام أو سلاسل نصية (Strings) ويمكن تغييرها ومعالجتها في جميع أنحاء المقطع البرمجي.

أمثلة على المتغيرات



اسم المتغير

كل متغير له اسم وقيمة، فعندما تنشئ متغيرًا، فإنك تحدد اسمه، ويجب أن يكون اسم كل متغير فريدًا. يمكن أن يحتوي الاسم على أي مزيج من الأحرف الكبيرة والصغيرة، ويمكنك استخدام أكثر من كلمة واحدة مع وجود مسافات بينهما، ويفضل أن يمثل الاسم الذي تعطيه للمتغير محتواه وذلك لفهم ما يمثله داخل المقطع البرمجي.



شروط تسمية المتغير



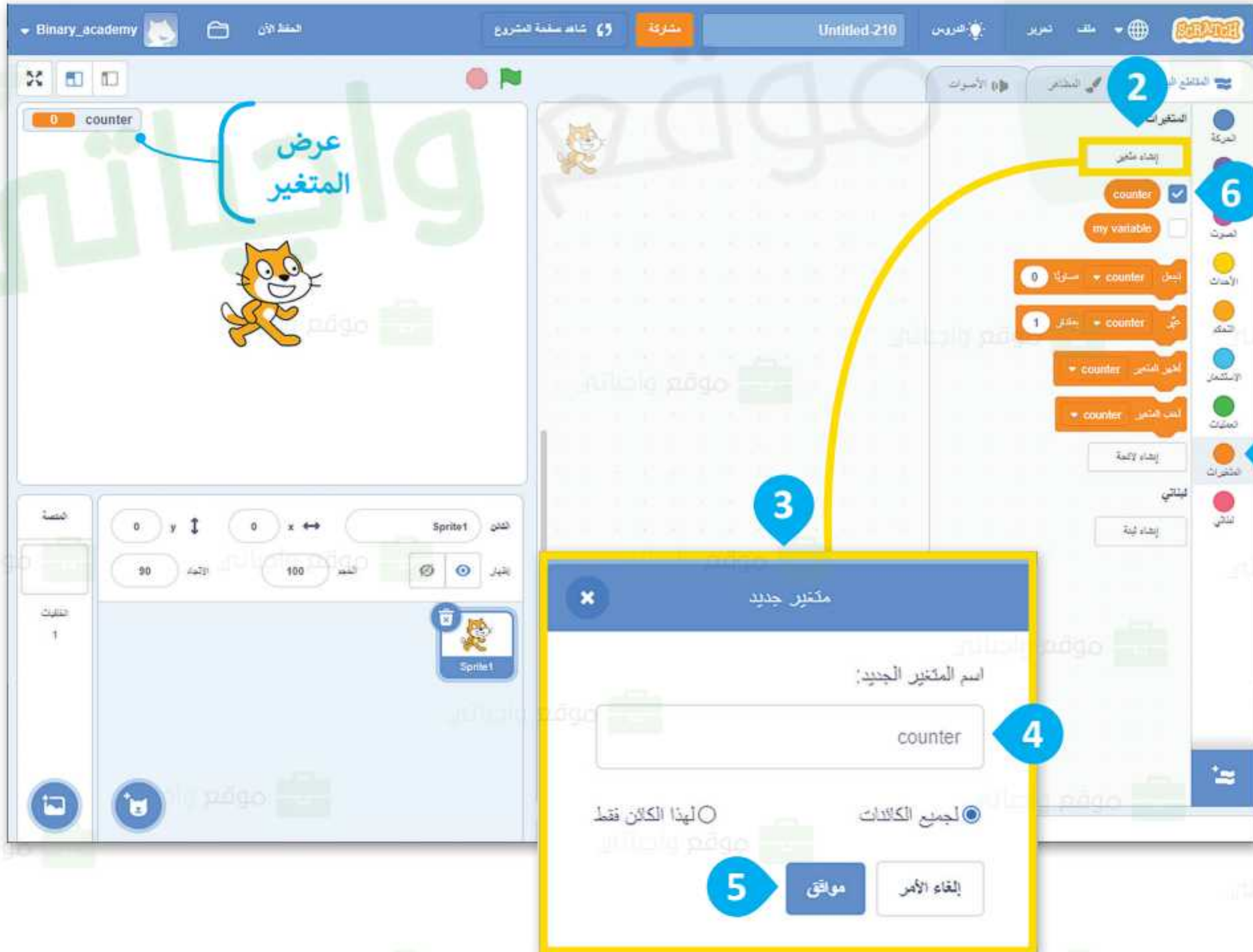
يسمح سكراتش باستخدام أسماء المتغيرات باللغتين العربية والإنجليزية. ومع ذلك، من المهم ملاحظة أن سكراتش هي في الأساس لغة برمجة قائمة على اللغة الإنجليزية، لذلك يوصى باستخدام أسماء المتغيرات باللغة الإنجليزية لتحسين التوافق مع مشاريع وموارد سكراتش الأخرى.

إنشاء متغير

كما ذكر بالدرس السابق، تجمع الدجاجة البيض في طريقها، وفي كل مرة تجمع بيضة تحصل على نقطة واحدة. أنشئ المتغير الأول الخاص بك، سمّ المتغير counter واستخدمه لحساب عدد البيض الذي تجمعه الدجاجة. ستستخدم هذا المتغير في لعبتك.

لإنشاء متغير:

- 1 < اضغط على فئة لبنات المتغيرات (Variables).
- 2 < اضغط على إنشاء متغير (Make a Variable).
- 3 < ستظهر نافذة متغير جديد (New Variable).
- 4 < سمّ المتغير "counter".
- 5 < اضغط على موافق (OK)، وسيتم إنشاء متغير جديد.
- 6 < اضغط على مربع الاختيار لتظهر قيمة المتغير على المسرح.





تهيئة متغير

عندما تريد تعيين قيمة محددة إلى متغير، يمكنك استخدام لبنة () مساوياً () to () set ().

يمكنك ضبط المتغير على أي رقم تريده.



تعديل متغير

قد تحتاج أحياناً إلى تعديل اسم المتغير، فيمكنك إعادة تسميته أو حتى حذفه.

لحذف متغير أو إعادة تسميته:

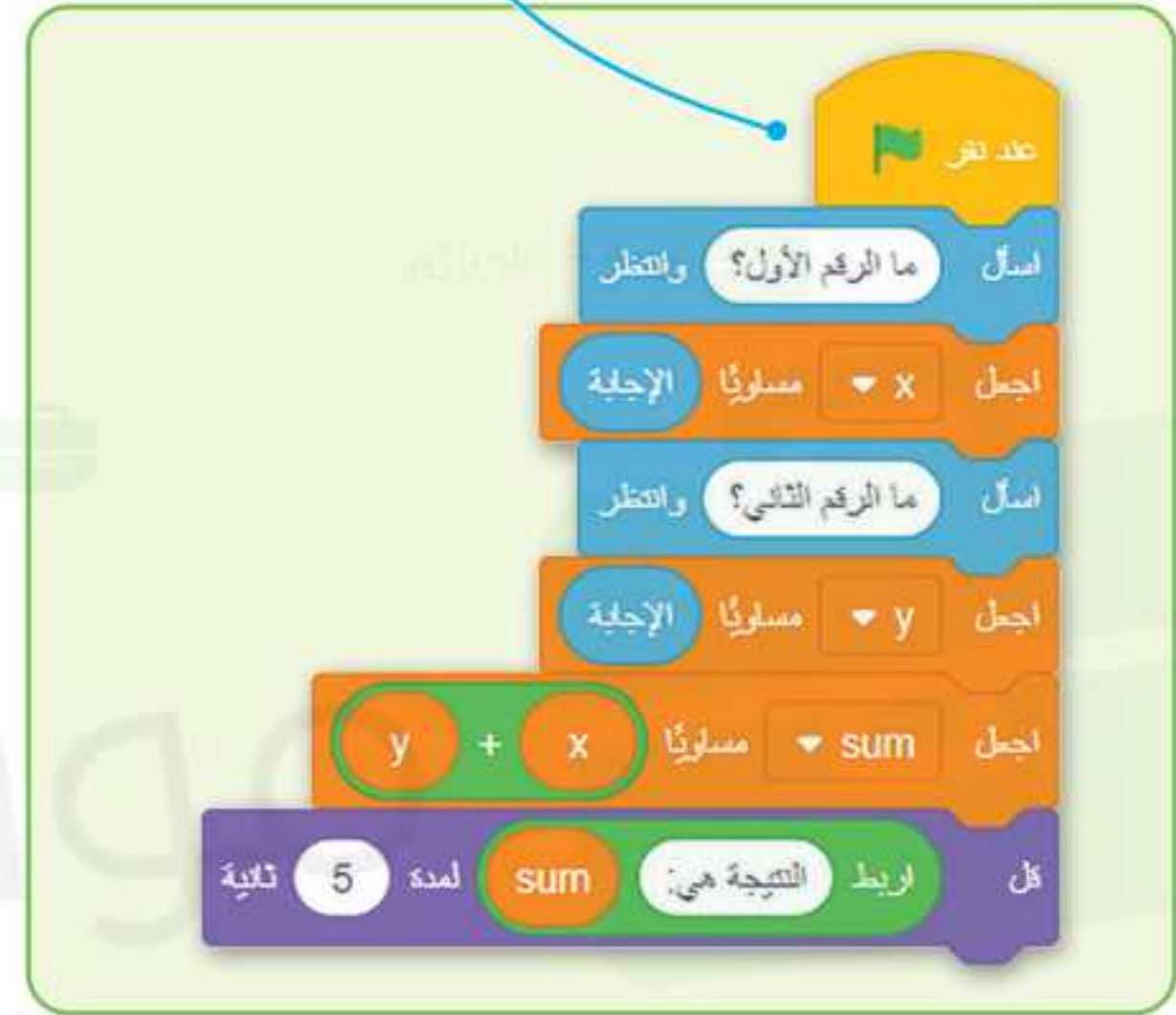
1. اضغط على فئة لبنات المتغيرات (Variables).
2. اضغط بزر الفأرة الأيمن على المتغير الذي تريد تعديله.
3. اضغط على إعادة تسمية المتغير (Rename variable) لتغيير اسم المتغير، أو حذف المتغير "counter" (Delete the variable "counter") إذا كنت تريد حذفه.
- 4.



العمليات الحسابية بالمتغيرات

في المثال الآتي، سيطلب المقطع البرمجي من المُستخدم إدخال رقمين، وسيتم تخزين كل رقم في متغير، وستخزن نتيجة عملية الجمع في متغير مختلف (sum)، وفي النهاية سيقول الكائن النتيجة.

اختبر المقطع البرمجي عن طريق كتابة أرقام مختلفة في كل مرة.



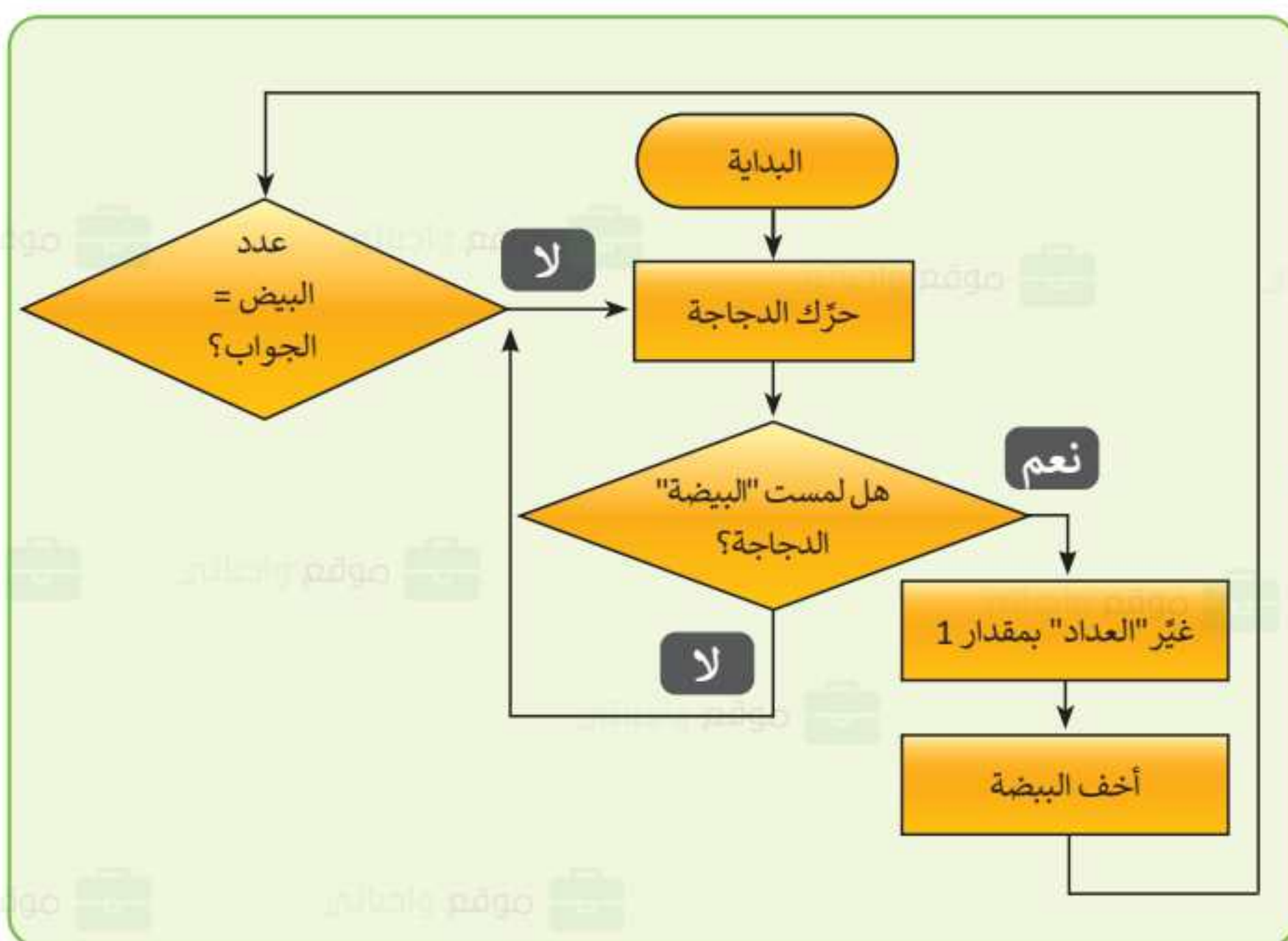
استخدام المتغير كعداد

في كل مرة تلمس دجاجة بيضة تحصل على نقطة، لذلك يجب أن تتغير قيمة المتغير بمقدار 1. للقيام بذلك، ستستخدم عامل الجمع (+)، وهو مفيد جدًا عندما تريد استخدام متغير كمقياس في لعبتك.

إنشاء المقطع البرمجي لكائن Egg

من أجل إعطاء الدجاجة نقطة واحدة عندما تلمس البيضة، عليك إضافة نص إلى الكائن Egg، فعندما تلمس الدجاجة البيضة تختفي البيضة.

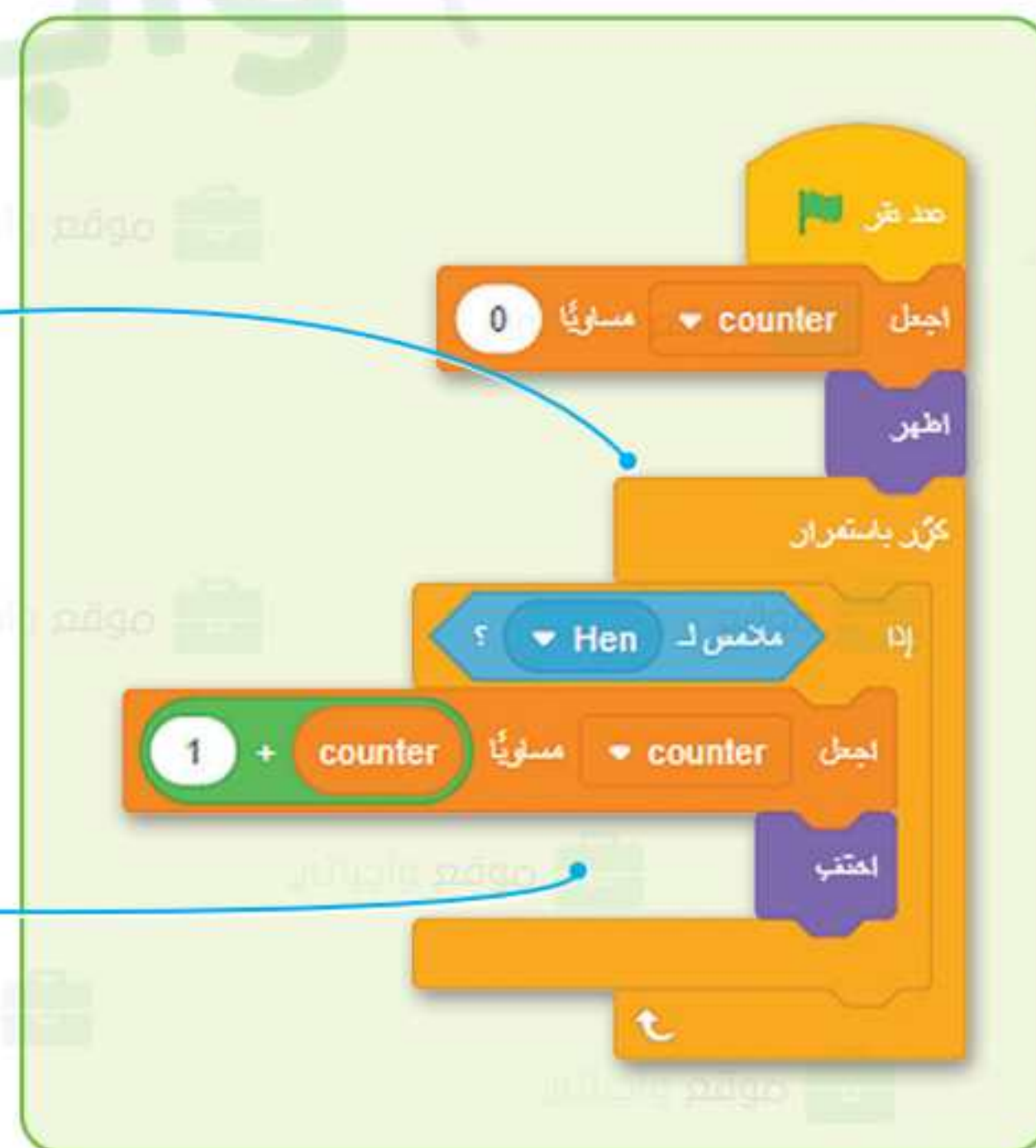
لهذا السبب ستستخدم لبنة اظهر (show) ولبنة اختف (hide)، وعادة ما يتم استخدام هاتين اللبنتين معًا، ويمكنك العثور عليهما في فئة لبنات الهيئة (Looks).



