WELCOME

Block-based Programming With Scratch

The Ninja Turtles

Anh Tang
Julissa Valdez-Ramos
Jean Makita Mbama

PRE-SURVEY

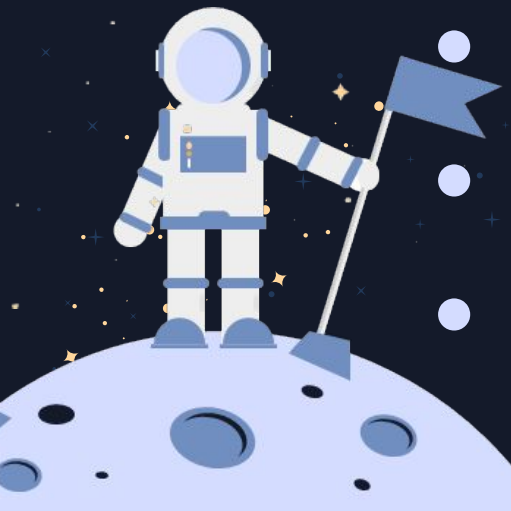
ABOUT TAP

Technology Ambassador Program-

The Technology Ambassadors Program (TAP) aligns with GGC's four pillars – scholarship, leadership, service, and creativity – with a focused commitment to service learning. This award-winning program also addresses a need to increase the number of students who persist in an IT major or IT minor, particularly those underrepresented in computing.

Let's ask Dr. Anca and Dr. Robertson

Project Goals



- Teach some fundamental rules of block coding and programming language.
- Have students be able to solve basic coding problems on their own after our workshop.
- Show a free, fun, and understandable way to encourage technological participation. Even at home!
- Create a game that is simple enough to use as a demonstration of simple coding, but also fun and filled with knowledge
- Engage participation and logical thinking

WHAT IS OUR PROJECT?

A crash course on programming with Scratch.

Through this workshop experience, we hope to help you develop your tool set to **build your own games** or **interactive stories** in Scratch.

In addition to that, you will be learning how to develop an **algorithm** using variables, if-else statements, loops, etc. which is a skillset that can be applied to other programming languages, not just Scratch.

Why Scratch?

Scratch is a programming language developed by the MIT Media Lab as an educational tool aimed primarily at children.

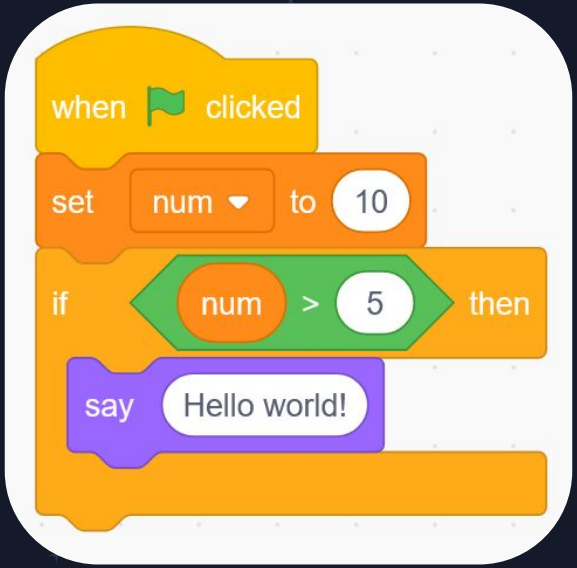
Block-based coding

Drag-and-drop interfaces

Beginner-friendly

Scratch is a block-based programming language.

Block-based programming



Text-based programming

```
1  public class Main {
2      public static void main(String[] args) {
3
4          int num = 10;
5
6          if (num > 5) {
7              System.out.println("Hello world!");
8          }
9
10     }
11 }
```

About our project...

- We created a simple space-themed role-playing game in Scratch.
- In the game, players are given tasks such as fixing computer screens or opening up treasure chests.
- In order to complete these tasks, the player must answer questions about fundamental programming concepts such as variables, if-else statements, and loops.

Makey Makey

You will be learning to set up a Makey Makey and how to use it to control character movement in the game without using the keyboard.

Algorithms

Definition: A set of instructions that must be followed step-by-step.

Variables

A container that holds a value. This value can change throughout the program, hence the name “variable”.

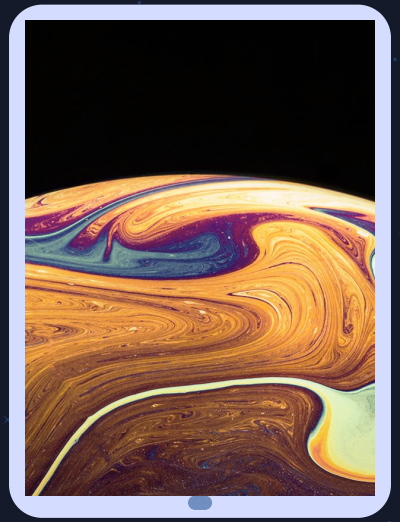
If-else statement

Conditional statement used for decision-making.

