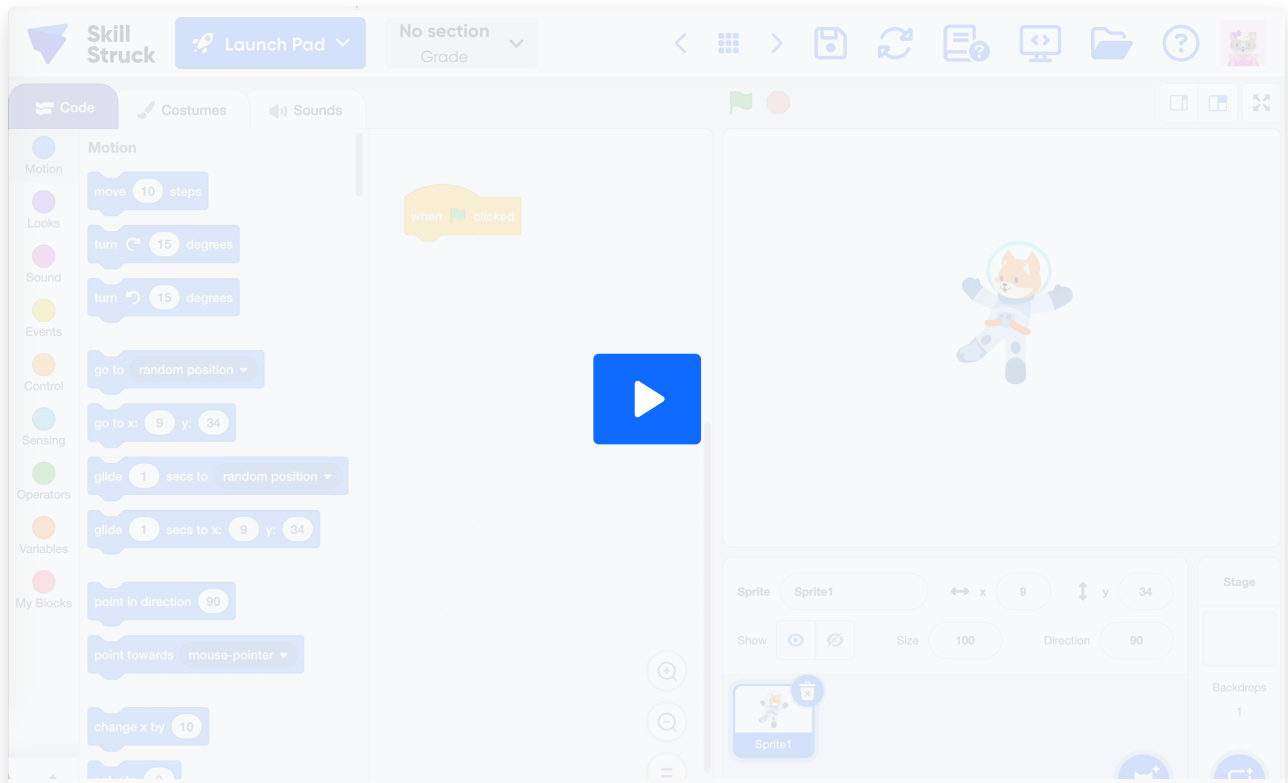
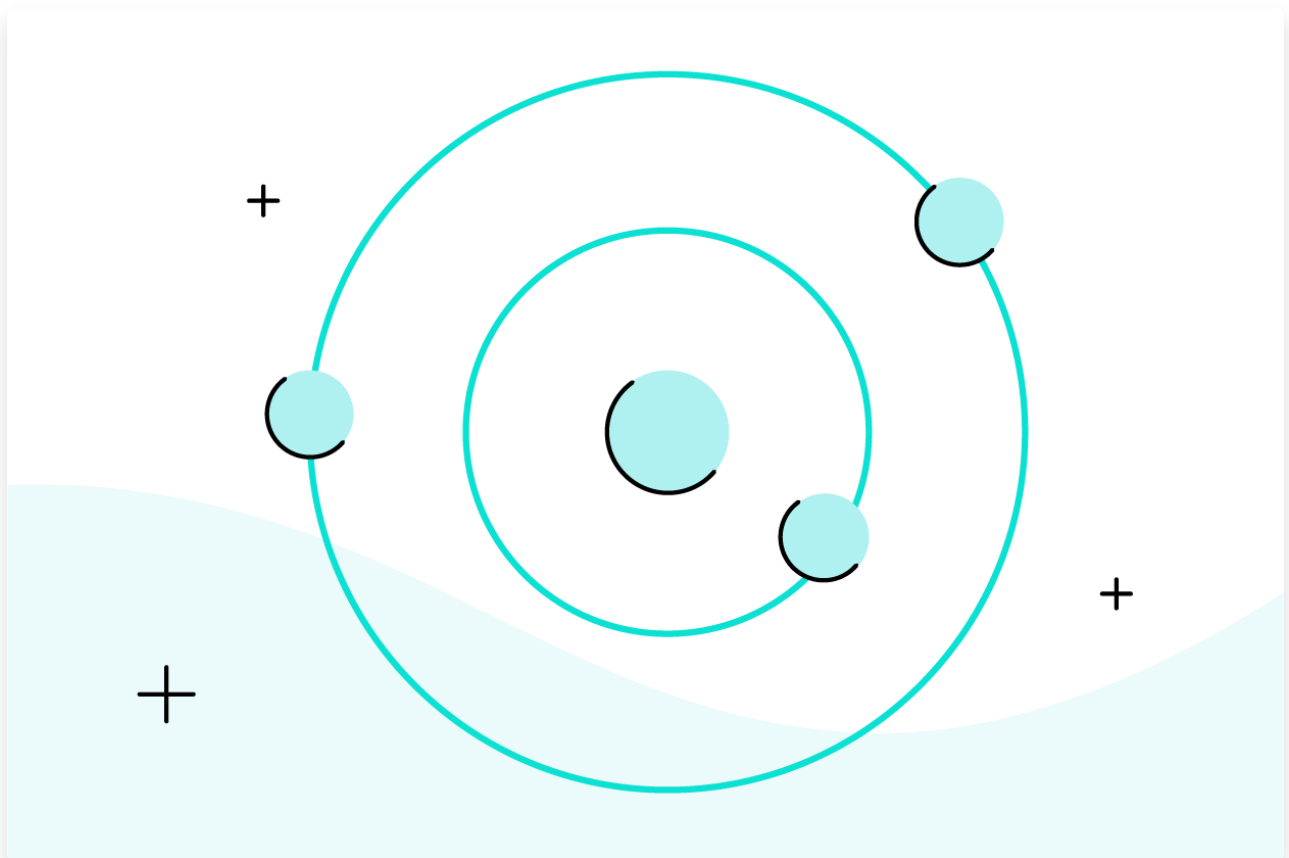