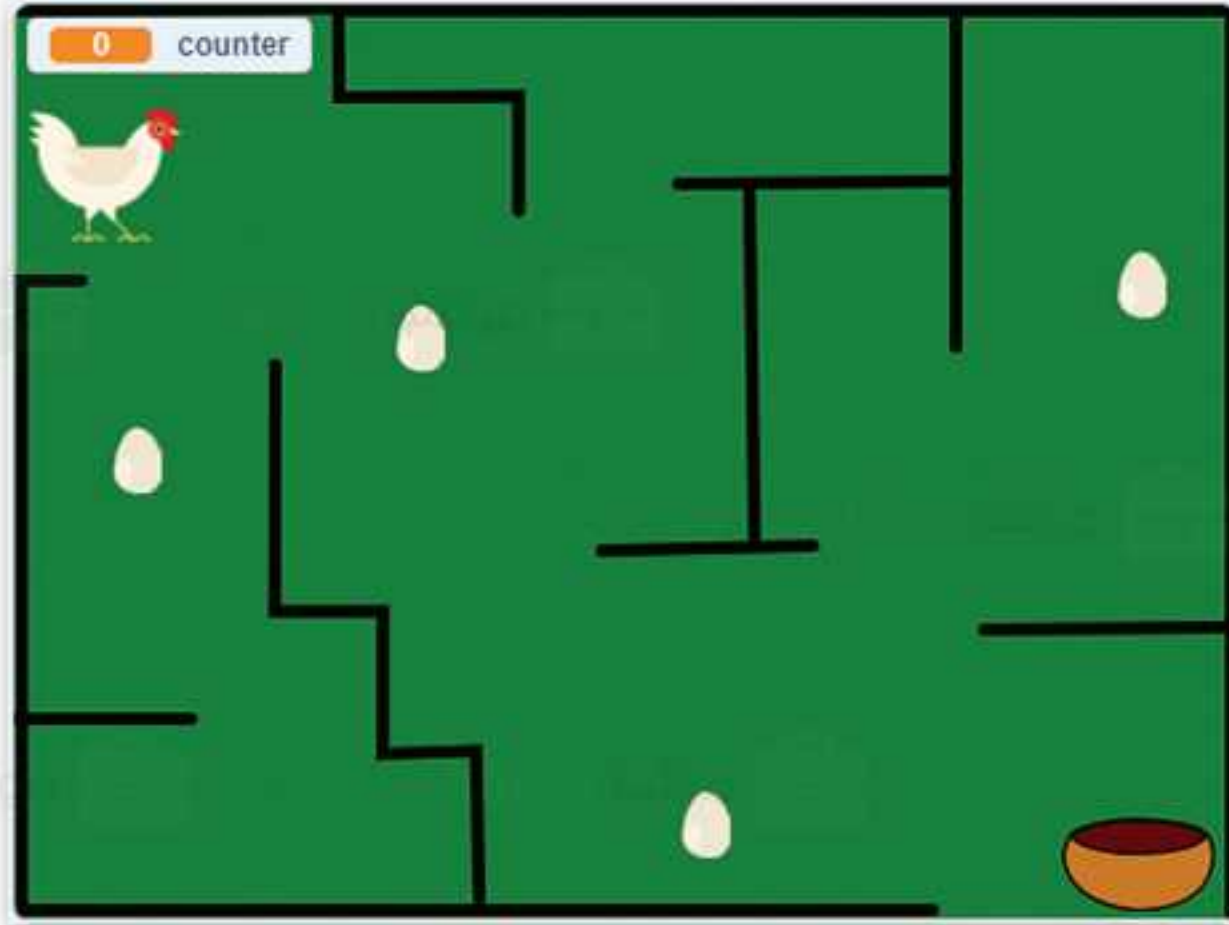
أضف النص الآتي إلى الكائن Egg.

تُظهر هذه اللبنة الكائن على المسرح.

هذه اللبنة تخفي الكائن من المسرح.



ستحتاج في لعبتك إلى أكثر من بيضة، عليك استخدام النسخ المكررة لإضافة 3 بيضات، فبهذه الطريقة ستوجد الكائنات الجديدة في المقطع البرمجي.



لبنة غيّر

ستغير هذه اللبنة قيمة المتغير المحدد إلى قيمة معينة.

يمكنك إدخال أي قيمة تريدها.

غير my variable بمقدار 5

تزيد لبنة غيّر (change) قيمة counter بمقدار 1 في كل مرة تلمس فيها الدجاجة بيضة.

غير counter بمقدار 1

هذا المقطع البرمجي مفيد للغاية عندما تريد استخدام متغير كمقياس في لعبة مثل لعبتك حيث سيتم زيادة العداد (counter) بمقدار 1 في كل مرة يلمس فيها واحدة من البيض.



لنطبق معًا

تدريب 1

المُعاملات الحسابية

نفذ العمليات الحسابية في سكراتش باستخدام المُعاملات الحسابية.

$$\begin{aligned}8 - 6 &= 2 \\ 3 * 7 &= 21 \\ 8 / 2 &= 4 \\ 9 / 3 &= 3 \\ 5 + 4 + 4 &= 13 \\ 2 * 9 - 3 &= 15 \\ 8 / 4 + 7 &= 9\end{aligned}$$

تدريب 2

مُعاملات المقارنة

اكتشف القيم الصحيحة للمتغيرات في نهاية المقطع البرمجي.

$$x = 3, y = 4$$



$$x = 5, y = 4$$



$$x = 6, y = 5$$



تدريب 3

المتغيرات

فكر في أسماء لهذه المتغيرات ثم اكتب أمام كل متغير اسمه.

سعر

سعر زجاجة الحليب.

درجة

درجة الطالب في المادة.

عدد الأسرة

عدد أفراد الأسرة.

مبلغ الفاتورة

إجمالي قيمة الفاتورة.

سعر اللوحة

تكلفة صنع لوحة جدارية.

تدريب 4

المعاملات والمتغيرات

أنشئ خوارزمية ومقطعًا برمجيًا لحساب عمرك.

● سيطلب المقطع البرمجي سنة ميلادك.

● سيطلب السنة الحالية.

● سيجري العملية الحسابية.

● يقول الكائن النتيجة.



خطوات الخوارزمية:

1. يبدأ البرنامج عند النقر على العلم الأخضر.
2. يسأل البرنامج عن سنة ميلادك.
3. تخزين الإجابة في المتغير.
4. يسأل البرنامج عن السنة الحالية.
5. تخزين الإجابة في المتغير.
6. يقوم البرنامج بطرح سنة عمرك من السنة الحالية.
7. يقول الكائن النتيجة.



الدرس الثالث: اتخاذ القرارات

في برمجة جهاز الحاسب، يؤدي صنع القرار دورًا مهمًا تمامًا كما هو الحال في الحياة الواقعية، حيث تتوفر خيارات مختلفة بناءً على ظروف مختلفة. على سبيل المثال، إذا كان الجو باردًا في الخارج، فستختار ارتداء معطف، وإذا كان الجو دافئًا، فقد تختار خلعه. في سكراثش، يرتبط تحديد خيار واتخاذ قرار ارتباطًا وثيقًا، حيث إن تحديد خيار غالبًا ما يتضمن اتخاذ قرار بناءً على شروط معينة. توجد العديد من الحالات التي يجب فيها الاختيار بين خيارين أو أكثر حسب الشرط. تستخدم عملية اتخاذ القرار في البرمجة لترتيب تنفيذ الأوامر.

لبنة إذا () وإلا

عندما يكون عليك اتخاذ قرار وترغب في تحديد ما يحدث عندما يكون الشرط صحيحًا أو خطأ، يمكنك استخدام لبنة إذا () وإلا (if () then , else). في هذه اللبنة إذا كان الشرط صحيحًا، فيتم تنفيذ اللبنة الموجودة تحت إذا (If)، وإذا كان الشرط خطأ، فيتم تنفيذ اللبنة الموجودة أسفل إلا (else).

يمكنك العثور على لبنة إذا () وإلا في فئة لبنات التحكم (Control).

اللبنة التي يتم تنفيذها
إذا كان الشرط صحيحًا.

اللبنة التي يتم تنفيذها
إذا كان الشرط خطأ.

استخدام لبنة إذا ().

استخدام لبنة إذا () وإلا.

ألق نظرة على المثال الآتي. إذا كانت نتيجة الطالب أقل من 50، فعليه المحاولة وكتابة النتيجة مرة أخرى، وفي الحالة الأخرى يجتاز الاختبار.

هذان المقطعان البرمجيان لهما نفس النتيجة.

هل يمكنك تمييز الاختلاف بينهما؟ أي مقطع برمجي تفضله؟

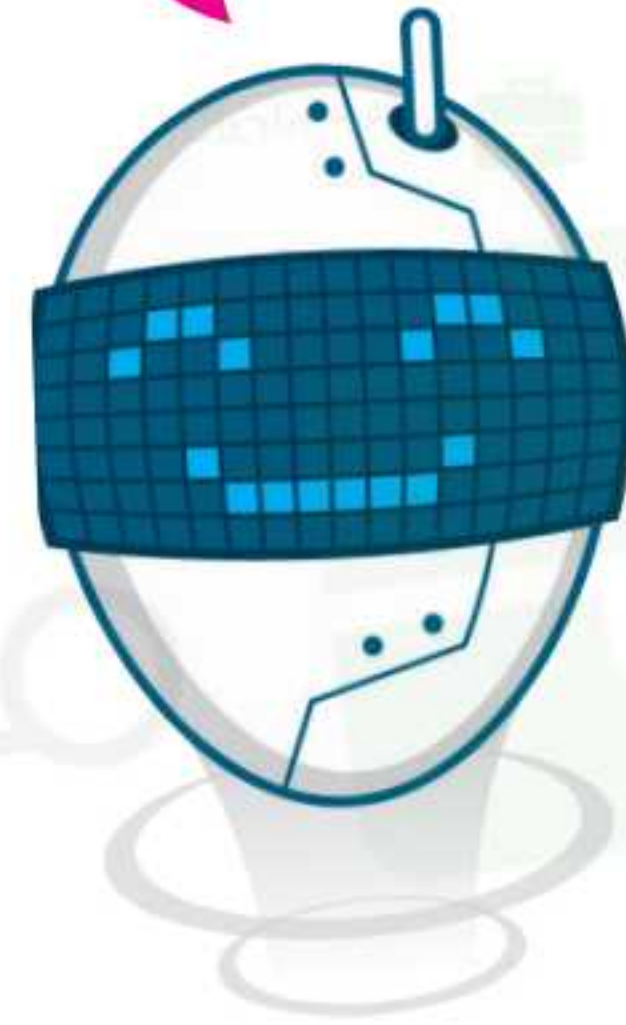


إنشاء مقطع برمجي لاتخاذ قرار

أنشئ مقطعًا برمجيًا أكثر صعوبة. في هذا المقطع البرمجي سيسأل الكائن إذا كنت تريد أن يلتف يمينًا أو يسارًا، وبناءً على إجابتك سيلتف الكائن ثم سيتوقف الكائن عن الالتفاف عندما تضغط على مفتاح "w" أو "m" من لوحة المفاتيح. على وجه التحديد، سوف يلتف الكائن إلى اليمين عند كتابة 'يمين' وسيستمر في الالتفاف إلى اليمين حتى يتم الضغط على مفتاح 'm'، بينما إذا كتبت أي كلمة أخرى غير كلمة 'يمين' سوف يلتف إلى اليسار ويستمر في الالتفاف إلى اليسار حتى يتم الضغط على مفتاح 'w'.



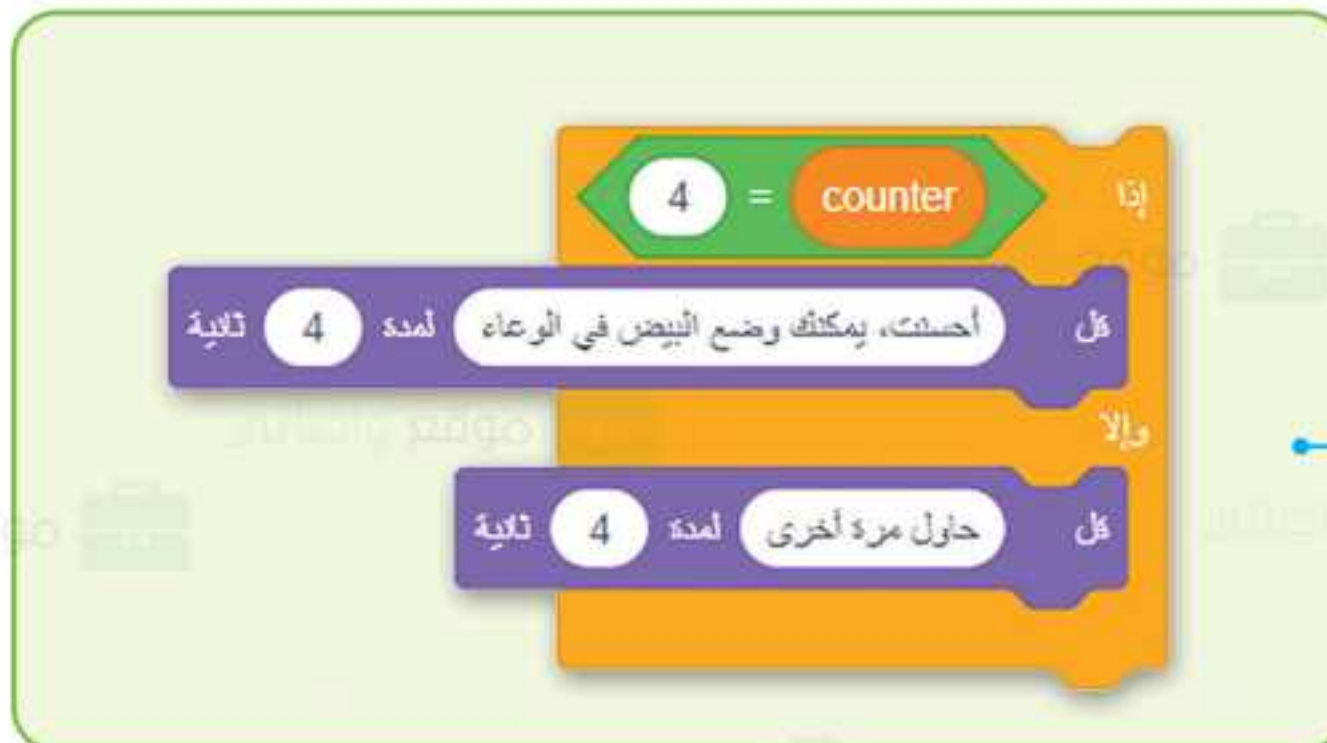
قبل تشغيل المقطع البرمجي، غير لغة لوحة المفاتيح إلى اللغة الإنجليزية.



جرب بنفسك

في الدرس السابق أنشأت متغيرًا ليحصى عدد البيض الذي جمعته الدجاجة، ويوجد في اللعبة 4 بيضات. ستضيف الآن لينة التحكم الجديدة التي تعلمتها في هذا الدرس. إذا جمعت الدجاجة كل البيض ولمست الوعاء، ستظهر رسالة "أحسن، يمكنك وضع البيض في الوعاء"، وإلا ستظهر رسالة "حاول مرة أخرى".

ستضيف الآن اللبنات الآتية إلى المقطع البرمجي للكائن الدجاجة:

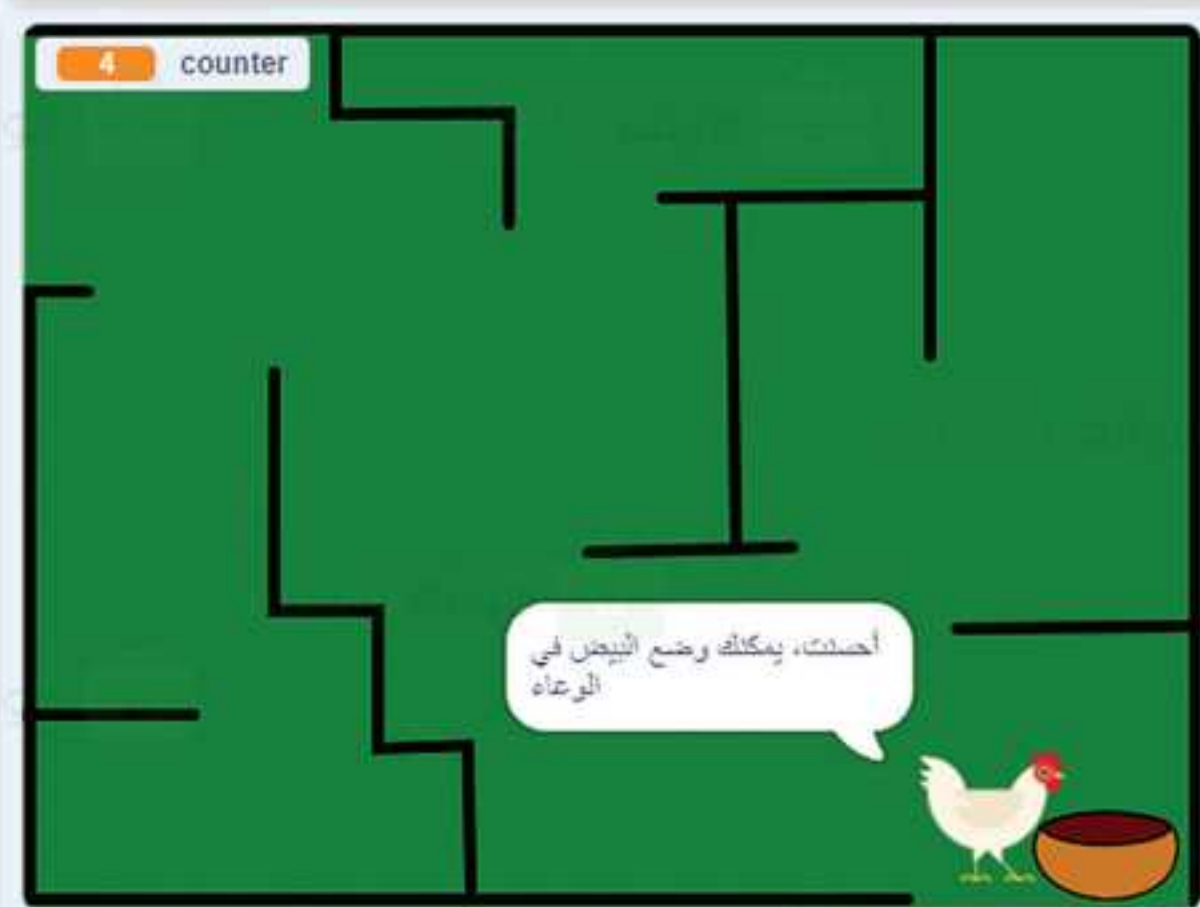
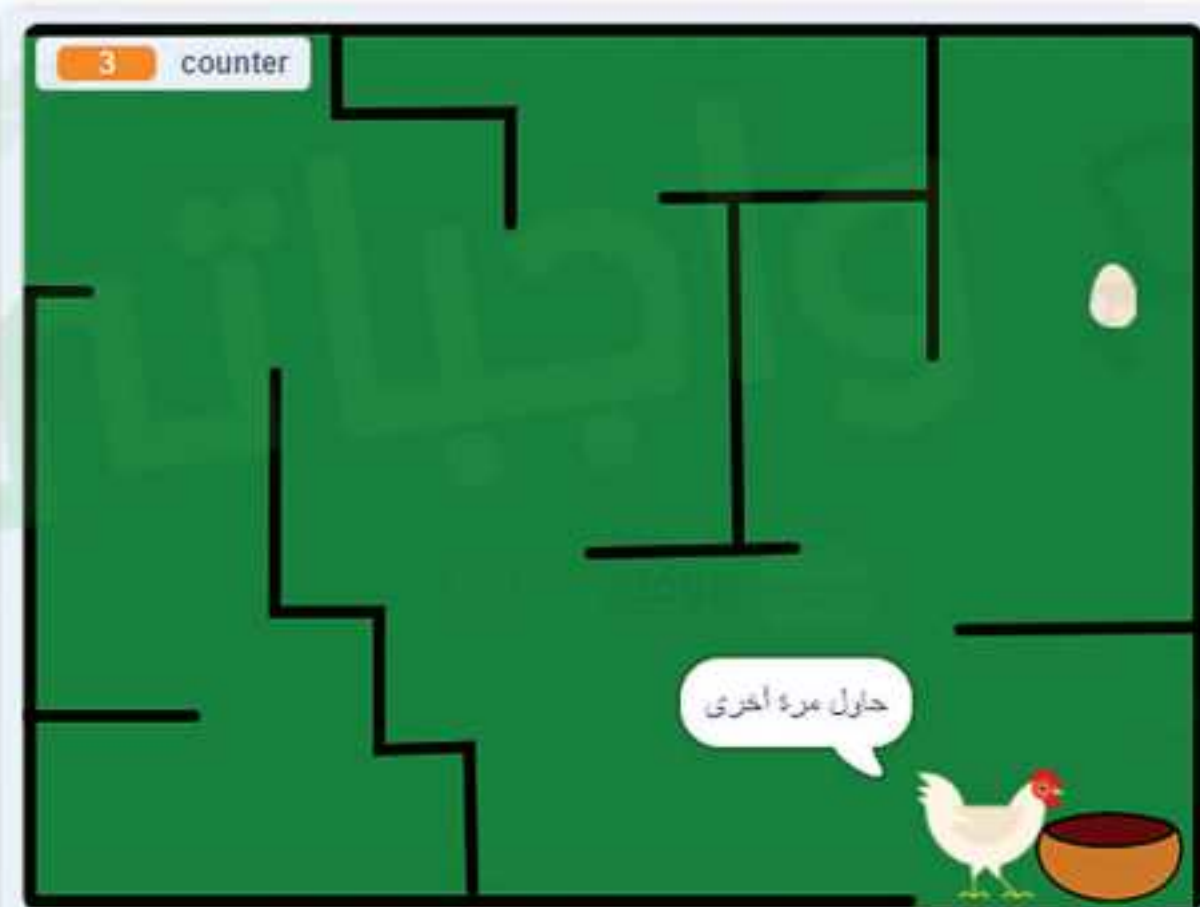
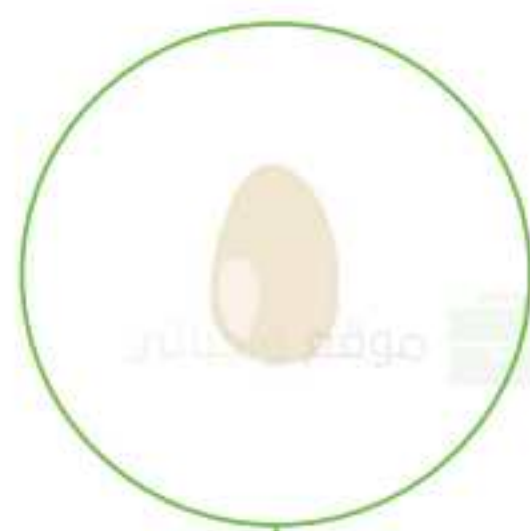


تحدد هذه العبارة الشرطية ما إذا كانت الدجاجة قد جمعت كل البيض.



مخطط انسيابي للعبة





لنطبق معًا

تدريب 1

كتابة مقطع برمجي

جَرِّب هذا المقطع البرمجي باستخدام برنامج سكراتش والذي يعرض الرقم الأكبر من رقمين يدخلهما المُستخدم.



نتيجة المقطع البرمجي هي: **الإجابة تعتمد على الطالب هذا مثال للحل**
الرقم الأول أكبر من الرقم الثاني.

x = 7
y = 5

ثم عدّل المقطع البرمجي باستخدام لبنة إذا ().

تدريب 2

إجراء العمليات الحسابية

طلب منك معلمك كتابة مقطع برمجي لحساب السعر الإجمالي الذي يجب أن يدفعه المشتري في متجر الأطعمة لأربعة عناصر من المواد الغذائية التي يشتريها:

■ يسأل المقطع البرمجي المشتري عما تدفعه في:

- حليب
- الشوفان
- عصير
- بيض

■ يجب أن يظهر السعر الإجمالي الذي يجب أن يدفعه المشتري على الشاشة.

■ قبل البدء في البرنامج عليك إنشاء خوارزمية له، وفيما يأتي بعض الخطوات السهلة المطلوبة لحل المشكلة، ولكن عليك ترتيبها:

3

احسب السعر الإجمالي.

1

اطلب من المشتري إدخال سعر كل عنصر.

4

اعرض النتيجة على الشاشة.

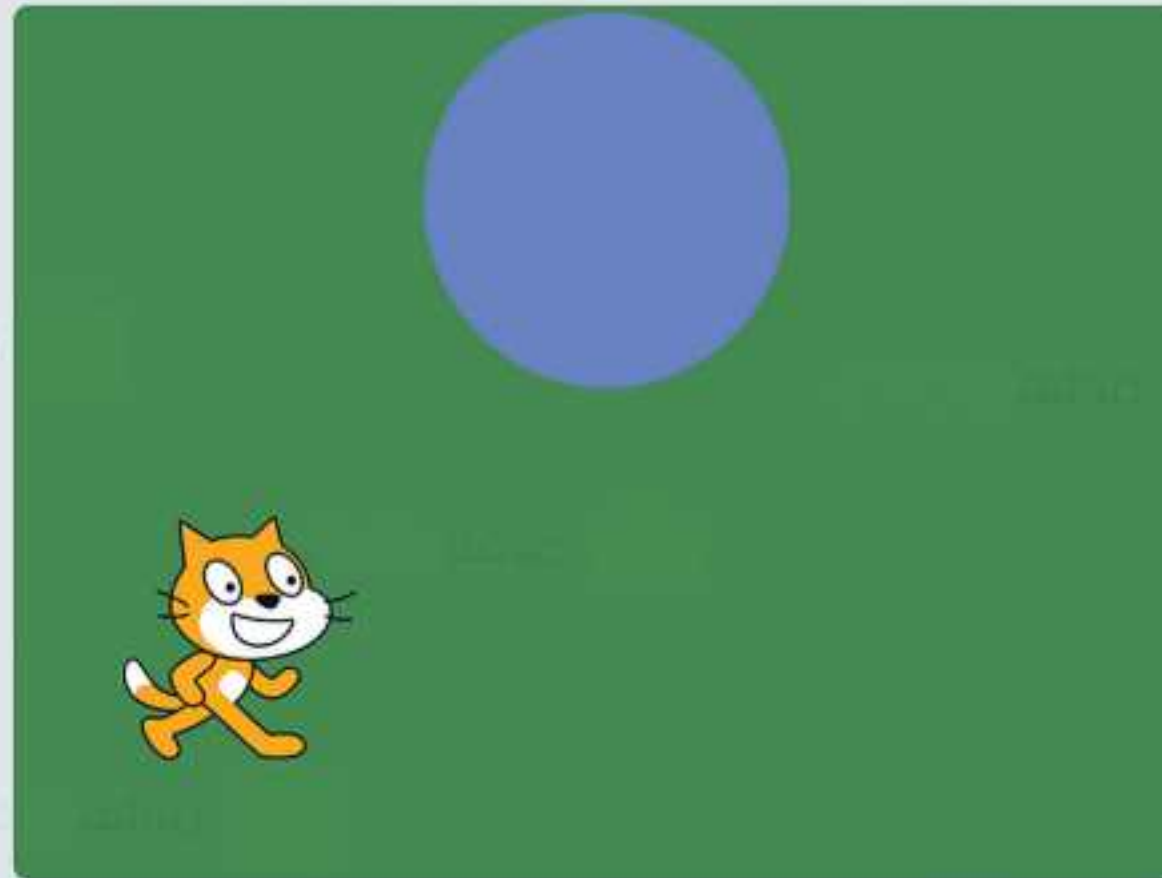
2

خزن الأسعار المُدخلة في متغيرات.

تدريب 3

اتخاذ القرار

ارسم هذه الخلفية.



ضع اللبانات بالترتيب الصحيح للحصول على النتيجة في الصورتين الأولى والثانية.



الصورة الثانية



الصورة الأولى

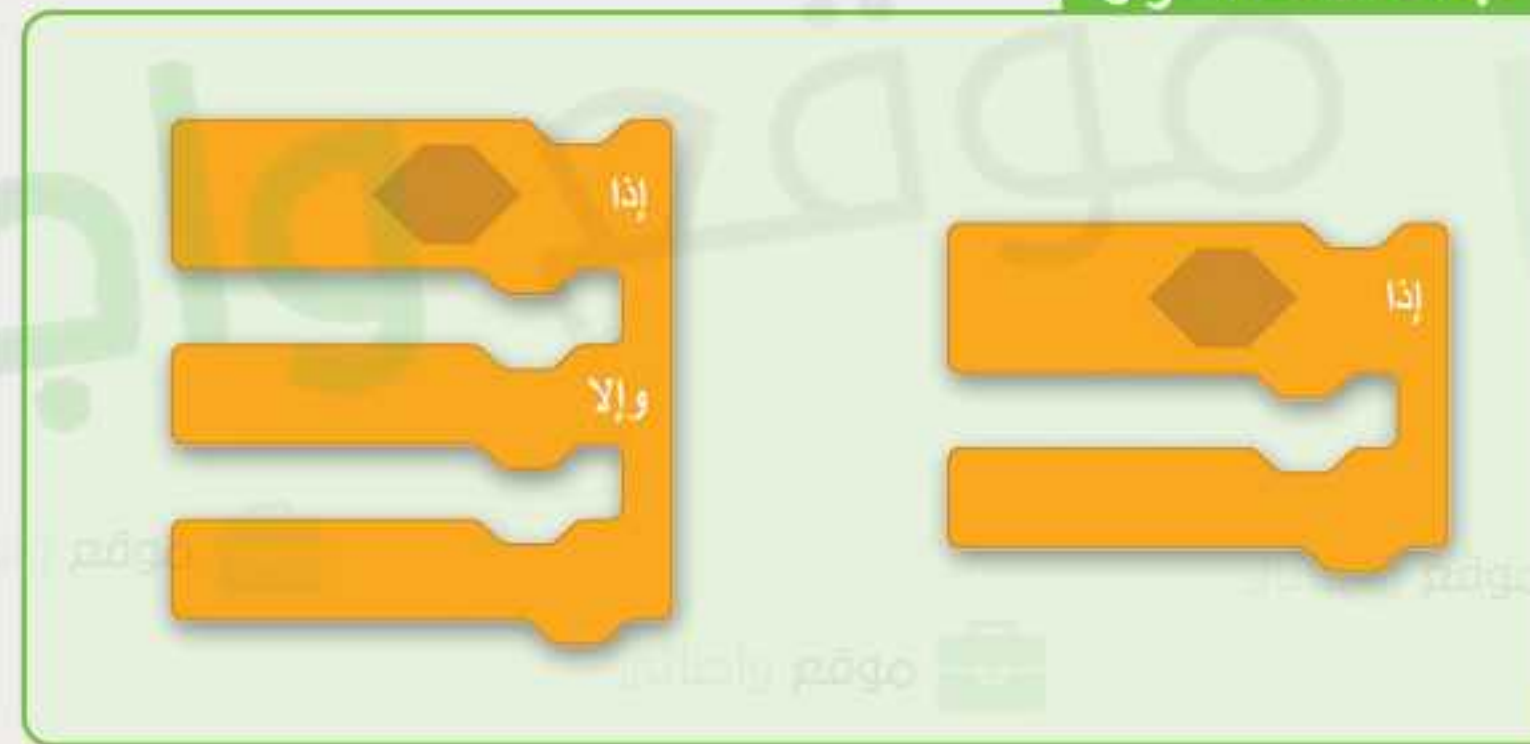
الدرس الرابع: الإحداثيات في سكراتش

لقد تعلمت سابقًا طريقة تكرار تنفيذ الأوامر بدلاً من إعادة كتابتها باستخدام لبنات التكرار (Repetition)، وإجراء العمليات الحسابية باستخدام المتغيرات واتخاذ القرارات باستخدام لبنات اتخاذ القرار.

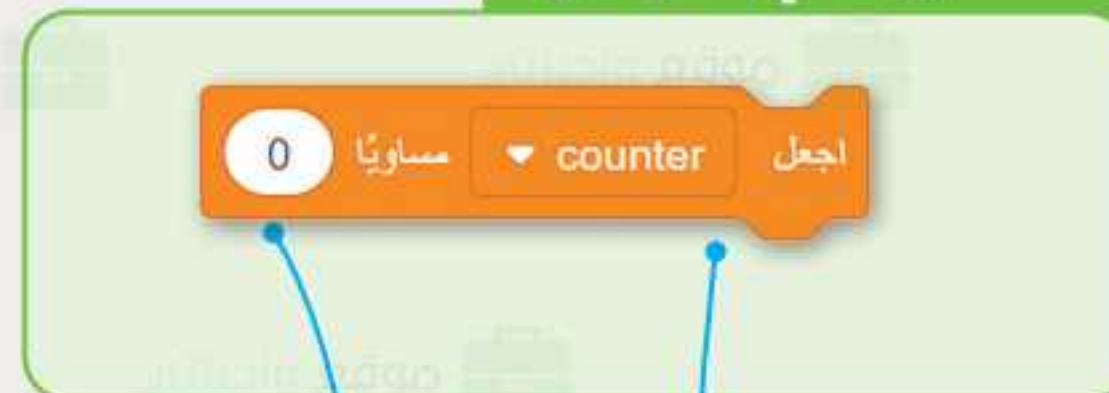
لبنات التكرار في سكراتش



لبنات اتخاذ القرار



المتغيرات في سكراتش



القيمة

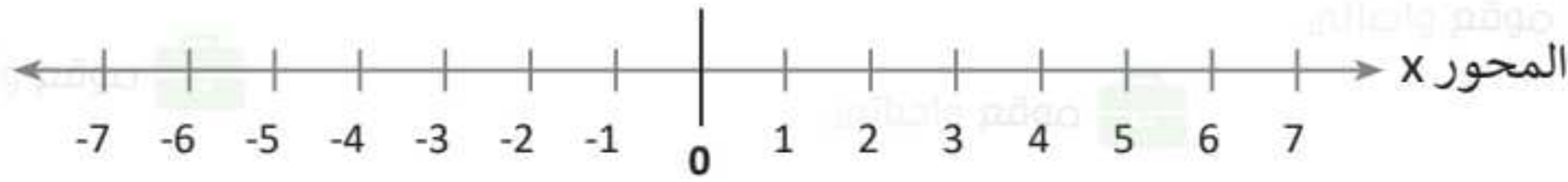
الاسم

نظام الإحداثيات

نظام الإحداثيات هو نظام يستخدم رقمًا أو عدة أرقام لتحديد موضع النقاط في مساحة محددة.

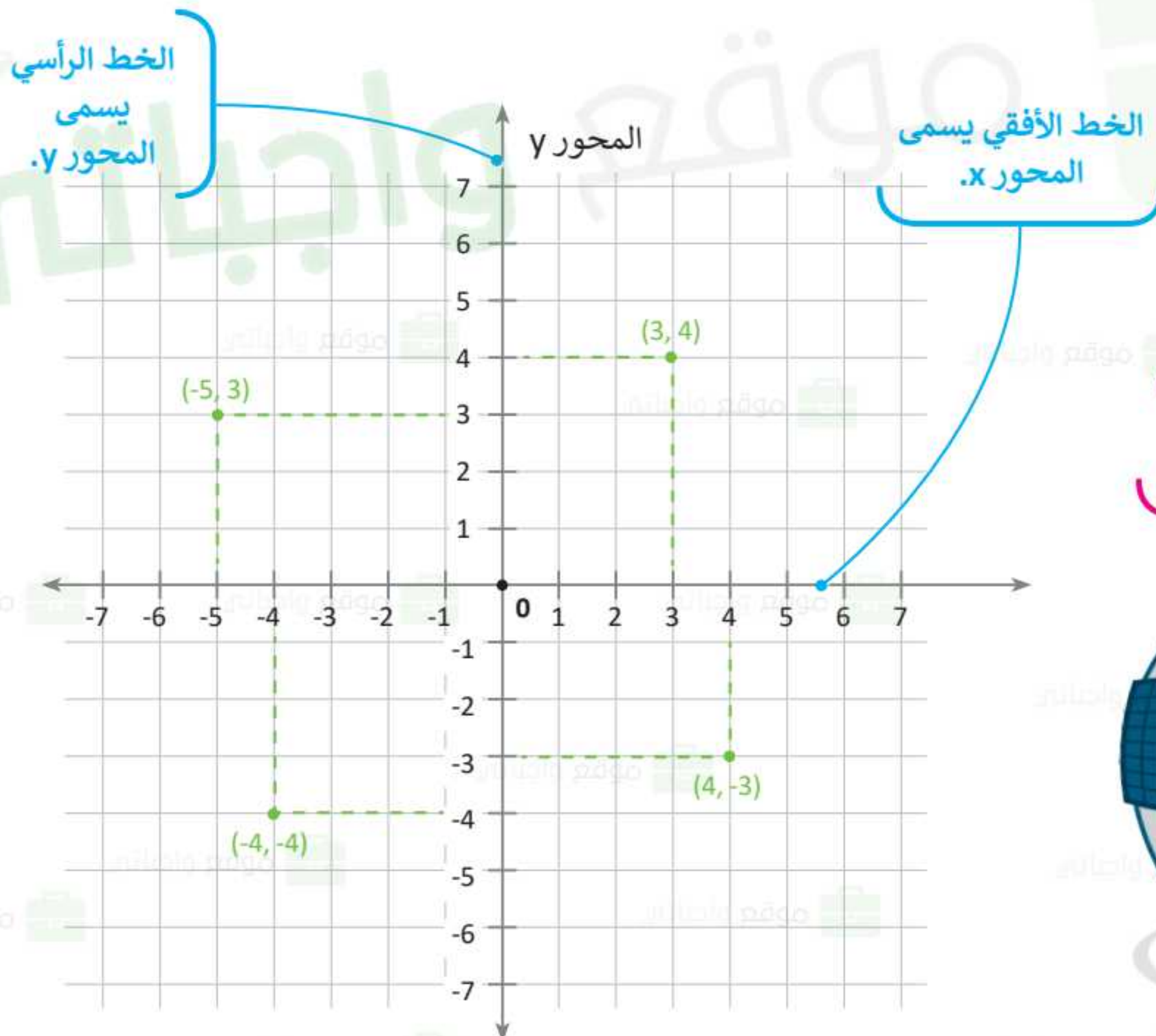
نظام الإحداثيات الخطي (Line coordinate system)

يعد نظام الإحداثيات الخطي أبسط أنواع نظام الإحداثيات، ويتكون من خط أفقي (محور واحد)، أو بُعد واحد مُرقم.