Loop

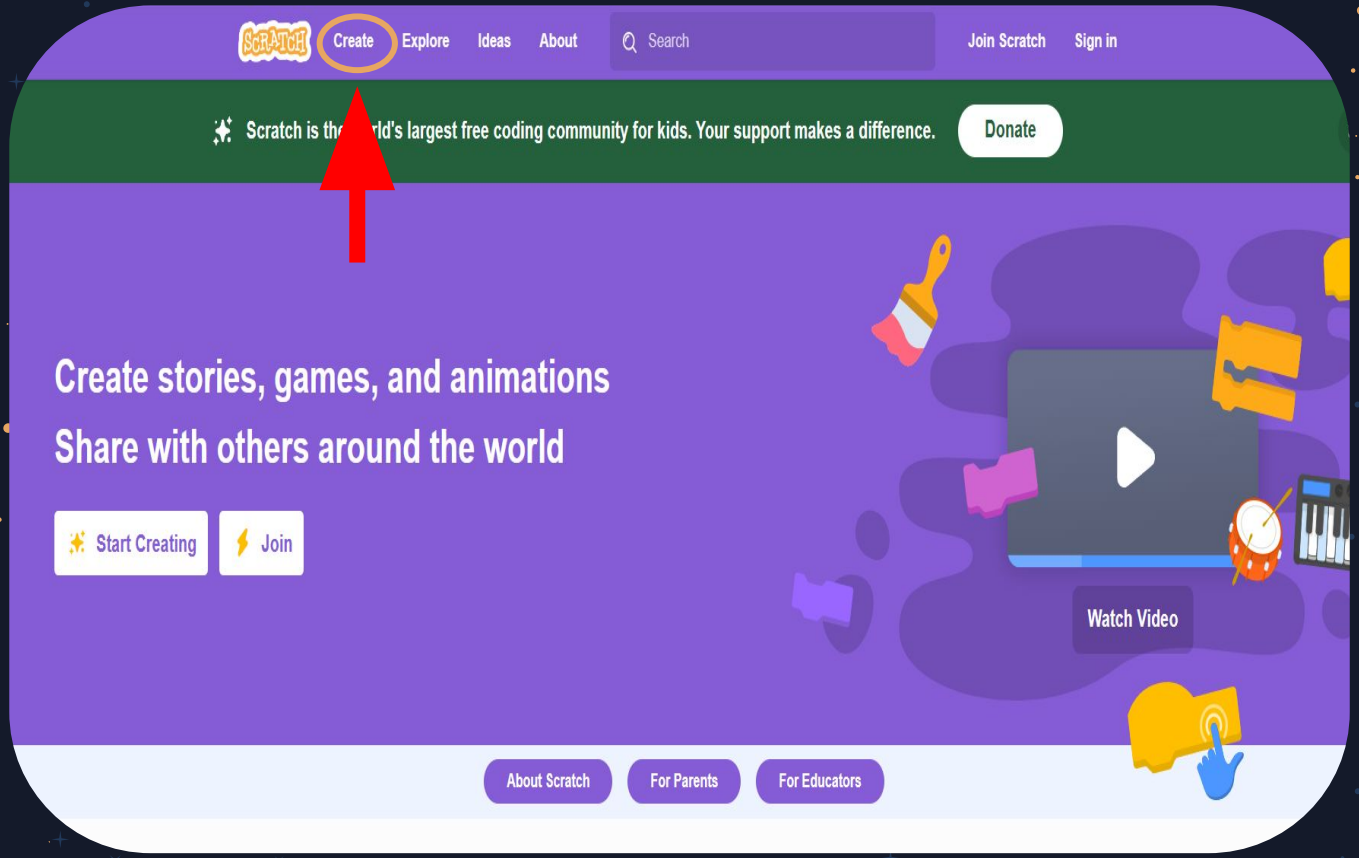
Used to repeat a block of code until some specified conditions are met.

Let’s open Scratch!