Sensing

Textbook

Sensing





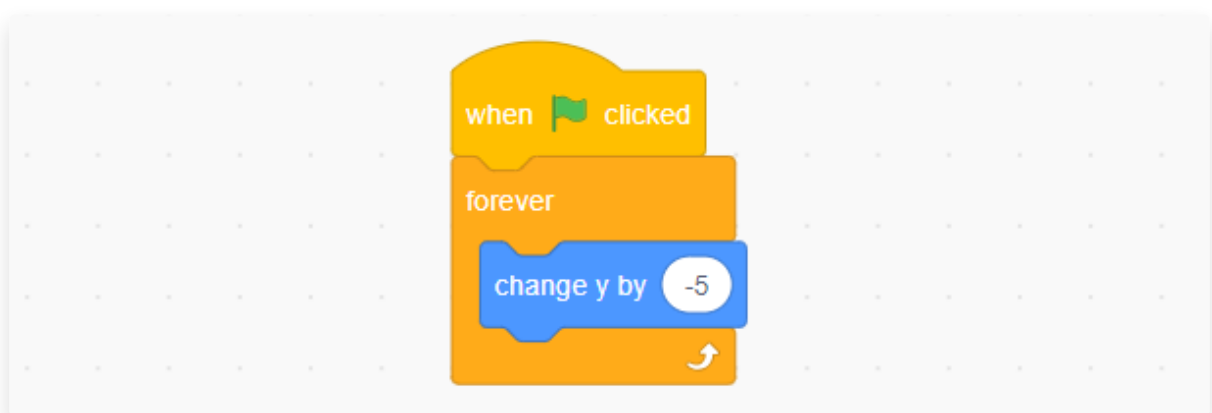
The frog hops from lily pad to lily pad, but as it gets too close to the edge of one, it leaps away just in time, as if it can feel the water waiting below. It's not just jumping at random—it's reacting to what's around it. Just like how a character in a game knows when it's about to bump into something, sensing helps things figure out what to do next. It's the quiet "feeling" that makes everything move just right.