نظام الإحداثيات الديكارتي (Cartesian Coordinate System)

في نظام الإحداثيات الديكارتي يتقابل خطان بزاوية قائمة بينهما، وإحداثيات النقطة هي بُعد النقطة عن كل خط. يُطلق على كل خط اسم محور الإحداثيات ويلتقي المحوران في نقطة الأصل والتي تمثل القيمة صفر (0) لكل منهما.

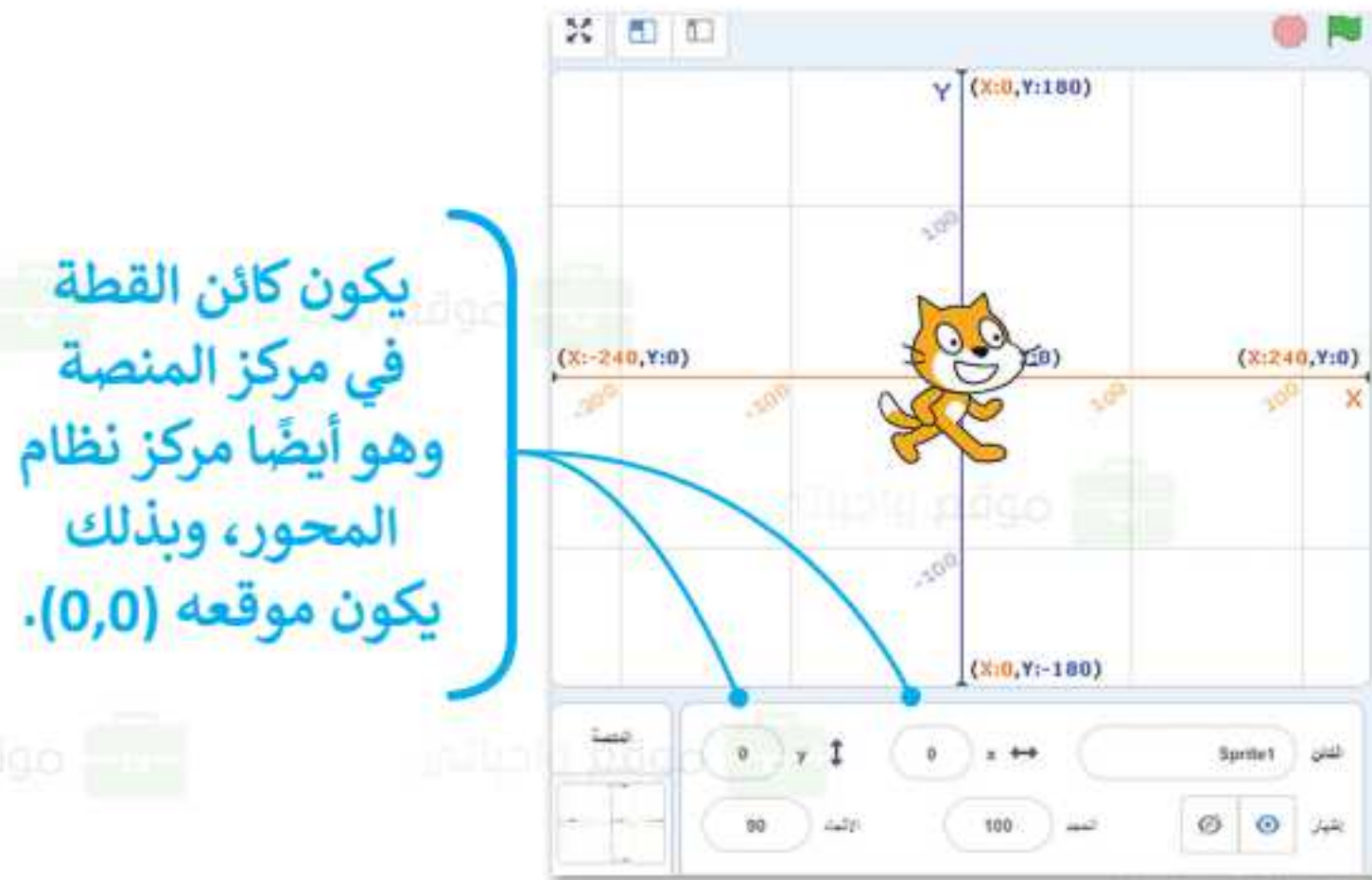


يمكنك وصف موضع نقطة باستخدام إحداثياتها.



الإحداثيات في سكراتش

تتكون المنصة في سكراتش من مجموعة من النقاط تسمى البكسل (Pixels)، فهي مثل جدول به العديد من الصفوف والأعمدة. يُشار إلى الموضع في العمود من خلال الرمز y وإلى الموضع في الصف من خلال الرمز x . يُمكنك زوج النقاط (x, y) من تحديد موقع كل بكسل في المنصة ويسمى هذا الزوج إحداثيات النقطة.



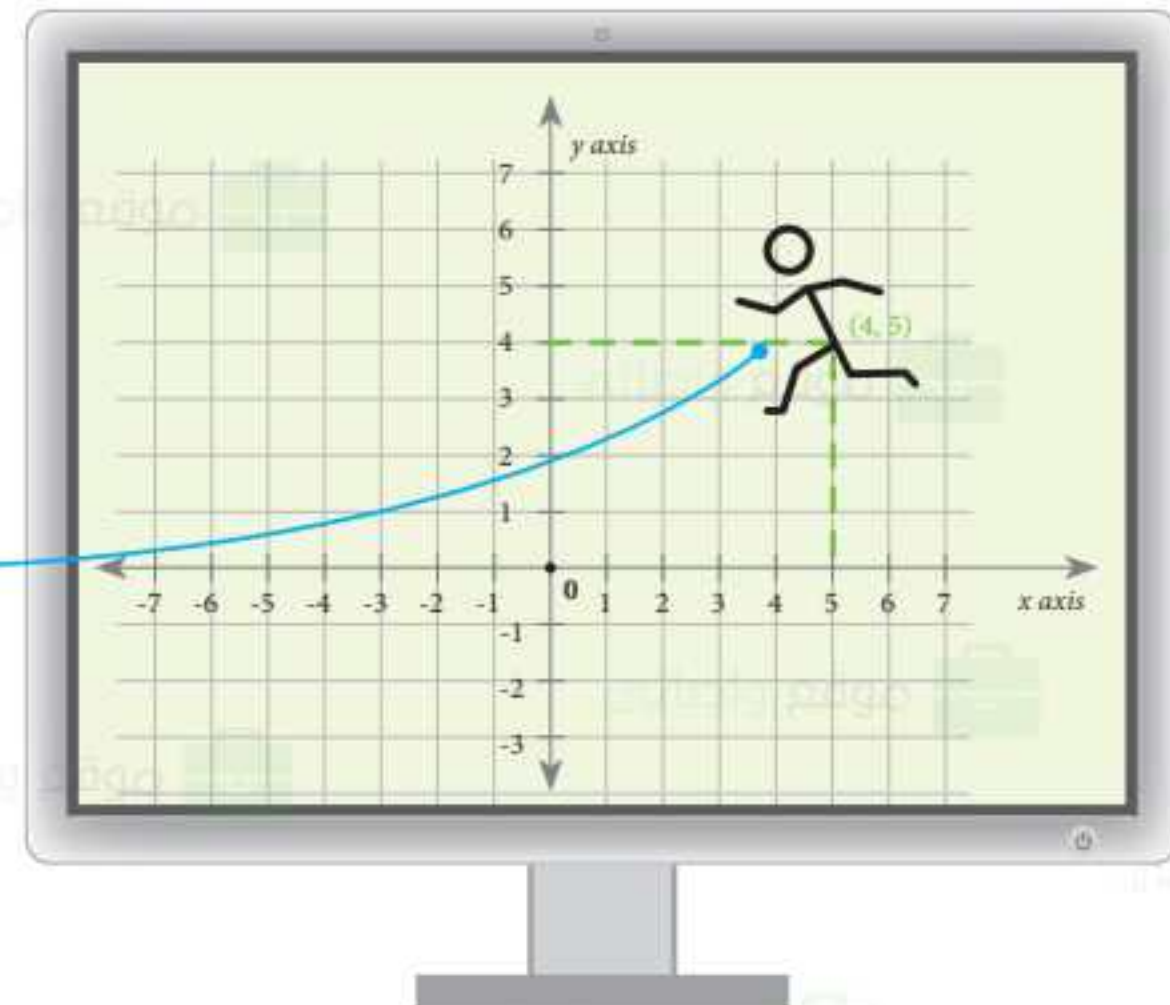
لكل كائن موقعه الخاص على الشاشة، ويتم تحديد هذا الموقع بإحداثيين (x, y) . انتبه إلى الاختلاف في هذه المرحلة بين اللغتين، باللغة العربية مكتوب (س، ص).



الإحداثيين

x	y
تشير قيمة x إلى موضع الكائن أفقيًا على طول المحور x ، ويتحرك موضع الكائن على المنصة يمينًا أو يسارًا بزيادة هذه القيمة أو إنقاصها، وبنفس الطريقة تتغير قيمة x عند تحريك الكائن أفقيًا باليد.	تشير القيمة y إلى موقع الكائن رأسيًا على طول المحور y ، ويتحرك موضع الكائن على المنصة لأعلى أو لأسفل بزيادة هذه القيمة أو إنقاصها، وبنفس الطريقة تتغير قيمة y عند تحريك الكائن عموديًا باليد.

إذا كان موضع الكائن $(5,4)$ ، فهذا يعني أن قيمة الإحداثي x هي $x=5$ وقيمة الإحداثي y هي $y=4$ ، وعندما يتحرك الكائن على المنصة تتغير إحداثيات موقع الكائن.



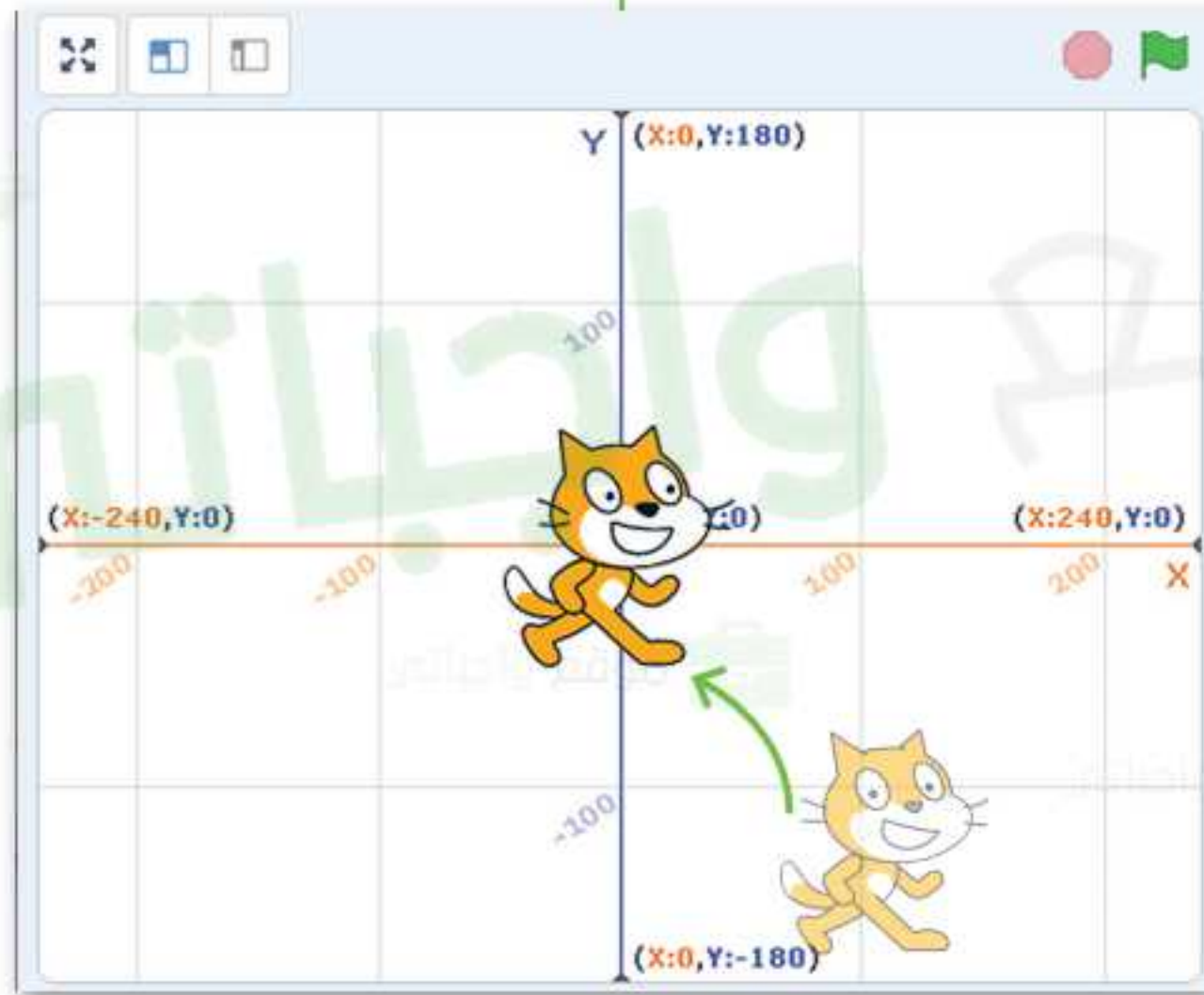
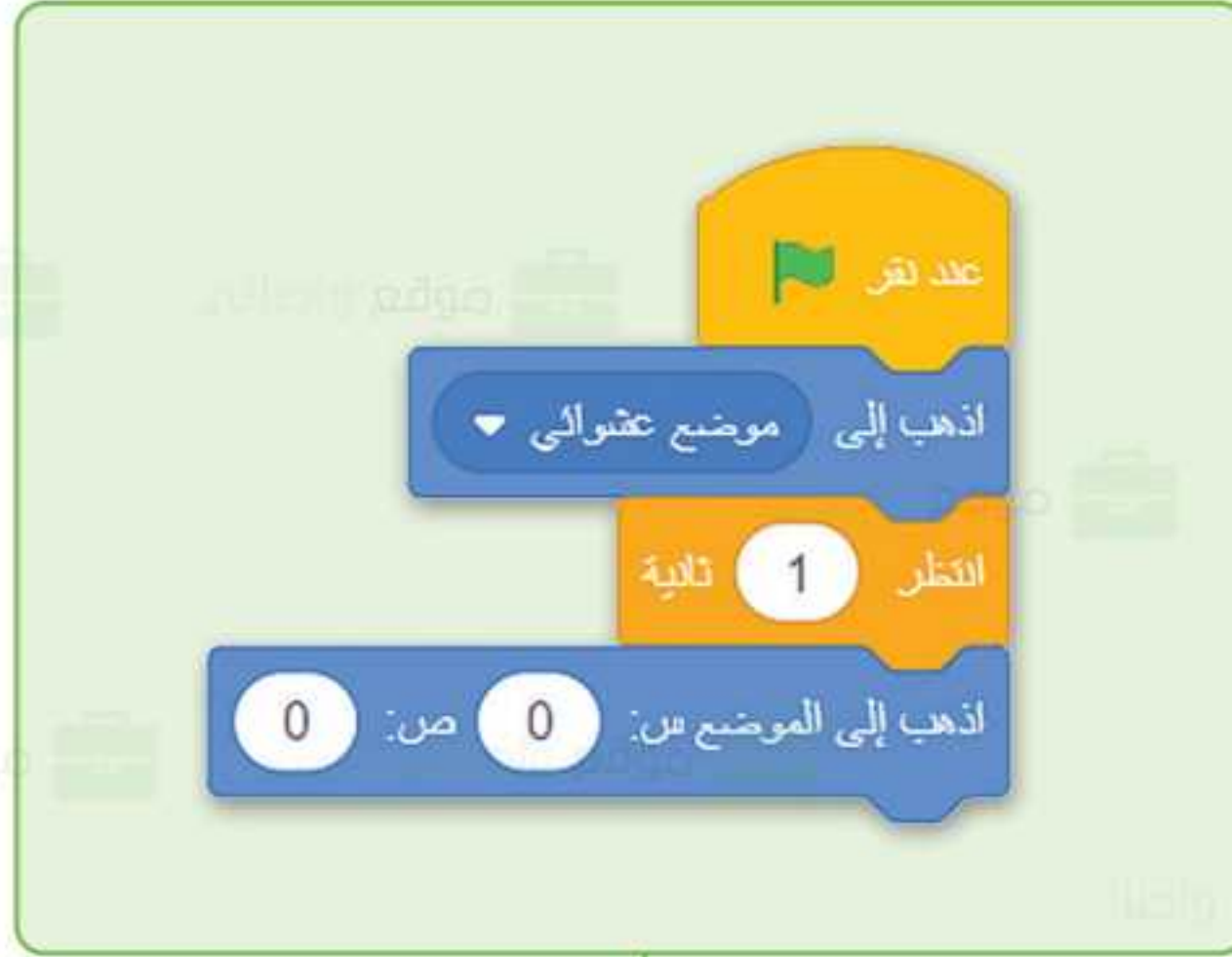
لبنة تغيير الإحداثيات

الوصف	اللبنة
لبنة اذهب إلى (موضع عشوائي) (go to (random position)) تنقل الكائن إلى موقع عشوائي على المنصة أو إلى مؤشر الفأرة.	
لبنة اذهب إلى الموضع س: () ص: () (go to position x: () y: ()) تنقل الكائن إلى موقع الإحداثيات المحددة.	
لبنة انزلق خلال () ثانية إلى الموضع س: () ص: () (glide () secs to x: () y: ()) تجعل الكائن يتحرك بسلاسة إلى موقع الإحداثيات x و y ، في عدد محدد من الثواني.	
لبنة غيّر الموضع س بمقدار () () (change x by ()) تغيير إحداثيات x الكائن وفقًا لقيمة الصندوق الأبيض. إذا كانت القيمة التي يتغير بها الإحداثي x موجبة، يتحرك الكائن إلى اليمين، وإذا كانت سالبة يتحرك إلى اليسار.	
لبنة غيّر الموضع ص بمقدار () () (change y by ()) تغيير إحداثيات y الكائن وفقًا لقيمة الصندوق الأبيض. إذا كانت القيمة التي يتغير بها الإحداثي y موجبة يتحرك الكائن لأعلى، وإذا كانت سالبة يتحرك لأسفل.	
باستخدام لبنة اجعل الموضع ص مساويًا () () (set y to ()) تعيين إحداثيات الكائن على طول المحور y وفقًا لقيمة المربع الأبيض.	
باستخدام لبنة اجعل الموضع س مساويًا () () (set x to ()) تعيين إحداثيات الكائن على طول المحور x وفقًا لقيمة المربع الأبيض.	
تمثل قيمة الموضع ص () () على طول المحور y للمرحلة.	
تمثل قيمة الموضع س () () على طول المحور x للمرحلة.	