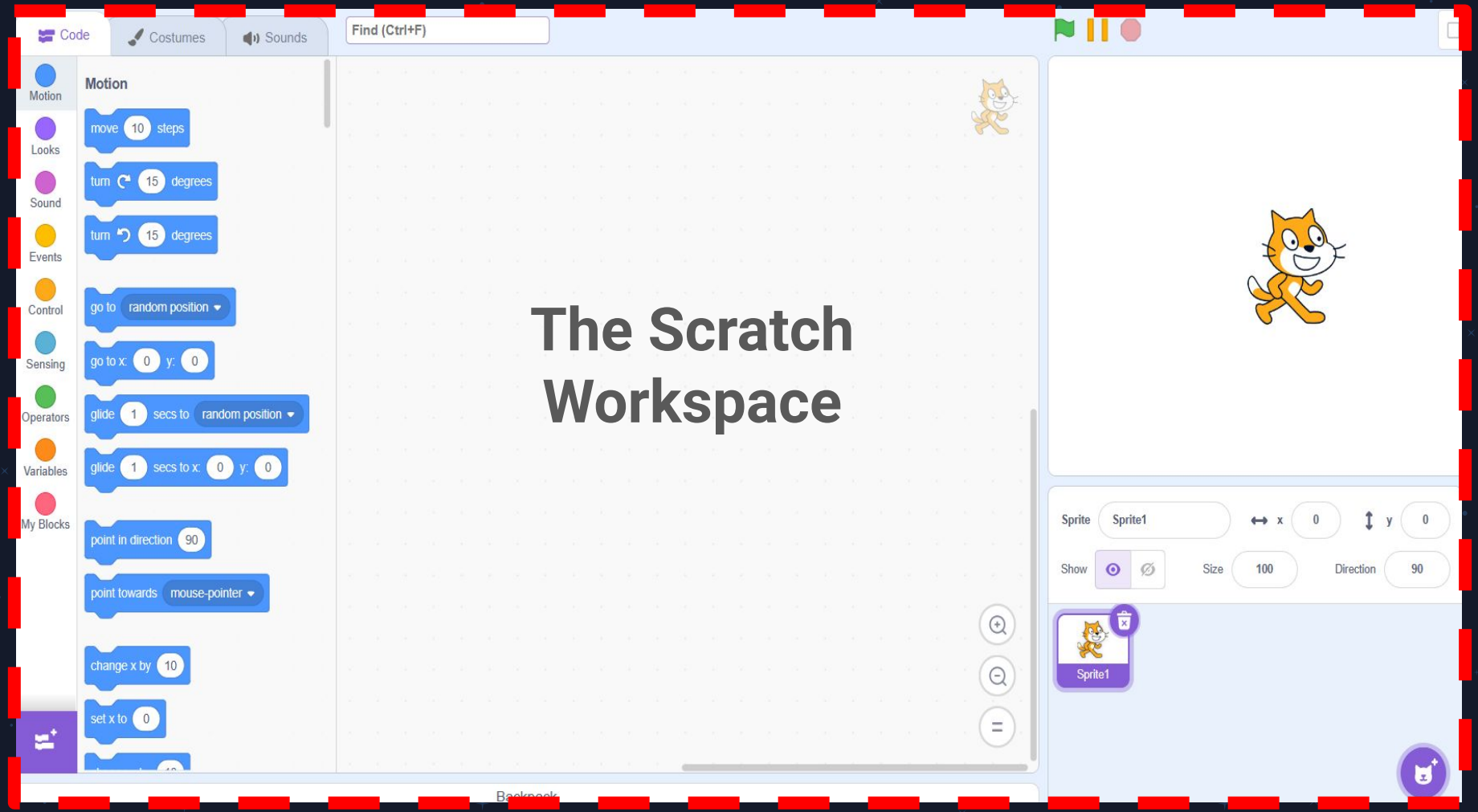


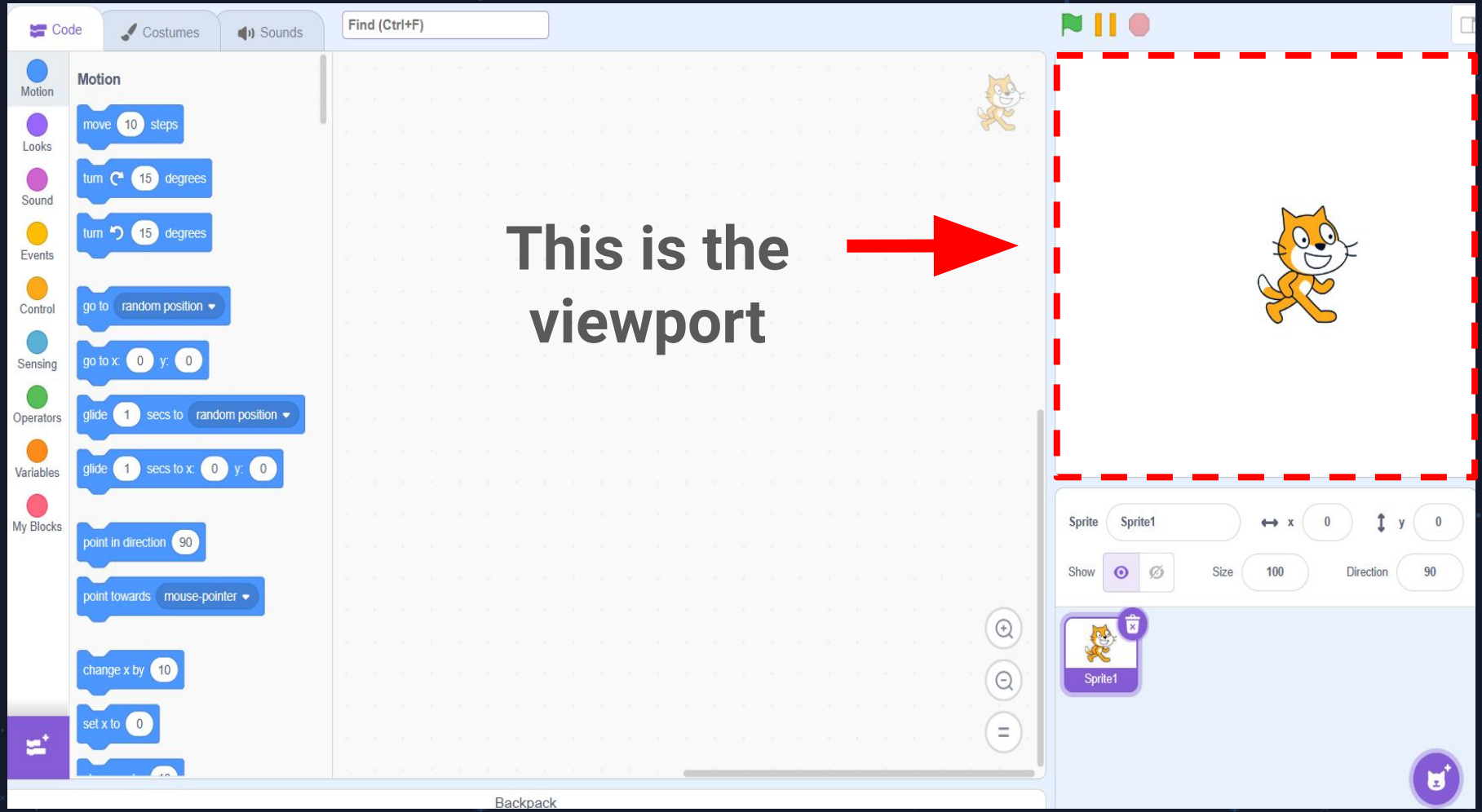
Link: https://scratch.mit.edu