Sensing Blocks

Sensing blocks help us see how things in a project are working together and can notice when a sprite touches the edge or when we move the mouse.

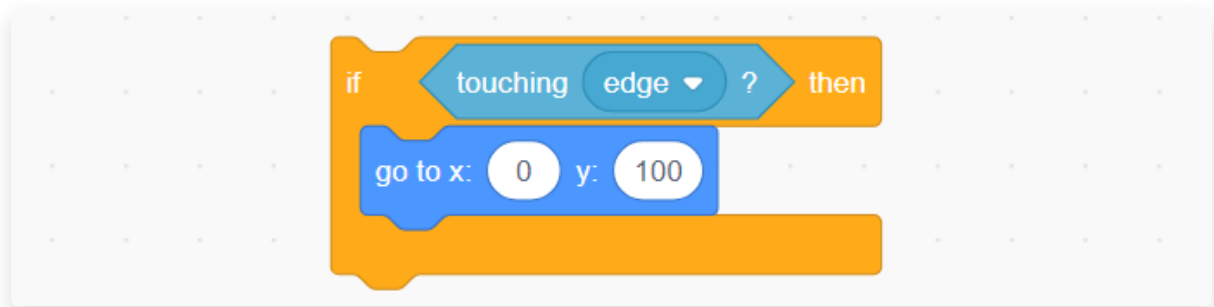
For this example, we will create an [algorithm](#) where your sprite starts at the top of your screen and moves down the page. The sprite will bounce back up to the top of the screen when it hits the edge. To do this:

1. Select one sprite and begin with the green flag event block.
2. Drag a **forever** block and connect it to your event block.
3. Add a **change y by 10** motion block inside the forever block. Change the number to -5.



4. Drag an **if then** control block and connect it inside the forever block, below the motion block.

5. Drag the `touching mouse-pointer` sensing block and connect it in the blank hexagonal space of the if line in your control block. Click the dropdown and change `mouse-pointer` to `edge`.
6. Drag the `go to x: y:` motion block and connect it in your conditional. Set the coordinates to be somewhere at the top of your scene. For example, (0, 100).



7. Press the green flag to watch your code!

Breaking Down Problems with Sensing

When we have a big problem to solve in coding, like making a game or an animation, it's often too big to solve all at once. Great coders know how to **decompose a problem**. This means they break a big, complex problem into smaller, individual, and manageable parts. It's like taking apart a giant block castle to build it piece by piece!

Let's look at our example: making a sprite move down and then bounce back up when it hits the edge. This might seem like one big thing, but we can break it down!

Our Big Problem: Make the sprite go down and bounce back up when it hits the bottom.

Let's Decompose It (Break it Down!):

1. **Part 1: Make the sprite move down all the time.**
 - *Question to ask:* How do I make the sprite move downwards forever? (This is where our `forever` loop and `change y by -5` come in!)
2. **Part 2: Know when the sprite touches the bottom edge.**
 - *Question to ask:* How can the sprite "feel" or "sense" if it's at the very bottom of the screen? (This is where our `touching edge?` sensing block is super important!)
3. **Part 3: If it touches the edge, make it go back to the top.**
 - *Question to ask:* What should happen *only if* the sprite touches the edge? (This is where our `if then` conditional block and `go to x: y:` block come in!)

By breaking down the big problem into these three smaller parts, it becomes much easier to figure out what code blocks we need and how they should fit together. We solve each small problem, and then put the solutions together to solve the big problem! This way, we **analyze the problem in a way that makes sense** and **create a plan** to logically order our steps.