تحريك الكائن

يمكنك تحريك الكائن إلى أي موضع على المنصة في سكراتش باستخدام لبنات تغيير الإحداثيات ومنها: لبنة اذهب إلى الموضع س: () ص: () () () أو لبنة انزلق خلال () ثانية إلى الموضع س: () ص: () () () (glide () secs to x: () y: ()) أنشئ المقطعين البرمجين الآتيين ولاحظ الفرق:



هل يمكنك معرفة الحالة التي يُستخدم فيها كل مقطع برمجي؟



الرسوم التوضيحية في سكراتش

الرسوم التوضيحية (pictograph) هو رسم تخطيطي مكون من صور لأشياء مختلفة تُستخدم لتمثيل معلومات مختلفة. هذا النوع من المخططات مفيد عندما تحتاج إلى مقارنة بين قيم مختلفة لشيء واحد.

لإنشاء رسم توضيحي في سكراتش عليك أولاً رسم المحور أو الجدول الذي سيتم ملؤه بالأشكال الخاصة بك، ثم عليك وضع الكائن بجانب كل قيمة على المحور أو في الجدول وطباعة العدد الصحيح للأشكال.

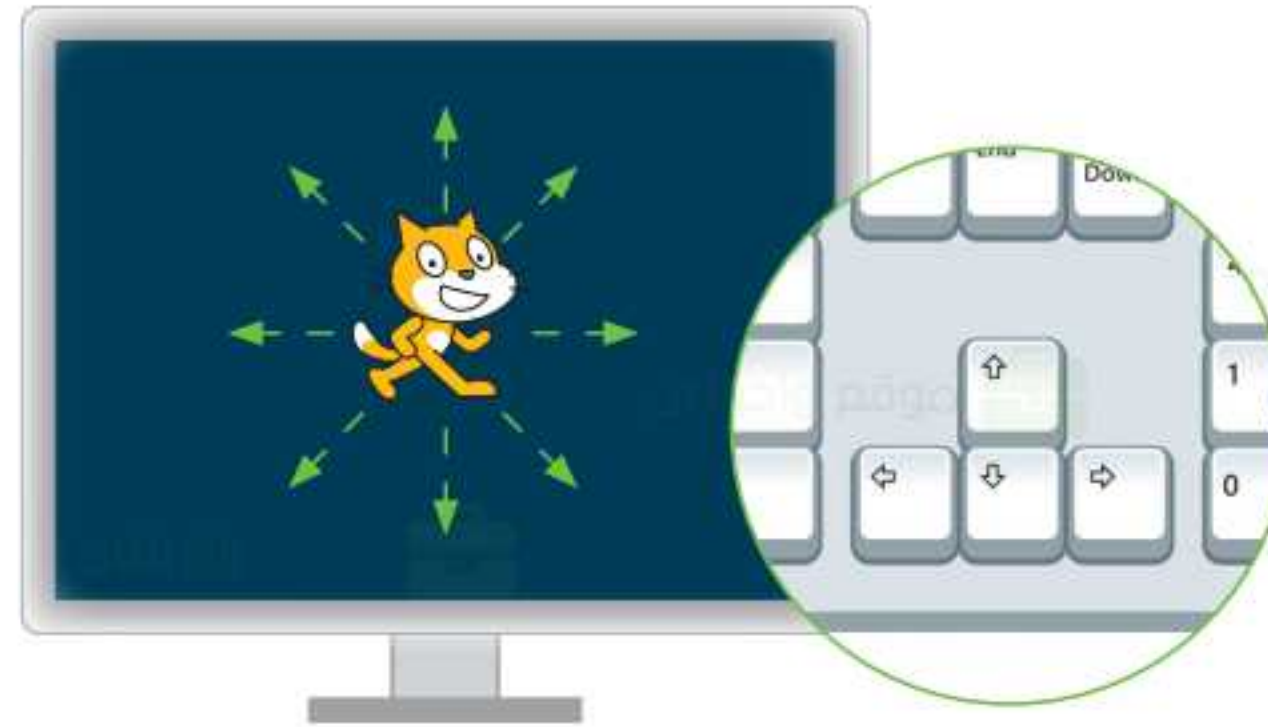
في هذا المثال يطلب المقطع البرمجي درجات الطلبة في اختبار الرياضيات ويستخدمها لإنشاء رسم توضيحي. يوضح الجدول أدناه تصنيف الطلبة إلى مجموعات وفقاً لدرجة الاختبار الخاصة بهم.

الدرجة	عدد الطلبة
70	1
75	2
80	2
85	4
90	6
95	8
100	5



التحكم في كائن باستخدام لوحة المفاتيح

الطريقة الأكثر استخدامًا لتحريك الكائنات في الألعاب هي استخدام لوحة المفاتيح، وفي سكراتش توجد طريقتان للتحكم الكامل في حركة الكائن باستخدام المفاتيح. الطريقة الأولى: استخدام لبنة عند ضغط مفتاح () (when () key pressed)، والثانية: استخدام لبنة المفتاح () مضغوط؟ (key () pressed?) في المقطع البرمجي الخاص بك.



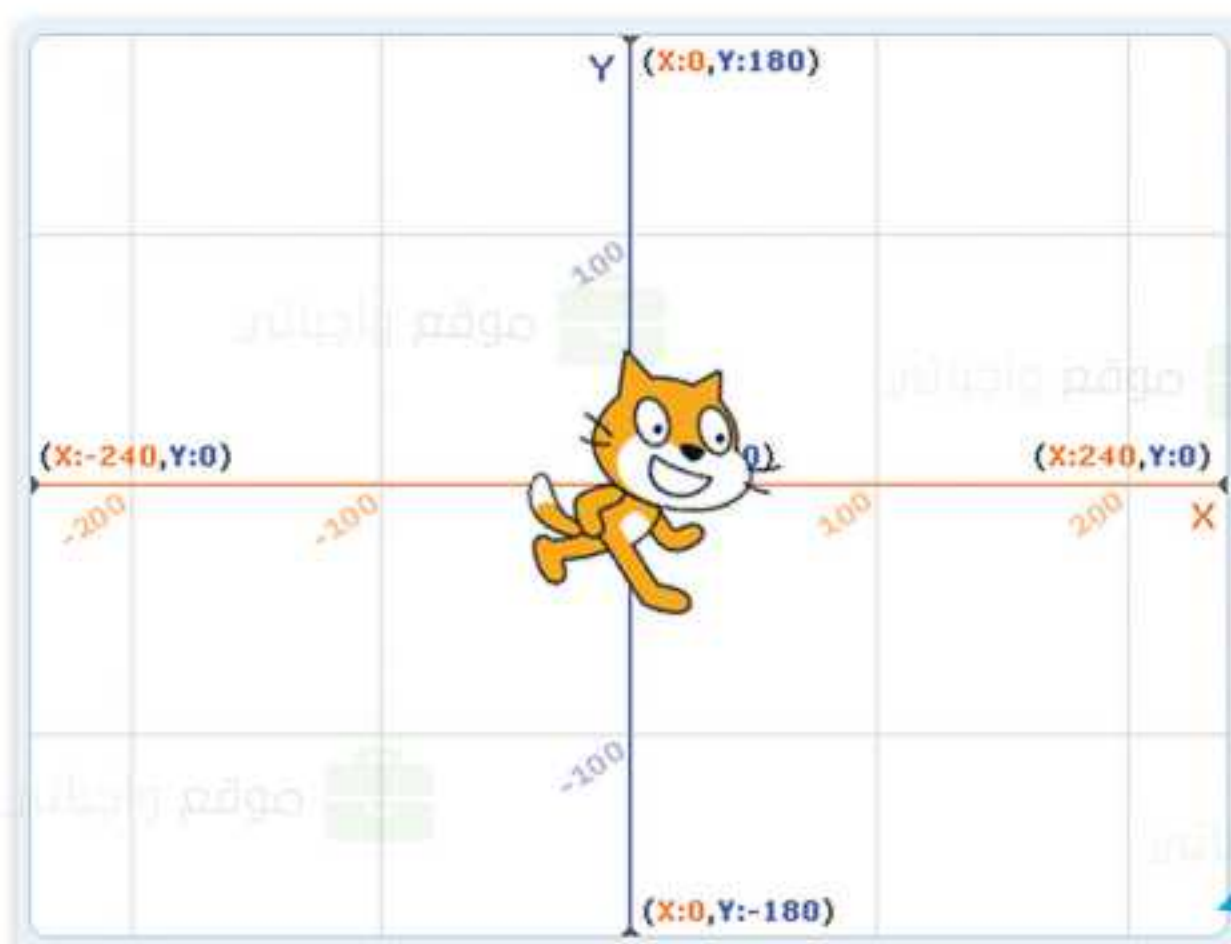
لبنة عند ضغط مفتاح () (when () key pressed)

عند ضغط مفتاح () (when () key pressed) هي لبنة من فئة لبيئات الأحداث (Events) مفيدة للغاية للتحكم في مقطع برمجي كامل من التعليمات البرمجية الخاصة بك. يتم استخدام هذه اللبنة للتحكم في الأحداث وفق مُدخلات المُستخدم، هناك مجموعة متنوعة من الاستخدامات مثل: التحكم في الكائن عن طريق كتابة حرف أو رقم أو كلمة محددة.

لن يتم تنشيط المقطع البرمجي الموجود أسفل هذا اللبنة إلا عند الضغط على المفتاح المحدد.

مرّر لأسفل لاختيار أي مفتاح من لوحة المفاتيح.





في هذا المثال، سيستدير الكائن إلى اليمين عند الضغط على مفتاح السهم الأيمن.



لبنة المفتاح () مضغوط (key () pressed)

يتم استخدام لبنة المفتاح () مضغوط للكشف عندما يتم الضغط على مفتاح معين على لوحة المفاتيح. حتى المفتاح المحدد في اللبنة يتم الضغط عليه، سيتم تنفيذ أي رمز مرفق. هذه اللبنة مفيدة جدًا عند إنشاء لعبة، حيث يحرك اللاعب الشخصية الرئيسة باستخدام مفاتيح الأسهم أو إنشاء رسم متحرك بحيث تنفذ الشخصية إجراءً معينًا عند الضغط على مفتاح معين. شاهد الفرق بين هذه اللبنات:

المقطع البرمجي الثاني

المقطع البرمجي الأول

يتم استخدام المقطع البرمجي الثاني بشكل متكرر للحركة من خلال لبنة المفتاح () مضغوط (key () pressed) وذلك لأنه يحرك الكائن بشكل أسرع ويعطي إحاءًا بالحركة.



لنطبق معًا

تدريب 1

الإحداثيات في سكراتش

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخطأ فيما يلي:
		1. تحدد قيمة y موقع الكائن على المحور الأفقي.
		2. إذا كانت إحداثيات موقع الكائن x و y تساويان صفرًا فإن الكائن يوجد في مركز المنصة.
		3. يمكنك العثور على لبنة عند ضغط مفتاح () () (when key pressed) داخل لبنات الحدث (Event).
		4. يُمكن نقل الكائن إلى موقع عشوائي على المنصة.
		5. يُمكن توضيح البيانات عن طريق إنشاء الرسوم التوضيحية (pictograph) في سكراتش.

تدريب 2

لبنة التحكم في الإحداثيات



صل الوصف باللبنة المناسبة.

1

تنقل الكائن إلى موقع الإحداثيات (-22,-15).

2

تنقل الكائن إلى موقع مؤشر الفأرة.

3

تحرك الكائن إلى أعلى.

4

تحرك الكائن إلى اليسار.

5

تعرض إحداثيات الكائن.

5

الموضع من

الموضع من

2

اذهب إلى موضع عشوائي

اذهب إلى مؤشر الفأرة

4

حيز الموضع من بمقدار -20

1

اذهب إلى الموضع من: -22 من: -15

3

حيز الموضع من بمقدار 15

تدريب 3

الإحداثيات في سكراتش

نفذ المقطع البرمجي الآتي ثلاث مرات مع ملاحظة إحداثيات الكائن الرسومي في كل مرة.

الإجابة تعتمد على تجربة الطالب هذا مثال للحل

y	x	
- 130	- 50	1
- 135	- 209	2
- 45	- 130	3

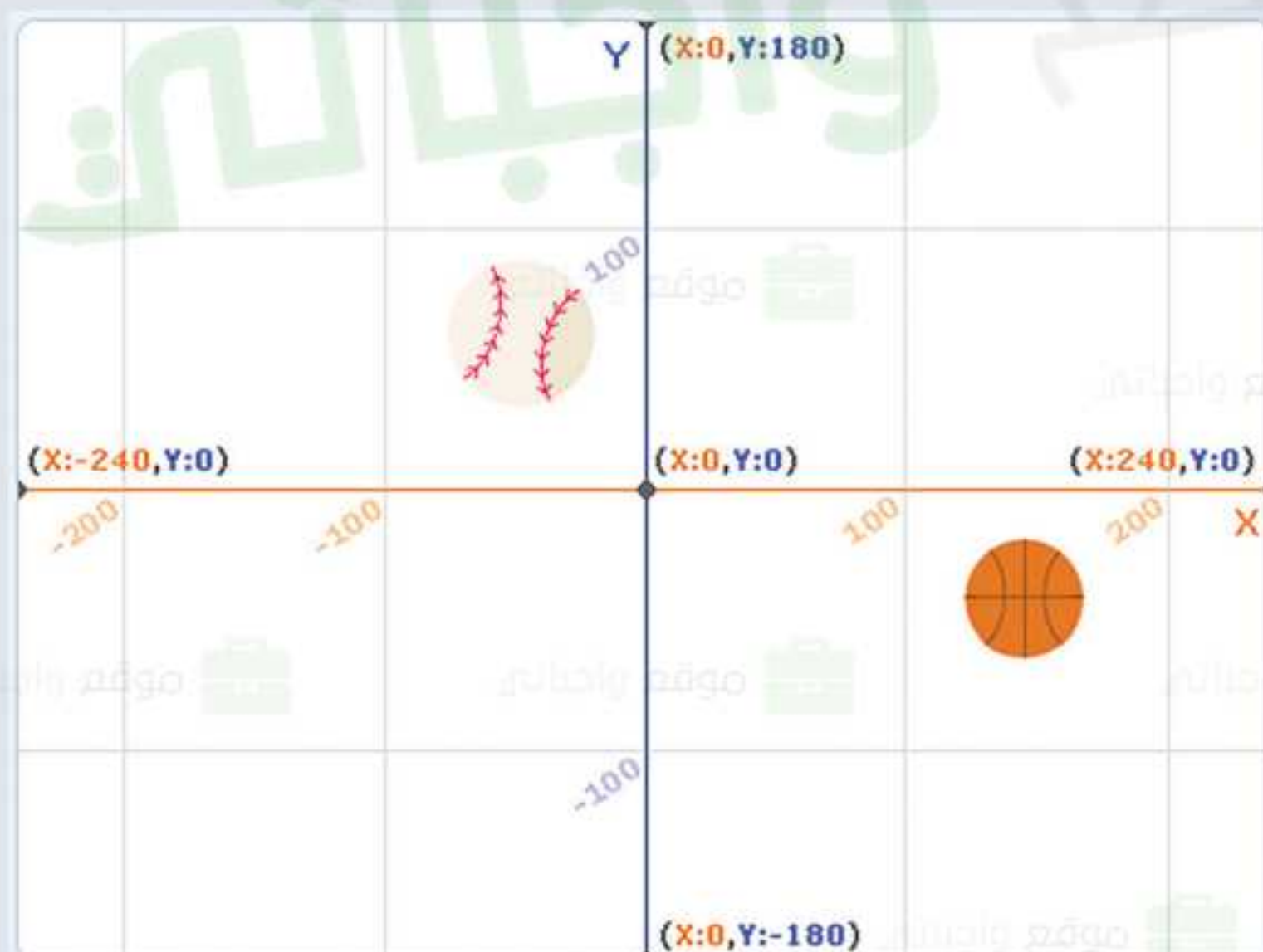


تدريب 4

الإحداثيات في سكراتش

وُضعت الكائنات في إحداثيات خطأ. اكتب المقطع البرمجي لنقل كل منها إلى الموقع الصحيح.

كرة السلة (-50, -20)
كرة البيسبول (-80, -20)



كرة البيسبول

كرة السلة



تدريب 5

التحكم في كائن باستخدام لوحة المفاتيح

أنشئ البرامج النصية الآتية:

- ما المفتاح الذي ستستخدمه لرسم خط أزرق؟ **المفتاح b**
- ما المفتاح الذي ستستخدمه لرسم خط أحمر؟ **المفتاح r**
- شغل المقطع البرمجي.