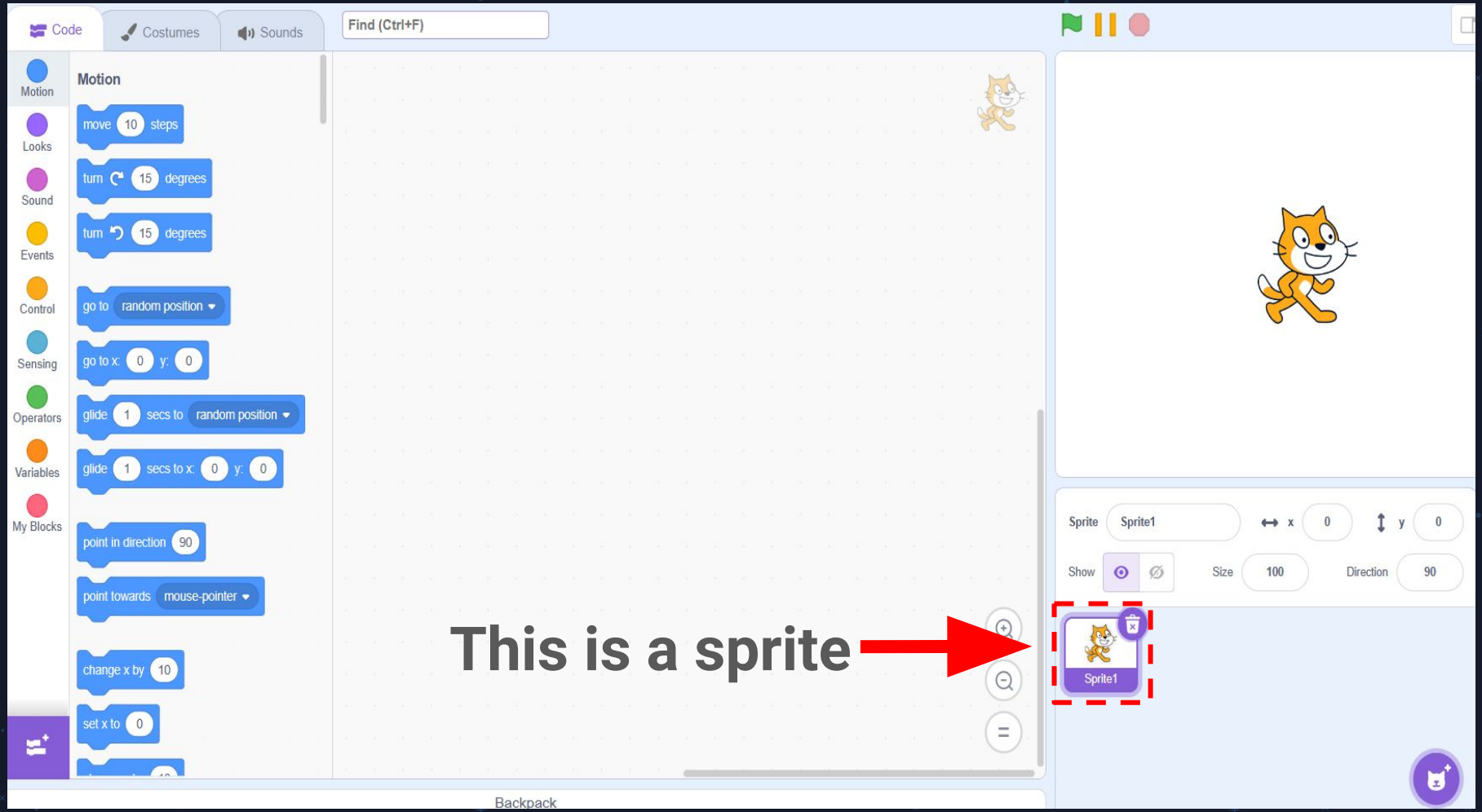
Or type ‘Scratch’ into your browser search bar!