Critical Thinking Questions

1. Can you think of other times in your daily life when you use your senses to react to something happening around you?

2. How would a game be different if it didn't have sensing? Would it be harder to play or less interesting? Why?

Questions (5)

1. True or False: Sensing blocks help us see how things in a project work together.

MULTIPLE CHOICE

Choose the correct answer:

- A. True
- B. False

2. Click the two examples of sensors below:

SELECT MULTIPLE

Select all that apply:

- A. when touching the edge
- B. when touching a color
- C. when variables are used

3. Which sensing block can detect when a sprite touches the edge of the screen?

MULTIPLE CHOICE

Choose the correct answer:

- A. Touching color
- B. Touching edge
- C. Touching sprite

4. What do you use to move a sprite?

MULTIPLE CHOICE

Choose the correct answer:

- A. Sound blocks
- B. Pen blocks
- C. Looks blocks
- D. Motion blocks

5. What happens when the condition in an "if then" block is true?

Choose the correct answer:

- A. The program stops running
- B. The program ignores the block
- C. The code inside the "if then" block runs
- D. The sprite changes color

Games (3)

1. Sensing Typing Game

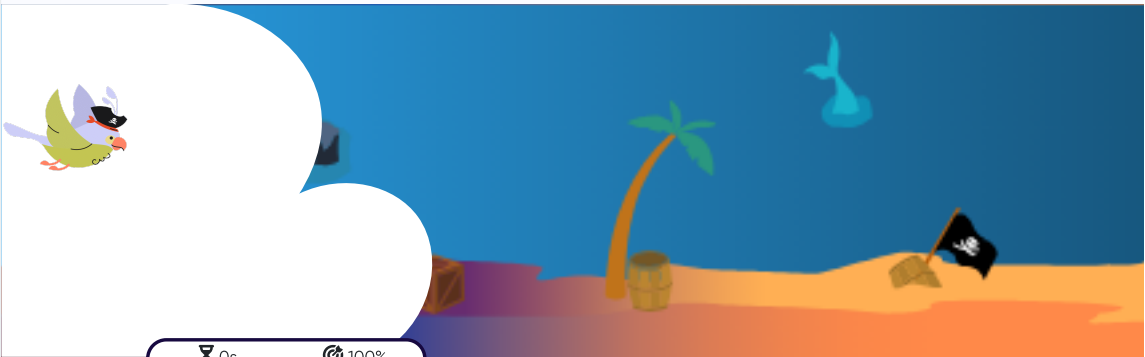
Full Screen

Audio

Instructions

Restart

Pause



0s 100%

Sensing blocks help us se

2. Sensing Memory Game

Full Screen

Audio

Instructions

Answer Key

Pause

Flips: 0

1

2

3


4



5

6

7

8





3. Sensing Matching Game

Full Screen

Audio

Instructions

Answer Key

Pause

Clear All

Check Matches

Attempts: 0

1

2

3


4



Senses if the space key is pressed

Senses if the sprite is touching the color green.

Senses if two colors are touching

Senses if the sprite is touching the mouse cursor.





Blocks Pro Challenges (4)

1. Hi There

Challenge

Textbook

Hi There

Select 2 sprites. Program one sprite so when the green flag is clicked, it will move across the screen. Do this using the motion block and a forever loop.

Then, program the sprite to say hello IF it is touching the other sprite.

Requirements0/5

1

Motion

1

Looks

1

Event

2

Control

1

Sensing

Blocks must be connected to an Event block in order to pass the requirements

Code

Costumes

Sounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

Sprite1

x0y0

Size100Direction90

Sprite1

Stage

Backdrops1

2. Growing Potion

Challenge

Textbook

Growing Potion

Imagine your character just drank a potion. Program your character sprite to walk across the screen and when it touches the potion sprite it will grow in size. Use the 'change size by 10' block to do this.

Requirements0/5

1

Motion

1

Looks

1

Event

2

Control

1

Sensing

Blocks must be connected to an Event block in order to pass the requirements

Submit

Code

Costumes

Sounds

Motion

move 10 steps

turn 15 degrees

turn 15 degrees

go to random position

go to x: 0 y: 0

glide 1 secs to random position

glide 1 secs to x: 0 y: 0

point in direction 90

point towards mouse-pointer

change x by 10

set x to 0

change y by 10

set y to 0

if on edge, bounce

set rotation style left-right

x position

Sprite1

x0y0

Size100Direction90

Sprite1

Stage

Backdrops1

3. Food Collector

Food Collector

Create a game where one sprite needs to collect the different foods around the screen.