الدرس الخامس: القرارات المركبة في سكراتش

المُعامِلات في سكراتش

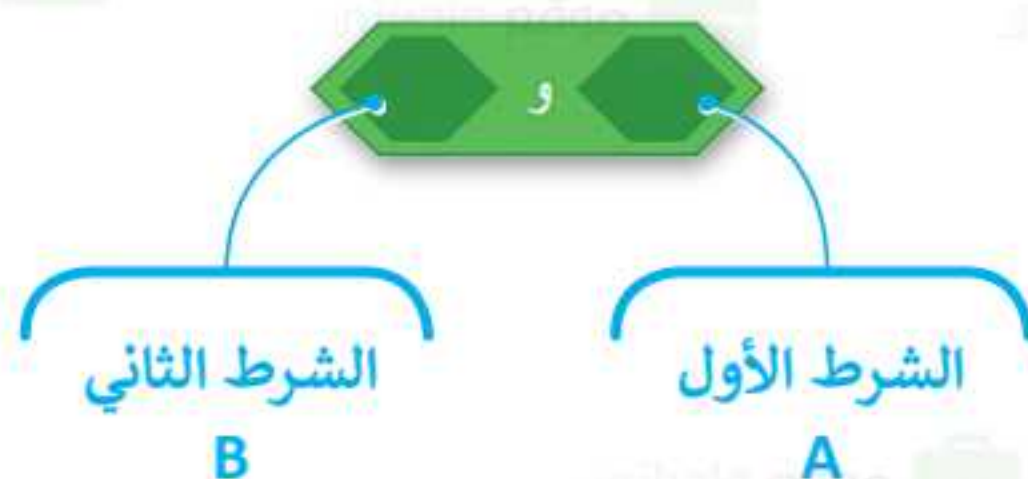
يوجد في سكراتش ثلاث فئات من لبنات المُعامِلات وهي: المُعامِلات الحسابية والمُعامِلات الشرطية والمُعامِلات المنطقية. لقد تعلمت بالفعل المُعامِلات الحسابية والشرطية. في هذا الدرس ستتعلم كيفية استخدام المُعامِلات المنطقية.

تُستخدم اللبّات والمُعامِلات الحسابية لإجراء العمليات الحسابية مثل الجمع، والطرح، والضرب، والقسمة.		المُعامِلات الحسابية (Arithmetic operators)
تُستخدم مُعامِلات المقارنة في مقارنة القيم والتصرف بناءً على النتيجة. يمكن أن تكون نتيجة التحقق الشرطي صحيحة أو خطأ.		مُعامِلات شرطية (Conditional operators)
تسمح لبّات المُعامِلات المنطقية بإجراءات مختلفة عن طريق التحكم في تغيير التدفق وذلك اعتمادًا على الشرط إذا كان صحيحًا أو خطأ.		المُعامِلات المنطقية (Logical operators)



المُعاملات المنطقية

للمُعاملات المنطقية ثلاثة أنواع وهي: () و () and () ، () أو () or () ، ليس () (not) ، ويتم استخدامها لإنشاء القرارات المركبة عن طريق التحقق من الشروط.



تضم لبنة () و () لبنتين منطقيتين، فإذا كان هناك شرطًا خطأ فإن اللبنة تُرجع خطأ.



تضم لبنة () أو () لبنتين منطقيتين، فإذا كان هناك شرطًا صحيحًا فإن اللبنة تُرجع صحيحًا.



تتحقق لبنة ليس () من الشرط بداخلها، فإذا كان خطأ فإنها ترجع صحيحًا، وإذا كان الشرط صحيحًا فإنها ترجع خطأ.



يوضح الجدول الآتي نتائج تطبيق المُعاملات المنطقية على سلسلة من الأزواج العددية المنطقية الصحيحة والخطأ، ويُطلق على هذا الجدول اسم جدول الحقيقة (Truth Table)، ويعرض ناتج المُعامل المنطقي للعديد من المُدخلات.

جدول الحقيقة

ليس A	A أو B	A و B	B	A
True	False	False	False	False
True	True	False	True	False
False	True	False	False	True
False	True	True	True	True

المُعامل المنطقي: و (and)

تحتاج في بعض الحالات إلى أن يكون الشرطان صحيحين في نفس الوقت لتنفيذ حدث ما. في المثال الآتي يُغيّر كائن القطة الألوان، ويتوقف عن تغيير الألوان ويبدأ في الدوران إذا ضغطت على مفتاحي السهم العلوي والمسافة في نفس الوقت.

تستدير القطة فقط إذا تم الضغط على كلا المفتاحين.



يجب أن يكون كلا الشرطين (A و B) صحيحين لتشغيل المقطع البرمجي داخل المساحة الأولى، وإذا كان أحدهما خطأ فسيتم تشغيل المقطع البرمجي الموجود في المساحة الثانية.

المُعامل المنطقي: أو (or)

تحتاج في بعض الحالات الأخرى إلى شرط واحد أو أكثر أن يكون صحيحاً لتنفيذ حدث ما. في هذه الحالة يُغيّر كائن القطة الألوان، ولكنه يتوقف عن تغيير الألوان ويبدأ في الدوران إذا ضغطت على مفتاح السهم العلوي أو مفتاح المسافة من لوحة المفاتيح.

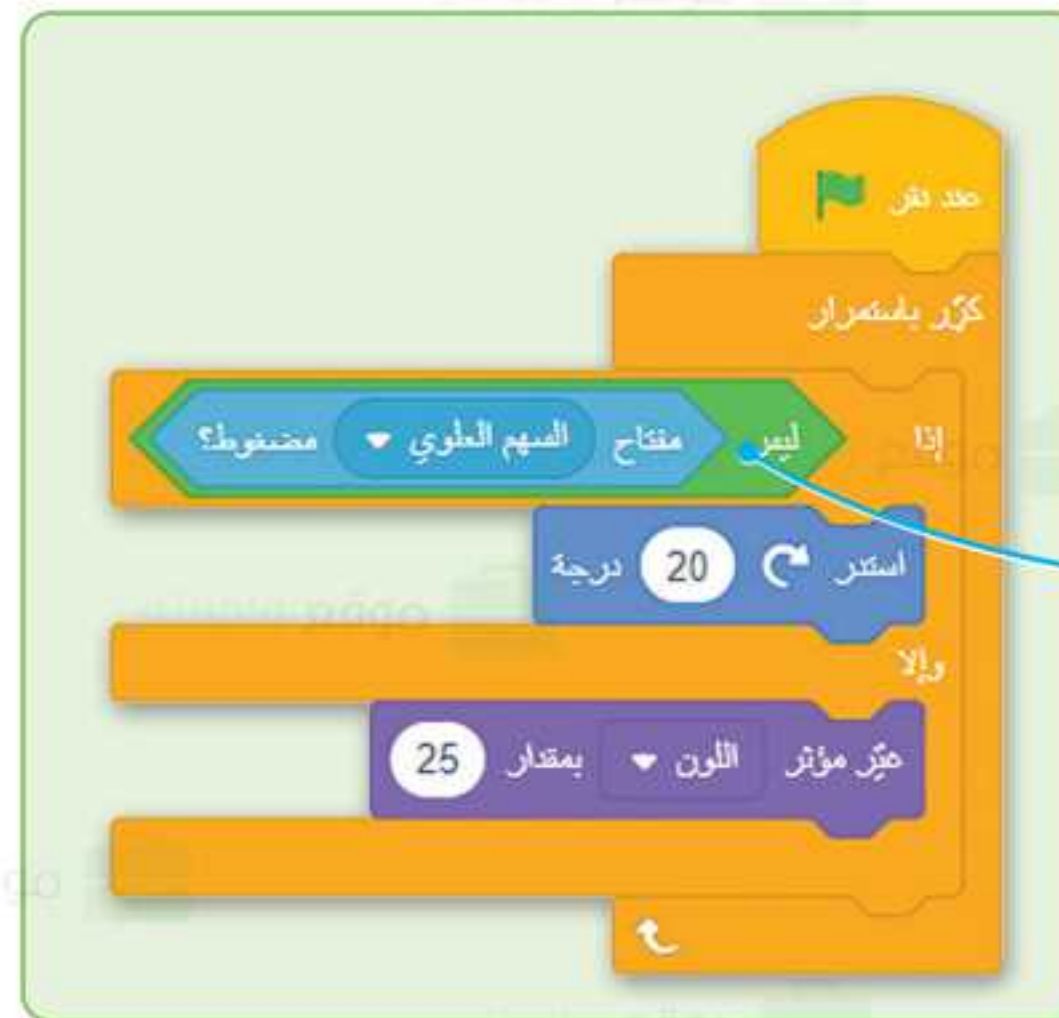
تستدير القطة إذا تم الضغط على أحد المفاتيح.



يجب أن يكون شرط واحد (A أو B) صحيحاً لتشغيل المقطع البرمجي داخل المساحة الأولى، وإذا كان كلاهما خطأ، فسيتم تشغيل المقطع البرمجي الموجود في المساحة الثانية.

المُعامل المنطقي: ليس (Not)

تحتاج في بعض الحالات الأخرى إلى أن يكون الشرط خطأ لتنفيذ حدث ما. في المثال الآتي يستدير كائن القطة، وعندما تضغط على مفتاح السهم العلوي تتوقف القطة عن الدوران وتبدأ بتغيير الألوان.



لن يتغير لون
القطة طالما لم يتم
الضغط على الزر.

يجب أن يكون الشرط (A) خطأ لتشغيل المقطع البرمجي داخل المساحة الأولى، وإذا كان الشرط صحيحًا، فسيتم تشغيل المقطع البرمجي الموجود في المساحة الثانية.

لبنة الانتظار



لبنة انتظر () ثانية (wait () secs)

في لبنة انتظر () ثانية، تنتظر اللبنة عددًا محددًا من الثواني ثم تستمر إلى اللبنة التالية.

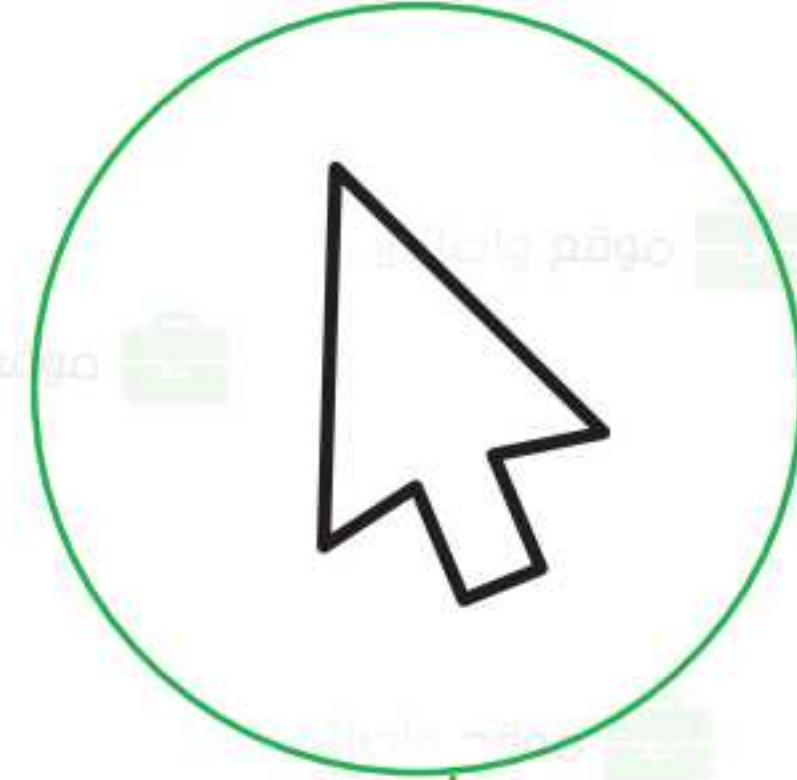


لبنة انتظر حتى () (Wait until ())

تحتاج في بعض الحالات الأخرى إلى إيقاف المقطع البرمجي والانتظار حتى حدوث حدث محدد. يمكنك استخدام لبنة انتظر حتى () التي توقف المقطع البرمجي مؤقتًا حتى يتحقق الشرط المحدد.



في المثال الآتي تريد أن تجعل الكائن ينتظر حتى يلمسه مؤشر الفأرة، وعندما يلمس مؤشر الفأرة القطة فإنه سيغير مظهره ويتحرك 30 خطوة.



إذا كان شرط انتظر حتى () تحقق، فسيتم تنفيذ المقطع البرمجي داخل المساحة.



لاحظ الفرق بين لبنة انتظر () ثانية (wait () secs) ولبنة انتظر حتى () (wait until ()). مربع لبنة انتظر () (wait ()) بيضاوي الشكل لأن المُدخل يكون عبارة فقط عن قيم، ولكن مربع انتظر حتى () (wait until ()) عبارة عن مضلع لأن المُدخل يكون شرطًا فقط.



لنطبق معًا

تدريب 1

المُعاملات المنطقية



صل اللبنة الآتية مع وظائفها.

يُرجع مُعامل اللبنة صحيحًا إذا
كان أحد الشرطين صحيحًا.

2

يُرجع مُعامل اللبنة صحيحًا إذا
كان الشرط خطأ.

3

يُحدد مُعامل اللبنة ما إذا كان
الرقم الأول لا يساوي الثاني.



يُرجع مُعامل اللبنة صحيحًا إذا
كان كلا الشرطين صحيحين.

1

1



2



3



تدريب 2

المُعامِلات المنطقية

شغل المقطع البرمجي في سكراتش ثم أكمل فراغ اللبنت بالعبرة الصحيحة:



ماذا سيحدث للمقطع البرمجي أعلاه إذا استخدمت المُعامِل المنطقي و (and) بدلاً من المُعامِل المنطقي أو (or)؟

إذا تم استخدام المعامل المنطقي "و" سيتطلب من الكائن لمس كرة السلة وكرة البيسبول معاً لقول عبارة "المس الكرة".

تدريب 3

المُعاملات المنطقية

أجب عن الأسئلة الآتية وفقًا للمقطع البرمجي أدناه:



ما المفتاح (أو المفاتيح) الذي تحتاج إلى الضغط عليه لطباعة وتحريك الكائن الرسومي على المنصة؟
نحتاج للضغط على مفتاح المسافة ومفتاح السهم الأيمن معًا.

ماذا سيحدث للمقطع البرمجي إذا استخدمت المُعامل أو (or) بدلاً من المُعامل المنطقي و (and)؟
إذا استخدمنا المعامل "أو" سيطلع ويتحرك الكائن الرسومي على المنصة عند الضغط على مفتاح المسافة أو مفتاح السهم الأيمن.

الدرس السادس: الألعاب في سكراتش

رابط الدرس الرقمي



www.ien.edu.sa

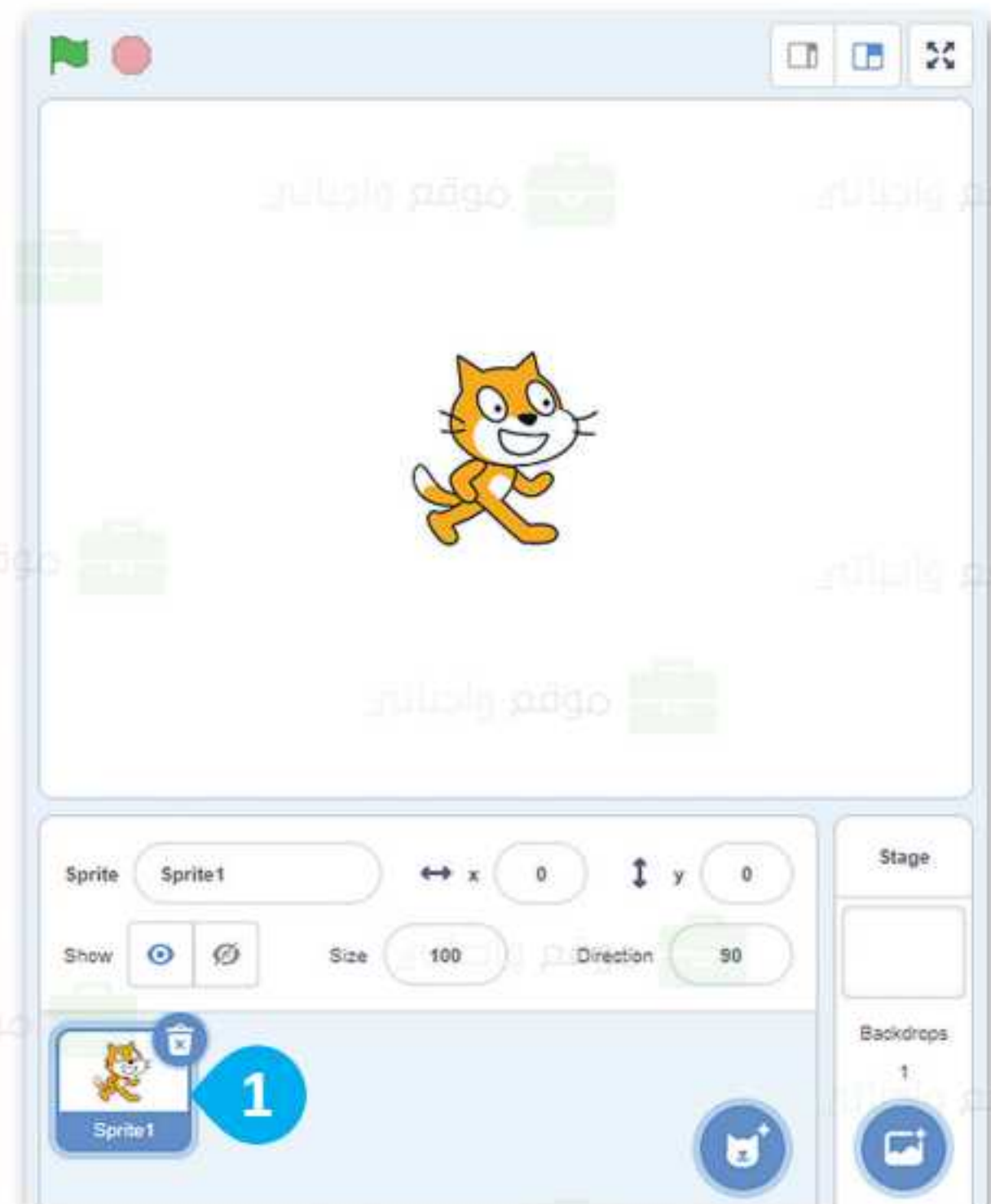
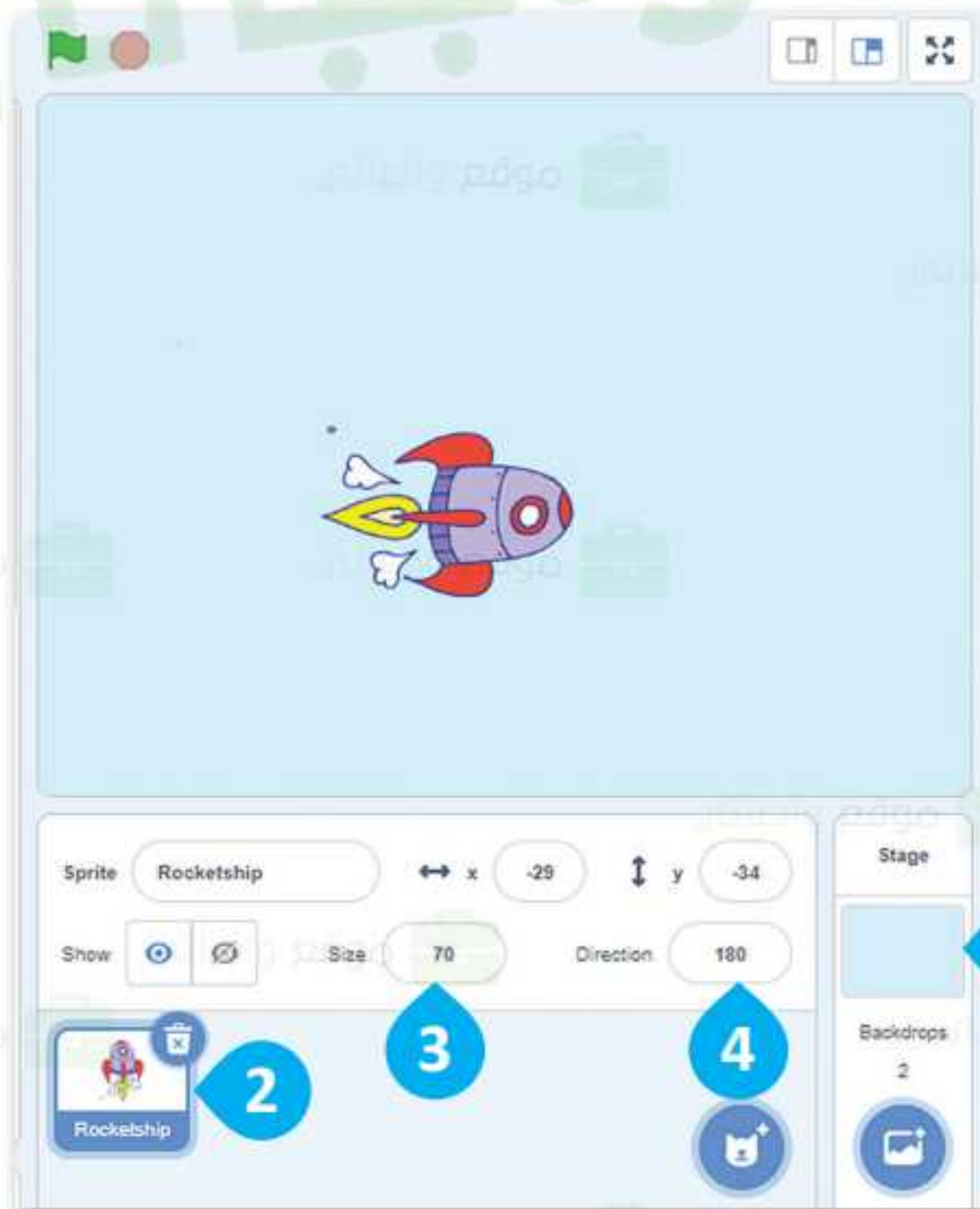
لقد أنشأت سابقًا مجموعة متنوعة من المقاطع البرمجية في سكراتش لتنفيذ مهام مختلفة، وستتعلم الآن كيفية استخدام اللبنت في سكراتش لإنشاء لعبة صغيرة.