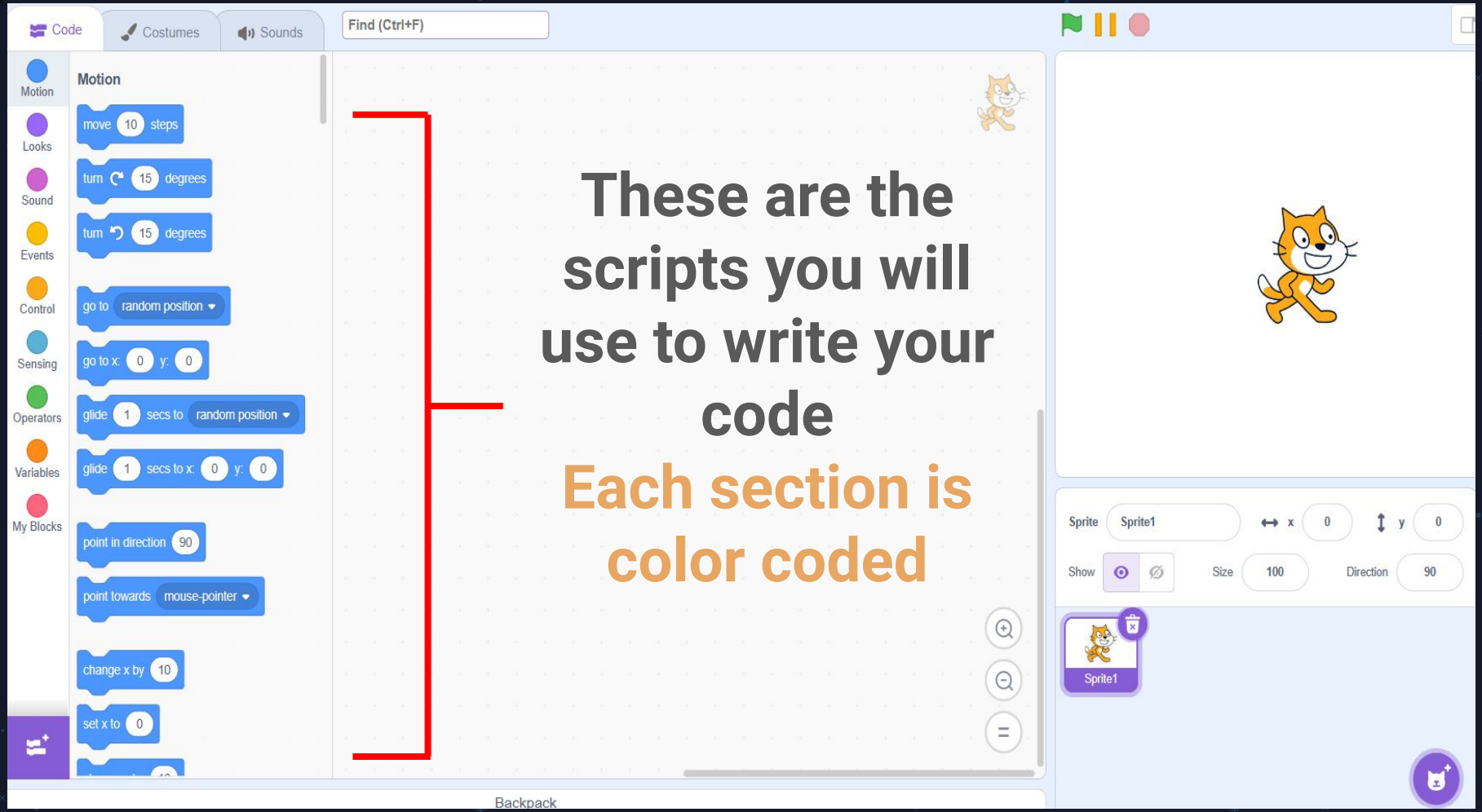

Create
Explore
Ideas
About
Search
Join Scratch
Sign in
Scratch is the world's largest free coding community for kids. Your support makes a difference.
Donate
Create stories, games, and animations
Share with others around the world
Start Creating
Join
Watch Video
About Scratch
For Parents
For Educators

The Scratch Workspace

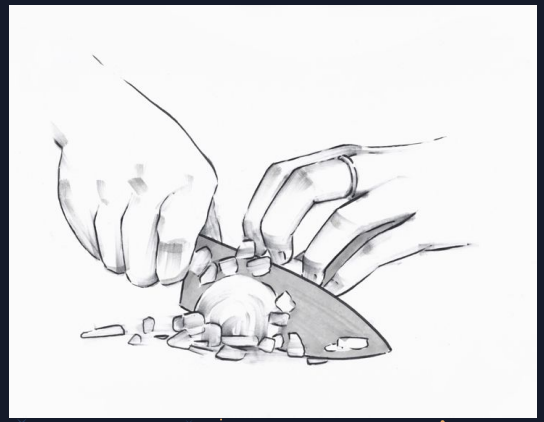



Code
Costumes
Sounds
Find (Ctrl+F)
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
This is the viewport
Sprite Sprite1
x 0
y 0
Show
Size 100
Direction 90
Sprite1
Backpack


Code
Costumes
Sounds
Find (Ctrl+F)
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
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
This is a sprite
Sprite
Sprite1
x
0
y
0
Show
Size
100
Direction
90
Sprite1
Backpack

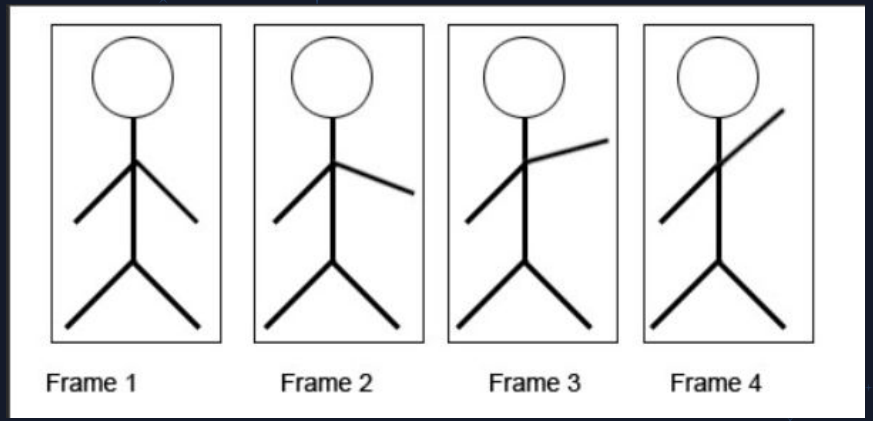

Code
Costumes
Sounds
Find (Ctrl+F)
Motion
Looks
Sound
Events
Control
Sensing
Operators
Variables
My Blocks
Motion
move 10 steps
turn 15 degrees
turn 15 degrees
go to random position
go to x: 0 y: 0
glide 1 secs to random position
glide 1 secs to x: 0 y: 0
point in direction 90
point towards mouse-pointer
change x by 10
set x to 0
These are the scripts you will use to write your code
Each section is color coded
Sprite Sprite1
x 0
y 0
Show
Size 100
Direction 90
Sprite1
Backpack

ANIMATING PROPS



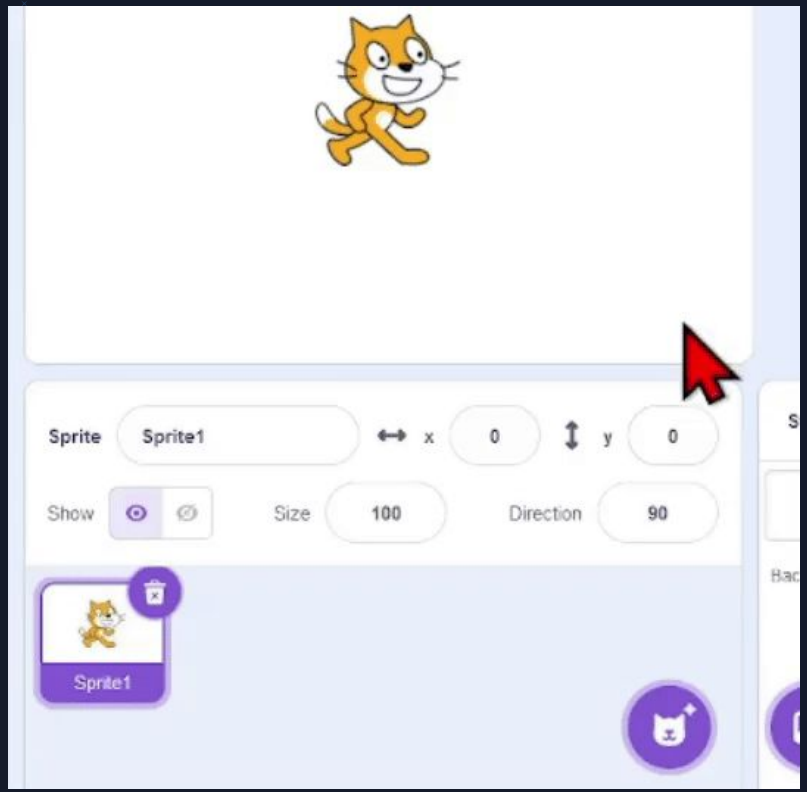
How does animation work?