First, program the collector sprite so it responds to all arrow controls (up, down, left, right). Then, program it so when the green flag is clicked, the sprite will go to the XY coordinate (0, 0).

Then, place at least 4 food sprites in different locations of the scene. Program each food sprite so when the green flag is clicked, it will hide if it is touching the collector sprite. Note: The IF control block must be inside the forever loop block.

Requirements 0/5

- 5 Motion
- 4 Looks
- 9 Event
- 8 Control
- 4 Sensing

The code editor shows the following blocks in the Motion category:

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point towards mouse-pointer
- change x by 10
- set x to 0
- change y by 10
- set y to 0
- if on edge, bounce
- set rotation style left-right

The sprite is a cat character, and the stage is empty.

4. The Floor is Lava

The Floor is Lava

Build a game where the floor is lava! Use the drawing tool to draw lava all around your scene. Create a path through the lava.

Choose a sprite to be your player. This sprite should respond to all arrow controls (up, down, left, right).

Move your sprite across the scene and program it to start over at its beginning XY coordinates if it touches the lava color.

Requirements 0/4

- 5 Motion
- 4 Event
- 1 Control
- 1 Sensing

Blocks must be connected to an Event block in order to pass

The code editor shows the following blocks in the Motion category:

- move 10 steps
- turn 15 degrees
- turn 15 degrees
- go to random position
- go to x: 0 y: 0
- glide 1 secs to random position
- glide 1 secs to x: 0 y: 0
- point in direction 90
- point towards mouse-pointer
- change x by 10
- set x to 0
- change y by 10
- set y to 0
- if on edge, bounce
- set rotation style left-right

The sprite is a cat character, and the stage is empty.

Answer Keys & Solutions

Questions

1. True or False: Sensing blocks help us see how things in a project work together.

MULTIPLE CHOICE

Correct Answer:

A. True

✓ Correct

B. False

✗ Incorrect

Explanation:

Sensing blocks help program the computer to know everything that is happening in your program.

2. Click the two examples of sensors below:

SELECT MULTIPLE

Correct Answers:

A. when touching the edge

✓ Correct

B. when touching a color

✓ Correct

C. when variables are used

✗ Incorrect

Explanation:

Variable blocks are different from sensing blocks.

3. Which sensing block can detect when a sprite touches the edge of the screen?

MULTIPLE CHOICE

Correct Answer:

A. Touching color

✗ Incorrect

B. Touching edge

✓ Correct

C. Touching sprite

✗ Incorrect

Explanation:

Think about the example provided where the sprite bounces back when it hits the edge.

4. What do you use to move a sprite?

MULTIPLE CHOICE

Correct Answer:

A. Sound blocks

✗ Incorrect

B. Pen blocks

✗ Incorrect

C. Looks blocks

✗ Incorrect

D. Motion blocks

✓ Correct

Explanation:

Consider the category of blocks used to control the movement of sprites.

5. What happens when the condition in an "if then" block is true?

MULTIPLE CHOICE

Correct Answer:

A. The program stops running

✗ Incorrect

B. The program ignores the block

✗ Incorrect

C. The code inside the "if then" block runs

✓ Correct

D. The sprite changes color

✗ Incorrect

Explanation:

Consider the purpose of conditional statements like "if then."









Games

1. Sensing Typing Game

Typing game - no answer key needed. Students practice typing the provided content.





2. Sensing Memory Game

Memory Game Pairs:

1.  A blue code block that says "color blue is touching color green" ↔  A blue code block that says "color blue is touching color green"
 2.  A blue code block that says "key space is pressed" ↔  A blue code block that says "key space is pressed"
 3.  A blue code block that says "Touching mouse pointer." ↔  A blue code block that says "Touching mouse pointer."
 4.  A blue code block that says "Touching color green" ↔  A blue code block that says "Touching color green"
- Students must find all matching pairs by flipping cards and remembering their positions.*

3. Sensing Matching Game

Matching Game Solutions:

1.  A blue code block that says "Touching mouse pointer." →
2.  A blue code block that says "Touching color green" →
3.  A blue code block that says "color blue is touching color green" →
4.  A blue code block that says "key space is pressed" →

Students must drag items from the left to match with corresponding items on the right.