إنشاء لعبة المركبة الفضائية

الشخصية الرئيسية في اللعبة هي المركبة الفضائية، حيث ستطير حول المدينة، ويمكنك التحكم بها من خلال لوحة المفاتيح، وستستخدم السهمين العلوي والسفلي لتجنب الغيوم والمباني. عندما تعبر المركبة الفضائية المباني أو الغيوم، يفقد اللاعب نقاطًا وعندما يعبر النجم، يكسب اللاعب نقاطًا.

إعدادات المنصة:

- 1 < احذف كائن القطة.
- 2 < أضف الكائن المركبة الفضائية (Rocketship).
- 3 < غيّر حجم الكائن إلى 70، واتجاهه إلى 180 درجة.
- 4 < أضف الخلفية السماء الزرقاء 2 (Blue Sky 2).
- 5



حركة المركبة الفضائية (Rocketship)

للتحكم في المركبة الفضائية من خلال لوحة المفاتيح عليك إنشاء مقطعين برمجيين باستخدام لبنة عند ضغط مفتاح (I).
عندما تضغط على مفتاح السهم العلوي فإن المقطع البرمجي الأول يحرك الكائن لأعلى، وعندما تضغط على مفتاح السهم السفلي فإن المقطع البرمجي الثاني يحرك الكائن لأسفل.



بعد إنشاء المقطعين البرمجييين للمركبة الفضائية اضغط على السهمين العلوي والسفلي. بدون تغيير موضع X يتحرك المقطع البرمجي لأعلى ولأسفل. عليك الآن ترقية اللعبة باستخدام تقنيات الرسوم المتحركة لإنشاء إحياء بأن المركبة الفضائية تتحرك للأمام.



نصيحة

لتحريك الكائن لأعلى أو لأسفل تحتاج إلى تغيير قيمة الإحداثي Y، ولتحريك الكائن للأمام أو للخلف عليك تغيير قيمة الإحداثي X.



تقنيات الرسوم المتحركة

الرسوم المتحركة هي تقنية تعالج الصور (أو الكائنات) الثابتة لتظهر كصور متحركة، ويتحقق الإيحاء بالحركة من خلال سلسلة سريعة من الصور المتسلسلة التي تختلف اختلافًا طفيفًا بينها.

ستستخدم هذه التقنية لكي تظهر المركبة الفضائية وهي تطير فوق المدينة في السماء. لقد أضفت سابقًا خلفية السماء الزرقاء 2 (Blue Sky 2)، والآن ابحث عن كائن المباني (Buildings) وكائن السحب (Clouds) في مكتبة سكراتش، ثم أنشئ المقاطع البرمجية الآتية لكل كائن.

الكائن السحب (Clouds)

أولاً اضبط المحور y على قيمة عالية ليرتفع الكائن السحب (Clouds) إلى السماء، ثم أرسل الكائن إلى الجانب الأيسر من المنصة بتقليل قيمة المحور x لتتحرك السحب من الجانب الأيمن إلى الجانب الأيسر للمنصة. غير أيضًا مظهر السحابة إلى الخيار التالي من القائمة لتظهر بأنها سحُب مختلفة.

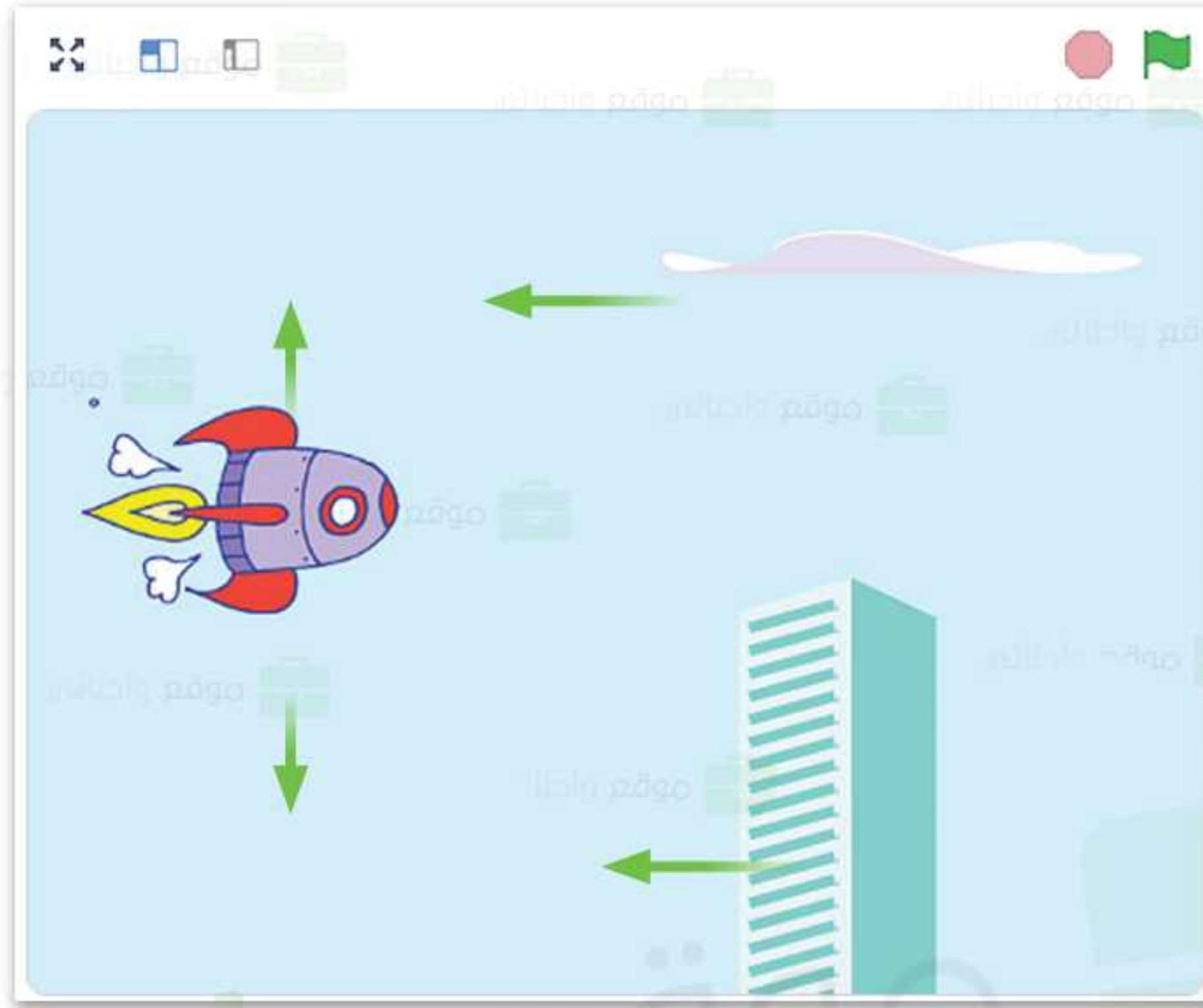


الكائن المباني (Buildings)

لوضع المباني أسفل المنصة غير حجمها إلى 80 واضبط المحور y على قيمة منخفضة. كما فعلت للكائن السحب (Clouds) أرسل الكائن المباني (Buildings) إلى الجانب الأيسر من المنصة بتقليل قيمة المحور x لتتحرك المباني من الجانب الأيمن إلى الجانب الأيسر للمنصة. غير مظهر المباني إلى الخيار التالي من القائمة لإعطاء الإيحاء بوجود مباني مختلفة.



اضغط على أيقونة ملء الشاشة وأيقونة العلم الأخضر لتحريك السحب والمباني، ثم استخدم الأسهم للسيطرة على المركبة الفضائية لتجنب الكائنات الأخرى.



برمجة الكائن لخسارة النقاط

تم بالفعل إعداد المنصة الرئيسة، وإضافة المركبة الفضائية وبرمجتها على أنها الشخصية الرئيسة والسحب والمباني على أنها عقبات.

في جميع الألعاب تقريبًا تتمتع شخصية اللعبة بعدد محدد من النقاط (points) وتخسرها بطرق متنوعة. يجب أن تخسر المركبة الفضائية واحدة من نقاطها في اللعبة عندما تلمس سحابة أو مبنى.

ستنشئ أيضًا سيناريو مثير للاهتمام لجعل لعبتك أكثر متعة، ولتحقيق ذلك يمكنك إضافة خلفيات وأصوات مختلفة وجعل المركبة الفضائية تتكلم.

انتقل إلى الكائن المركبة الفضائية (Rocketship) واتبع الخطوات الآتية لإنشاء المقطع البرمجي الجديد. يوجد بالفعل مقطعان برمجان يحركان الكائن لأعلى ولأسفل على المنصة.





إنشاء متغير النقاط (points):

- 1 < انتقل إلى فئة لبنات المتغيرات (Variables).
- 2 < اضغط على إنشاء متغير (Make a Variable).
- 3 < سمّ المتغير النقاط (points) في النافذة التي تظهر، ثم اضغط على موافق (OK).
- 4 < حدّد المربع بجوار متغير النقاط (points) لتنشيطه.





برمجة الكائن المركبة الفضائية (Rocketship)



عند الضغط على أيقونة العلم يضبط المقطع البرمجي عدد النقاط على 5. ثم يضع المركبة الفضائية على الجانب الأيسر من المنصة ويغير الخلفية إلى السماء الزرقاء 2 (Blue Sky 2)، ثم يحرك الكائن إلى الأمام بحيث تتحرك الغيوم والمبنى خلفه، وبعد ذلك يصدر صوتًا يشير إلى أن اللعبة قد بدأت.

حرك الكائن
Rocketship
(المركبة
الفضائية) أمام
كل الكائنات.

بداية اللعبة.

بعد الانتهاء من المقطع
البرمجي اضغط على
أيقونة العلم الأخضر
وستبدأ اللعبة.

مرحلة يتحقق
المقطع
البرمجي مما إذا
كانت المركبة
الفضائية تفقد
أحد نقاطها.

نهاية اللعبة.

تقوم لبنة أوقف (الكل)
(stop (all))
المقاطع البرمجية بمجرد
تنفيذ جميع الحركات.



برمجة الكائن لكسب النقاط

الآن، بعد أن أنشأت الكائن النجمة (Star)، عليك برمجة الكائن المركبة الفضائية (Rocketship) من أجل التفاعل مع النجمة وكسب النقاط.

انتقل إلى المقطع البرمجي للمركبة الفضائية وأضف لبنة إذا () وإلا (if () then) من فئة التحكم (Control) للتحقق مما إذا لمست المركبة الفضائية الكائن النجمة (Star). يتم تشغيل الصوت عندما تكون حالة اللبنة صحيحة وتزداد قيمة النقاط.

المقاطع البرمجية جاهزة.
استمتع باللعبة.



```

    عند غر
    اجعل points مساوياً 5
    اذهب إلى الموضع من: -150 من: 0
    غير الخلفية إلى Blue Sky 2
    انتقل إلى الطبقة الأولى
    ابدأ الصوت space ripple
    قل دعنا نظهر لمدة 2 ثانية

    كثر باستمرار
    إذا 0 < points
    إذا متمسك لـ Star
    ابدأ الصوت laser1
    غير points بمقدار 1
    إذا متمسك لـ Clouds أو متمسك لـ Buildings
    شغل الصوت laser2 حتى انتهاء
    غير points بمقدار -1
    انتظر 1 ثانية
    وإلا
    غير المظهر إلى rocketship-e
    غير الخلفية إلى Galaxy
    فكر انتهت اللعبة لمدة 2 ثانية
    أوقف الكل
  
```

يفحص المقطع البرمجي ما إذا كانت المركبة الفضائية ستكسب النقاط أم لا.

لنطبق معًا

تدريب 1

المعاملات في سكراتش

أكمل الفراغات في اللبنت الآتية لتنفيذ المهمة.

إذا كانت قيمة العمر (age) أكبر من أو تساوي 18، اكتب "يمكنك التقدم بطلب للحصول على رخصة قيادة السيارة".

The image shows a Scratch script within a green rectangular frame. The script consists of the following blocks:

- When green flag clicked** (yellow block with a green flag icon).
- Ask and wait** (blue block) with the text "ما هو عمرك؟" (What is your age?) and "وانتظر" (Wait).
- Set age to** (orange block) with a dropdown menu showing "age" and a value of "الإجابة" (The answer).
- If-then** (orange block) with the condition "إذا" (If) and the expression $age \geq 18$ (age greater than or equal to 18).
- Say** (purple block) with the text "يمكنك التقدم بطلب للحصول على رخصة قيادة السيارة" (You can apply for a driving license), a duration of "2" seconds, and the option "ثانية" (Seconds).

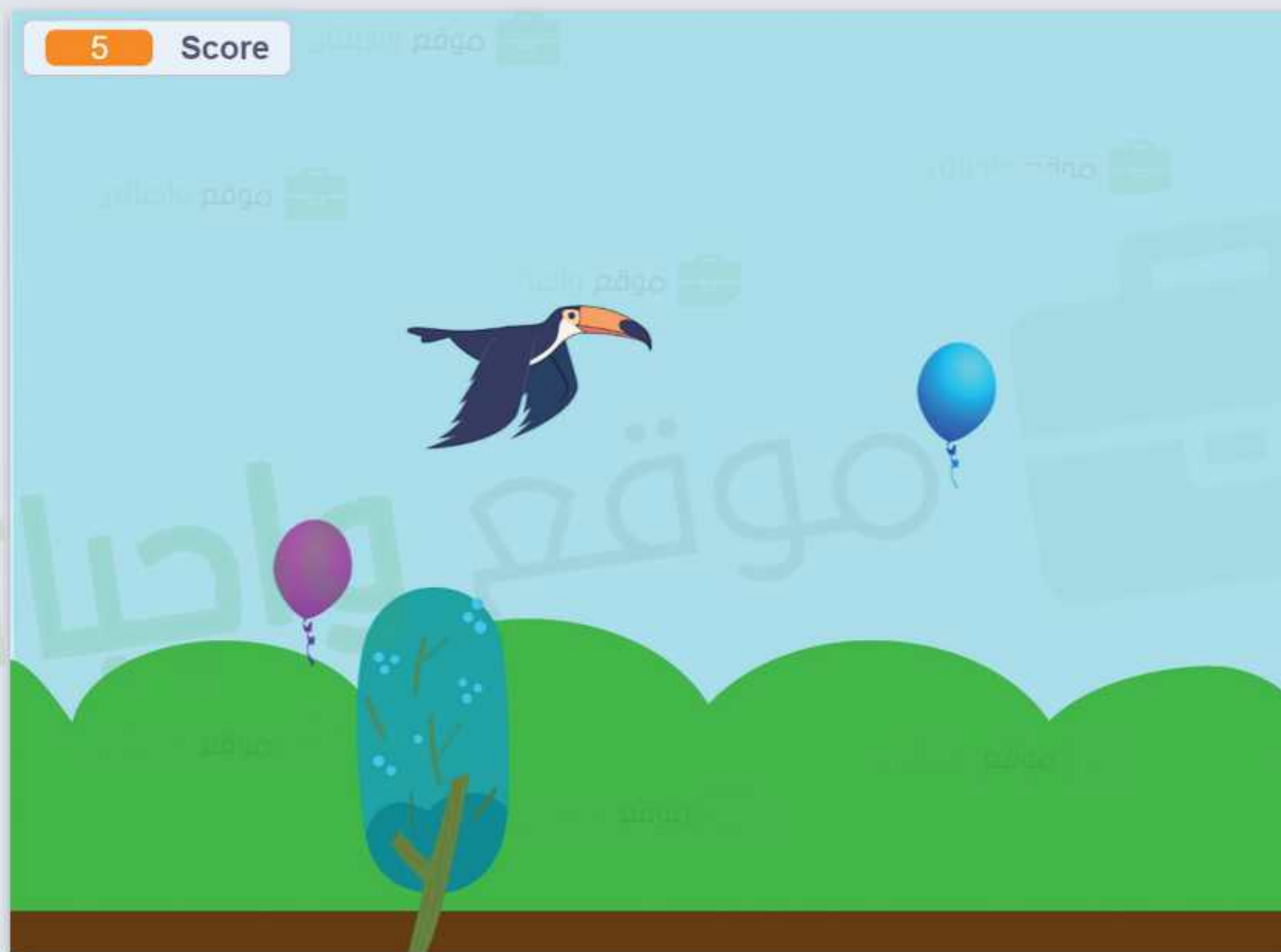
A blue arrow points to the text area of the "Say" block.