Animation is a process in which still images appear in rapid succession to create the illusion of movement.



HOW TO ANIMATE PROPS IN SCRATCH?

- Create a sprite with at least 2 costumes.
- Select the next costume block from the Looks category.
- Add the wait block from the Control category.
- Put everything together in a forever block (Control category). This will create a seamless animation effect.



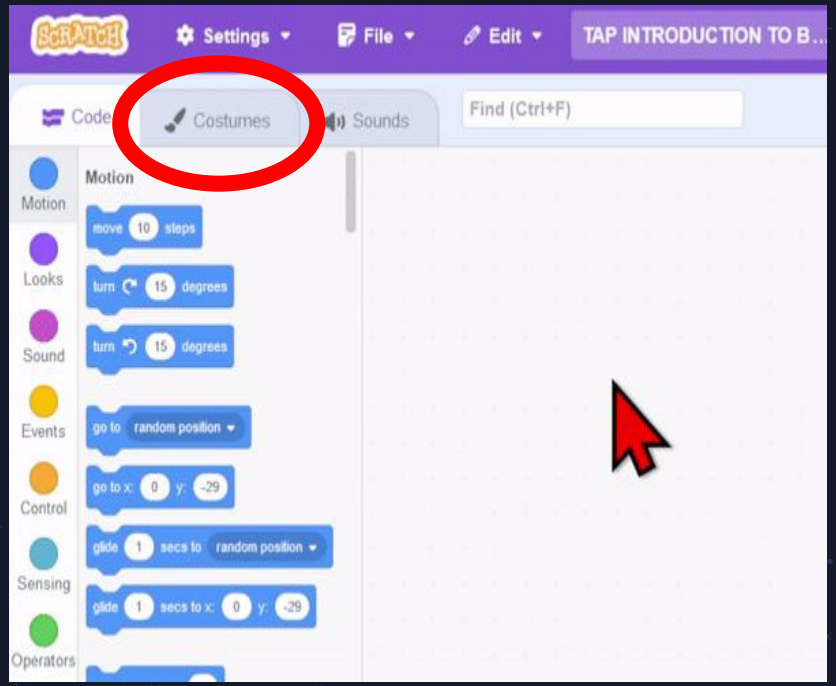
Now we're going to create our Own animated sprite!

Step 1: Hover over the Cat icon in the bottom right corner of your workspace

Step 2: Click on the Paint icon

A new sprite named "Sprite2" will appear.

Click on the Costumes tab in the top left corner



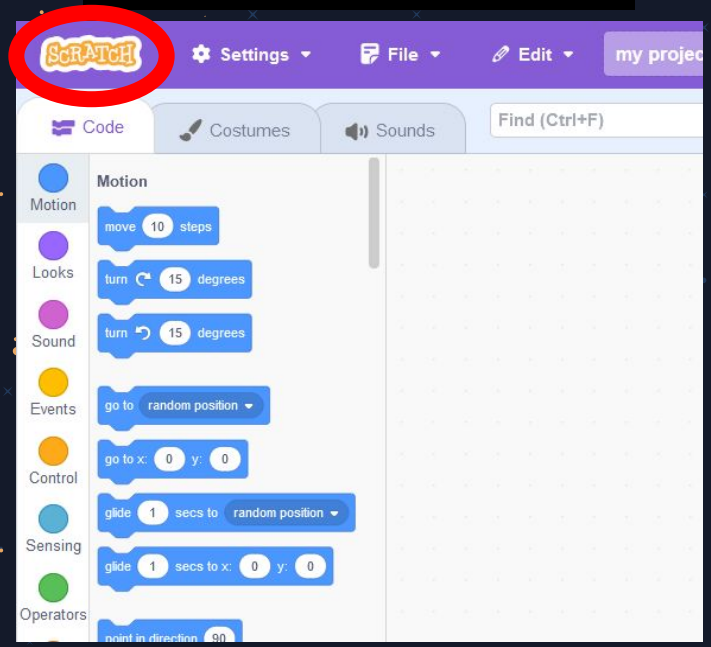
Select the paint brush tool

This is the code/script we will use to animate our new sprite

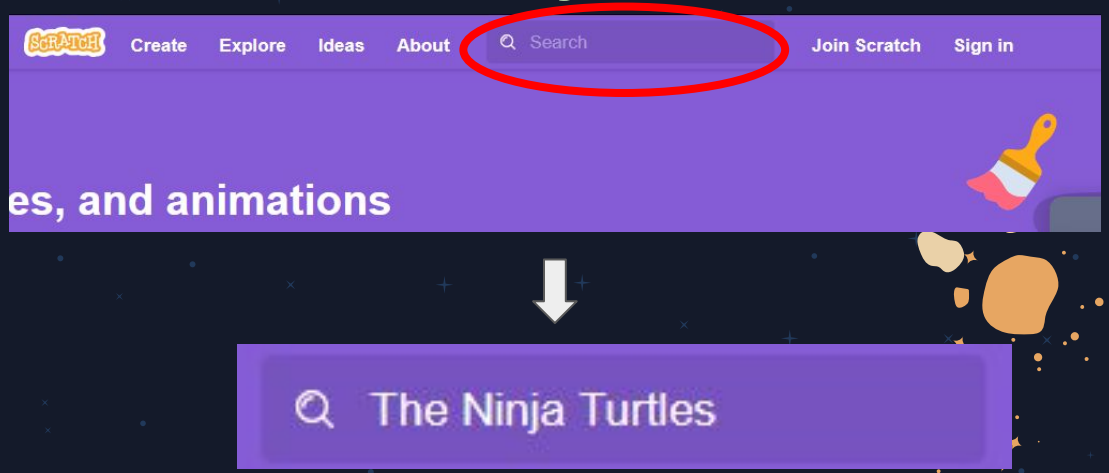
Notice how it's in the ***forever*** block?

- What would happen if it wasn't inside the forever block?
- What would happen if it was in a repeat (2) time(s) block?

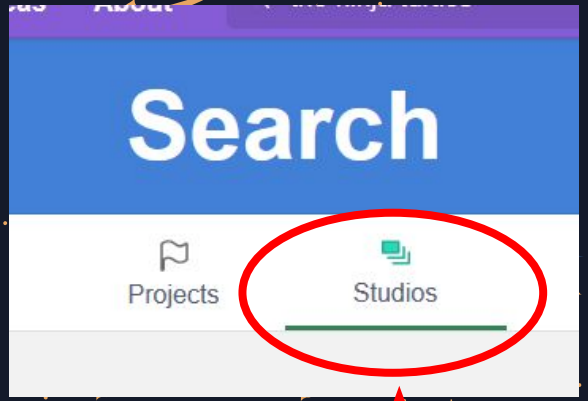
Click on the Scratch Logo to go back to the Home Menu



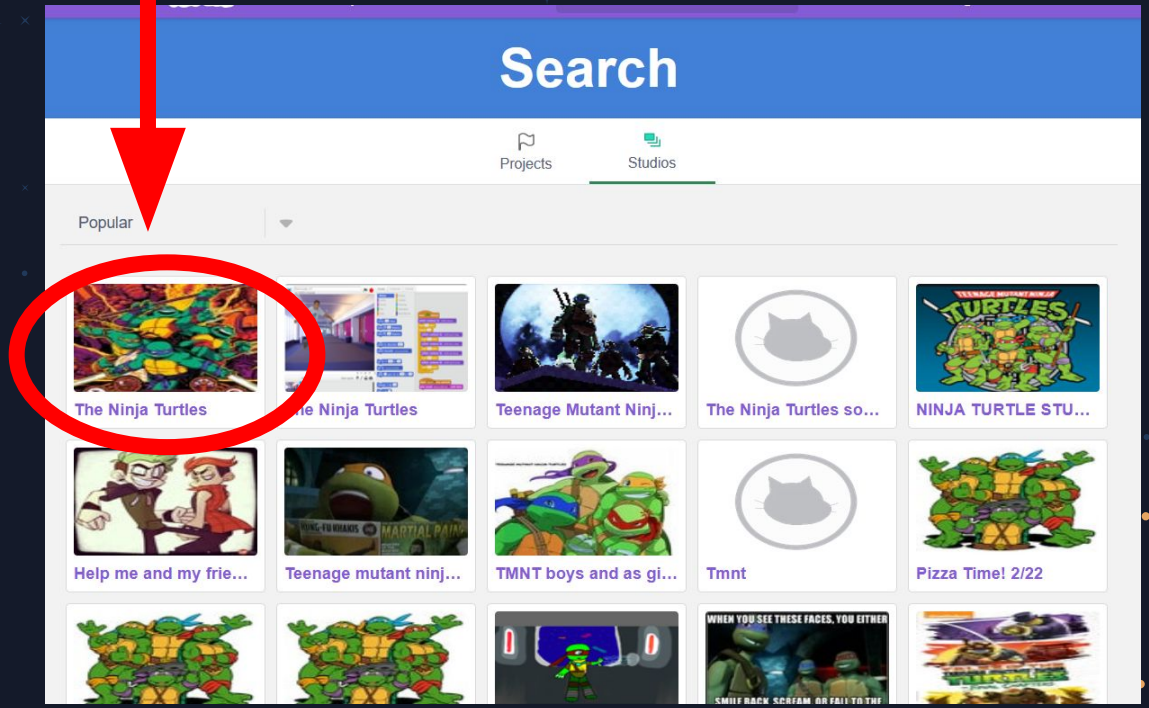
Using the Search Bar, type in "The Ninja Turtles"