تدريب 2

إنشاء لعبة

تطبيق عملي يترك للطالب

ستبرمج طائرًا يطير ويتحكم فيه من خلال لوحة المفاتيح، وستستخدم الأسهم لأعلى ولأسفل لتجنب المعوقات مثل الأشجار والبالونات. عند بدء اللعبة ستمتلك 5 نقاط، ثم يبدأ بخسارة نقاطه كلما لامس الطائر شجرة أو بالونًا.



تدريب 3

إنشاء برنامج



خطوات الخوارزمية:

1. يبدأ البرنامج عند النقر على العلم الأخضر.
2. يسأل البرنامج ما إذا كانت السماء تمطر وينتظر الإجابة.
3. يتحقق البرنامج من الإجابة.
4. إذا كانت الإجابة نعم سيقول البرنامج "افتح المظلة".
5. إذا كانت الإجابة ليست نعم سيقول البرنامج "ضع المظلة في حقيبتك".

اكتب خوارزمية لإنشاء مقطع برمجي يسأل ما إذا كانت السماء تمطر.

بناءً على إجابة المُستخدم سيقول البرنامج "افتح المظلة"، أو "ضع المظلة في حقيبتك". حوّل الخوارزمية إلى مقطع برمجي ثم نفذ.



تدريب 4

تابع لعبة المركبة الفضائية

تطبيق عملي يترك للطالب

لاحظ أنه إذا لم تنفذ النقاط لديك، فلن تنتهي اللعبة. أضف عدادًا لإيقافها، بمجرد نفاد وقت اللاعب. اضبط قيمة العداد إلى 30 وابدأ بتقليلها. ثم تحقق من قيم "العداد" أو "النقاط" للتحكم في تدفق اللعبة.



مشروع الوحدة

الغوص وصيد الأسماك



1

في البداية أنشئ مشروعًا جديدًا في سكراتش وسمّه، ثم اختر الخلفية التي ستحتاجها اللعبة.

2

حدد الكائنات المناسبة، ستحتاج في هذه اللعبة إلى الكائنات الآتية:



Diver

الغواص



Key

المفتاح



Fish

السمكة



Jellyfish

قنديل البحر

3

برمج الكائن الغواص 1 (Diver1) ليتحرك حول المنصة باستخدام لوحة المفاتيح، ثم اختر المفاتيح التي تريدها للحركة.

4

أنشئ المقاطع البرمجية للكائن السمكة (Fish) والكائن قنديل البحر (Jellyfish) والكائن المفتاح (Key) لجعلها تتحرك من الجانب الأيمن إلى الجانب الأيسر على المنصة، حاول أن تجعل الحركة واقعية قدر الإمكان، ولا تنس تغيير مظهرها أيضًا.

5

أضف المزيد من المقاطع البرمجية للكائن الغواص 1 (Diver1)، حيث يجعل هذا المقطع حورية البحر تكسب نقاطًا (Points) عندما تمسك بمفتاح وتفقدتها عندما تلمسها سمكة أو قنديل البحر.

6

اكتمل المشروع، شغل المقطع البرمجي واختبره. استمتع باللعبة.

في الختام

جدول المهارات

درجة الإتقان		المهارة
لم يتقن	أتقن	
		1. استخدام لبنة كرّر حتى لتكرار مقطع برمجي.
		2. تصنيف أنواع المتغيرات وكيفية استخدامها لتخزين المعلومات.
		3. إنشاء مقطع برمجي لإجراء العمليات الحسابية باستخدام الأرقام والمتغيرات.
		4. إنشاء مقاطع برمجية يتم فيها استخدام لبنات اتخاذ القرارات إذا () وإلا.
		5. تحديد موقع نقطة باستخدام إحداثياتها.
		6. استخدام الإحداثيات لتحريك الكائنات حول المنصة.
		7. تحريك الكائنات باستخدام لوحة المفاتيح وإحداثياتها.
		8. استخدام المُعامِلات المنطقية في المقطع البرمجي.
		9. اتخاذ القرارات باستخدام ظروف مركبة.
		10. استخدام تقنيات الرسوم المتحركة.
		11. إنشاء لعبة تفاعلية.

المصطلحات

Numeric Variables	المتغيرات الرقمية	Animation Techniques	تقنيات الرسوم المتحركة
Operators	المُعاملات	Axis	محور
Pictographs	الرسوم التوضيحية	Calculations	العمليات الحسابية
Position	موضع	Condition	شرط
Random	عشوائي	Control	تحكم
String Variables	المتغيرات النصية	Coordinate System	نظام الإحداثيات
Touch	ملامس	Decision	قرار
Truth Table	جدول الحقيقة	Game	لعبة
Value	قيمة	Graphs	رسوم
Variable	متغير	Keyboard	لوحة المفاتيح
		Loops	التكرارات



اختبر نفسك

السؤال الأول

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخطأ فيما يلي:
	✓	1. تتضمن النمذجة ثنائية الأبعاد إنشاء صور مسطحة باستخدام الخطوط والمنحنيات والأشكال.
	✓	2. المحاور الثلاثة للفضاء ثلاثي الأبعاد هي المحاور X و Y و Z.
	✓	3. يُمثل المكعب في نظام إحداثيات ثلاثي الأبعاد على ثلاثة محاور هي الطول والعرض والارتفاع.
✗		4. تينكر كاد هو برنامج نمذجة ثنائية الأبعاد.
	✓	5. يمكنك إنشاء حساب في برنامج تينكر كاد باستخدام بريدك الإلكتروني أو تسجيل الدخول باستخدام حساب جوجل الخاص بك.
	✓	6. تتيح لك أداة المرآة في برنامج تينكر كاد إنشاء صورة معكوسة للشكل ثلاثي الأبعاد.
	✓	7. الأشكال الصلبة هي الأشكال التي لها سطح صلب وتشغل مساحة محددة.
	✓	8. تسمح لك أداة فك التجميع فصل مجموعة من الأشكال إلى أشكال مستقلة.
	✓	9. تتيح لك أداة اللصق إضافة شكل منسوخ في تصميمك.
✗		10. عليك القيام بمحاذاة الأشكال يدويًا في برنامج تينكر كاد.
✗		11. تتيح لك أداة فيوكيوب في برنامج تينكر كاد تغيير لون الشكل ثلاثي الأبعاد.

اختبر نفسك

السؤال الثاني

اختر الإجابة الصحيحة.	
☐	إنشاء أشكال افتراضية ثلاثية الأبعاد.
☒	إنشاء صور مسطحة.
☐	تصميم المباني والهندسة المعمارية.
☐	إنشاء الأشكال من الحياة اليومية.
1. ما استخدام النمذجة ثنائية الأبعاد؟	
☐	أداة المرآة.
☐	أداة المحاذاة.
☒	أداة التجميع.
☐	أداة الحذف.
2. ما الأداة التي تسمح لك بتجميع شكلين أو أكثر معًا للتعامل معهم كشكل واحد في برنامج تينكر كاد؟	
☐	إنشاء شكل صلب ثلاثي الأبعاد.
☒	إنشاء فراغ في أي جسم صلب.
☐	تحريك الأشكال في مساحة العمل.
☐	تغيير لون الشكل ثلاثي الأبعاد.
3. ما الهدف من استخدام خيار الشكل المفرغ في برنامج تينكر كاد؟	

اختبر نفسك

السؤال الثالث

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخطأ فيما يلي:
	✓	1. يسمح مايكروسوفت إكسل للمستخدمين بتخزين البيانات في الخلايا وإجراء العمليات الحسابية باستخدام شريط الصيغة.
✗		2. في مايكروسوفت إكسل، لا يمكن تغيير تنسيق الخلايا لاستيعاب التنسيقات المختلفة للقيم الرقمية.
	✓	3. يستخدم الرمز "+" لجمع القيم في مايكروسوفت إكسل.
✗		4. يستخدم الرمز "*" لطرح القيم في مايكروسوفت إكسل.
	✓	5. يتم ترتيب أولوية إجراء العمليات الحسابية في مايكروسوفت إكسل وفقًا لترتيب قواعد العمليات.
	✓	6. في ترتيب العمليات، يتم تنفيذ عمليات الضرب والقسمة قبل عمليات الجمع والطرح.
✗		7. يتم تنفيذ العمليات بين الأقواس بعد إجراء عمليات الضرب والقسمة في مايكروسوفت إكسل.
	✓	8. إذا حذفت الرمز "=" في صيغة في مايكروسوفت إكسل، فسيتم التعامل مع المعادلة كنص ولن يتم إجراء الحسابات.
	✓	9. تستخدم الأقواس لتغيير ترتيب العمليات الحسابية في مايكروسوفت إكسل.
✗		10. في الصيغة $5+(2*3)=$ ، سيقوم مايكروسوفت إكسل أولاً بتنفيذ عملية الجمع.
	✓	11. نتيجة الصيغة $3*(5+2)=$ هي 21 عند حسابها في مايكروسوفت إكسل.
	✓	12. في مايكروسوفت إكسل، لا تنسخ ميزة التعبئة التلقائية المعادلة فحسب، بل تنسخ أيضًا تنسيق الخلية.

اختبر نفسك

السؤال الرابع

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخاطئة فيما يلي:
	✓	1. في العمليات الحسابية في مايكروسوفت إكسل، يكون للأسس أو القوى أولوية أعلى من الضرب والقسمة والجمع ويجب حسابها أولاً.
✗		2. ليس من المهم اتباع ترتيب العمليات في مايكروسوفت إكسل، ولا تؤثر الأقواس على الترتيب الذي يتم تنفيذ العمليات الحسابية به.
	✓	3. النسبة المئوية هي طريقة للتعبير عن رقم في صورة كسر من 100.
	✓	4. غالبًا ما يستخدم الرمز "%" للإشارة إلى النسبة المئوية.
	✓	5. تُستخدم النسب المئوية بشكل شائع لتمثيل أجزاء من الكل أو للتعبير عن التغييرات في القيم بمرور الوقت.
✗		6. لكتابة رمز النسبة المئوية (%) في مايكروسوفت إكسل، تحتاج إلى الضغط على Shift + 7 .
	✓	7. عند كتابة معادلة في إكسل، يجب عليك كتابتها من اليسار إلى اليمين على الرغم من كتابة النص العربي من اليمين إلى اليسار.
	✓	8. المخططات البيانية عبارة عن تمثيلات رسومية للبيانات التي تساعد في تحليل المعلومات المعقدة وتسهيل فهمها.
	✓	9. يمكن أن تساعد إضافة تسميات البيانات في المخطط الدائري المجوف في جعل بياناتك أكثر وضوحًا وإفادة وسهولة في الفهم.
✗		10. إذا كان عدد الأعمدة كبيرًا في صفحتك فإنه من الأفضل تعيين اتجاه الصفحة ليكون عموديًا (الارتفاع أكبر من العرض).
	✓	11. عند طباعة مستند في إكسل، من المهم مراجعة إعدادات الطباعة للتأكد من تحديد اتجاه الصفحة الصحيح والهوامش والإعدادات الأخرى لتحقيق النتائج المرجوة.

اختبر نفسك

السؤال الخامس

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخطأ فيما يلي:
	✓	1. تُعدُّ قاعدة البيانات نظامًا لتنظيم البيانات.
✗		2. تحتوي قاعدة البيانات على جدول واحد.
✗		3. حقل قاعدة البيانات هو مجموعة كاملة من المعلومات.
	✓	4. تسمى كل خلية حقلًا في قاعدة البيانات.
	✓	5. يُعدُّ مايكروسوفت أكسس من البرامج الشائعة لقواعد البيانات.
	✓	6. يجب أن تتعلق جميع المعلومات الموجودة في قاعدة البيانات بنفس الموضوع.
✗		7. يمكنك استخدام علامة تبويب تخطيط الصفحة لتنسيق نطاق خلايا كجدول.
✗		8. ميزة التنسيق كجدول في مايكروسوفت إكسل لا تضيف وظيفتي الفرز والتصفية إلى بياناتك.
	✓	9. يساعدك فرز البيانات وتصنيفها على فهم البيانات وتنظيمها بشكل أفضل.
	✓	10. يوجد نوعان من الفرز في مايكروسوفت إكسل وهما: الفرز التصاعدي والفرز التنازلي.
	✓	11. عند تطبيق التصفية على جدول قاعدة البيانات ستبقى السجلات، ولكن لن يتم عرضها.

اختبر نفسك

السؤال السادس

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخاطئة فيما يلي:
X		1. تُستخدم البيانات لاتخاذ القرارات، بينما تُستخدم المعلومات لتخزين البيانات.
	✓	2. البيانات هي الحقائق الأولية الموجودة حولك.
	✓	3. البيانات هي قيم لا تعطي معنى وهي منفردة.
	✓	4. تُعدّ المعلومات سهلة الفهم لأنها مترابطة.
X		5. يتوفر التنظيم في البيانات، بينما لا يتوفر في المعلومات.
X		6. المعلومات هي مُدخلات للحاسب.
	✓	7. تُعدّ البيانات العددية نوعًا من أنواع البيانات.
	✓	8. قد تكون البيانات أحيانًا على شكل صور ومقاطع فيديو وأصوات.
X		9. هناك نوعان فقط من البيانات وهما: النصوص والأرقام.
	✓	10. يُعدّ عدد الأشخاص الذين زاروا متحفًا خلال العام مثالًا على البيانات العددية.
X		11. تتكون البيانات الأبجدية من جميع الحروف الأبجدية والأرقام والرموز الخاصة، مثل # و \$ و % وما إلى ذلك.
	✓	12. يمكن أن تُستخدم البيانات الأبجدية لتمثيل اسم دولة مثلًا.

اختبر نفسك

السؤال السابع

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخطأ فيما يلي:
	✓	1. تسمح لك لبنة كُرّر حتى () () (repeat until) بتكرار مجموعة من الإجراءات حتى يتم استيفاء شرط معين.
✗		2. ستكرر لبنة كُرّر حتى () () (repeat until) المقطع البرمجي بداخلها لعدد محدد من المرات.
	✓	3. يمكن تداخل لبنة كُرّر حتى () () (repeat until) داخل لبنة كُرّر حتى (repeat until) أخرى.
	✓	4. تسمى المُعاملات المستخدمة لإجراء العمليات الحسابية بالمُعاملات الرياضية (Mathematical Operators).
	✓	5. المتغير في سكراتش يمكن أن يحتوي على قيمة واحدة.
	✓	6. يمكن استخدام لبنة إذا () () وإلا (if () then, else)، لإنشاء هيكل لاتخاذ القرار في سكراتش.
	✓	7. المتغيرات في سكراتش تستخدم لتخزين ومعالجة البيانات.
✗		8. بمجرد إنشاء متغير في سكراتش، لا يمكنك تغيير قيمته.
	✓	9. في سكراتش يمكن استخدام مُعامل التشغيل "+" لربط سلسلتين نصيتين.
✗		10. سكراتش يسمح لك فقط بإنشاء متغيرات رقمية.
	✓	11. لبنة إذا () () وإلا (if () then, else) إذا كان الشرط صحيحًا، فسيتم تنفيذ اللبنة الموجودة أسفل إذا (if)، وإذا كان الشرط خطأ، فسيتم تنفيذ اللبنة الموجودة أدنى لبنة إلا (else).

اختبر نفسك

السؤال الثامن

اقرأ المقطع البرمجي الآتي بعناية واستنادًا إلى عُمر سعد، اكتشف عُمر خالد.



عُمر خالد = 4

1 عُمر سعد = 5

عُمر خالد = 14

2 عُمر سعد = 10

عُمر خالد = 26

3 عُمر سعد = 16



اختبر نفسك

السؤال التاسع

خطأ	صحيحة	حدد الجملة الصحيحة والجملة الخطأ فيما يلي:
X		1. يكون المحور y أفقيًا والمحور x عموديًا في نظام الإحداثيات الديكارتية.
	✓	2. يمكن استخدام المُعامل المنطقي ليس () (not) لعكس حالة الشرط.
X		3. تقع النقطة ذات الإحداثيات (0,0) في نظام الإحداثيات الديكارتية في الركن الأيسر السفلي من المنصة.
	✓	4. يُرجع المُعامل المنطقي () و () () and () في سكراتش صحيحًا فقط إذا كان كلا الشرطين فيه صحيحين.
X		5. يُرجع المُعامل المنطقي () أو () () or () في سكراتش صحيحًا فقط إذا كان كلا الشرطين فيه صحيحين.
	✓	6. يعكس المُعامل المنطقي ليس () (not) في سكراتش القيمة المنطقية للعملية.
	✓	7. تنتقل لبنة اذهب إلى الموضع س: () ص: () y: () x: go to () الكائن إلى موقع الإحداثيات المحددة.
X		8. تحرك لبنة غيّر الموضع ص بمقدار () (change y by) الكائن عددًا معينًا من الخطوات لأعلى.
	✓	9. لبنة اذهب إلى (موضع عشوائي) (go to random position) تنقل الكائن إلى موقع عشوائي على المنصة.
	✓	10. تجعل لبنة انزلق خلال () ثانية إلى الموضع س: () ص: () y: () x: glide () secs to () الكائن يتحرك بسلاسة إلى موقع الإحداثيات x و y، في فترة زمنية محددة.

اختبر نفسك

السؤال العاشر



اقرأ المقطع البرمجي الآتي بعناية وشرح وظيفة كل جزء:

1 **يغير الكائن مظهره إلى Grasshopper-a.**

2 **ينتقل الكائن إلى موقع عشوائي على المنصة.**

3 **يتحقق البرنامج من الشرط "إذا كان الكائن ملامس اللون البني أو اللون الأخضر".**

4 **إذا لم يتحقق الشرط يقوم البرنامج بتغيير مظهر الكائن إلى Grasshopper-c.**