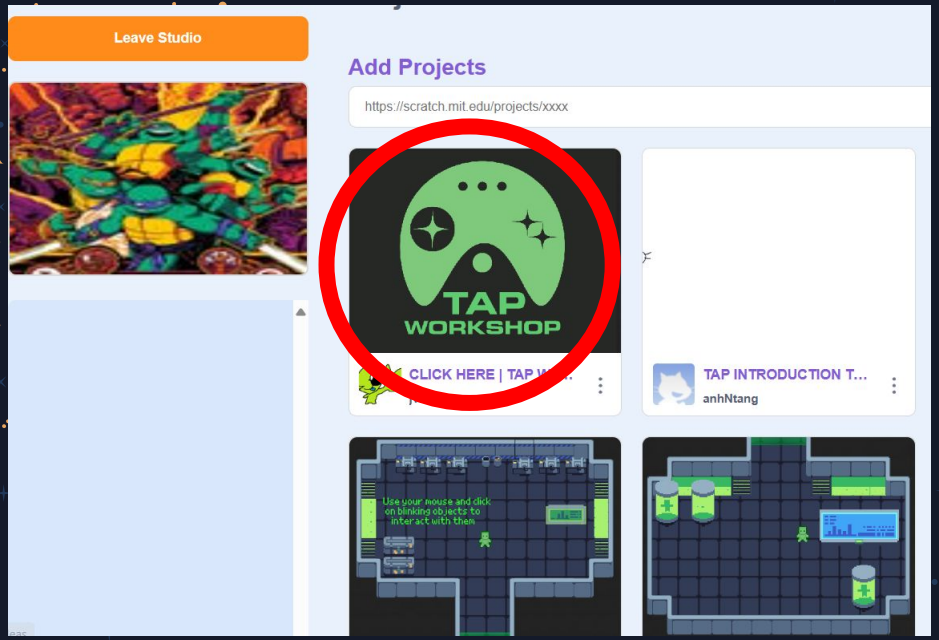
Click on Studios



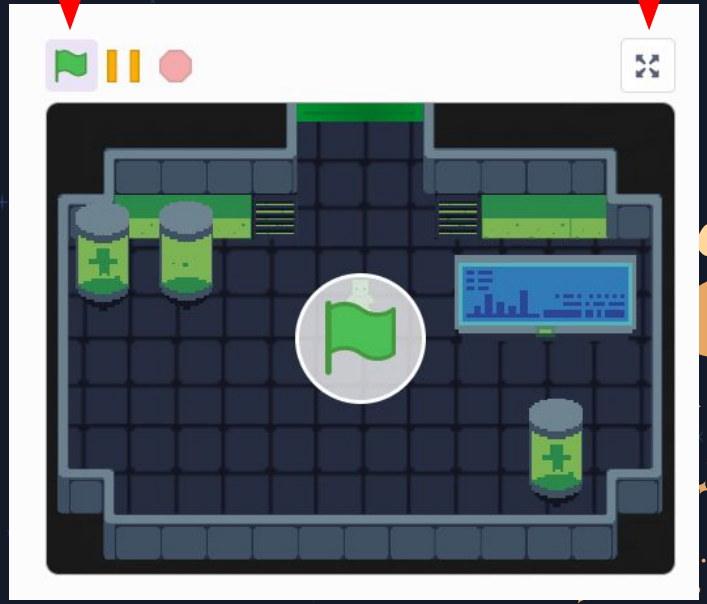
Select this Studio



Click the first project

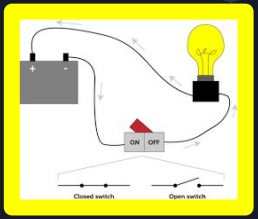


Full screen, then press the green flag



Makey Makey?

- An invention kit designed to connect everyday objects to computer keys.
- Using a circuit board, alligator clips, and a USB cable, the toy uses closed loop electrical signals to send the computer either a keyboard stroke or mouse click signal.
- In other words, the Makey Makey allows you to “make” anything that conducts electricity into a controller.
- Although the Makey Makey replaces the arrow keys, you must still program Scratch to tell it what those keys do!



PLAY THE GAME!

FIRST ROOM: this room contains block-code related questions based on the previous lesson

Everyone will get 10 minutes to complete the questions. If you have time left, you can move onto the second room.

SECOND ROOM

When you enter the second room, take a look at the monitor and one of the tanks.

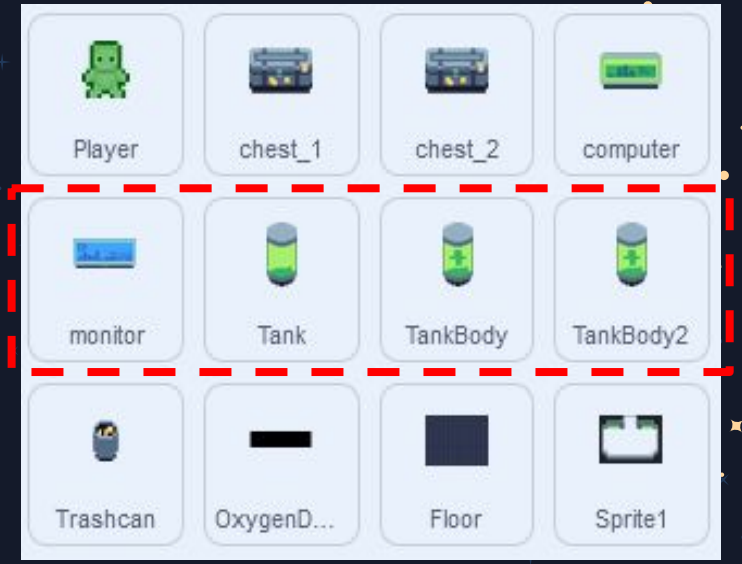
These sprites are **frozen**!

Try and figure out what went wrong.

Exit full screen and click on ‘See inside’

The bottom right corner of the workspace is dedicated to our sprites.

Identify which sprites are having issues and click on whichever one you'd like to debug (fix) first

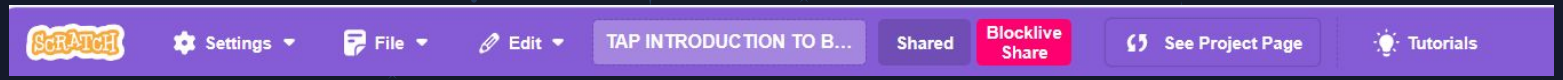


Bonus Task #1: Re-Animate the sprites in the second room.

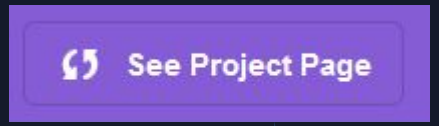
Bonus: Add in your own question. Only when the answer is correct that the sprite animation starts.

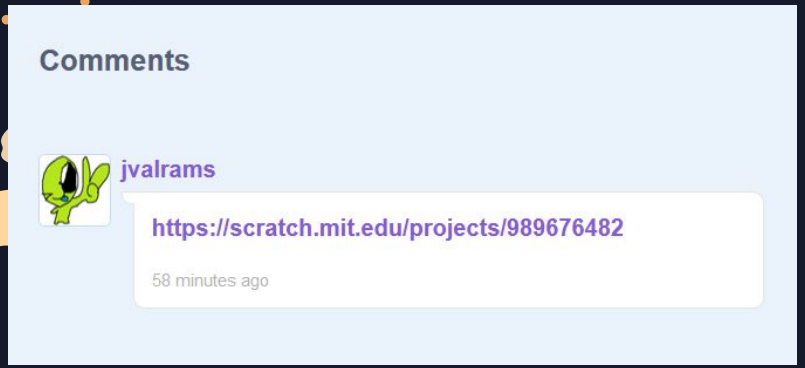
Psst...remember this script?

BONUS TASK #2



Click on the 'See Project Page' button on the middle-top section of the screen

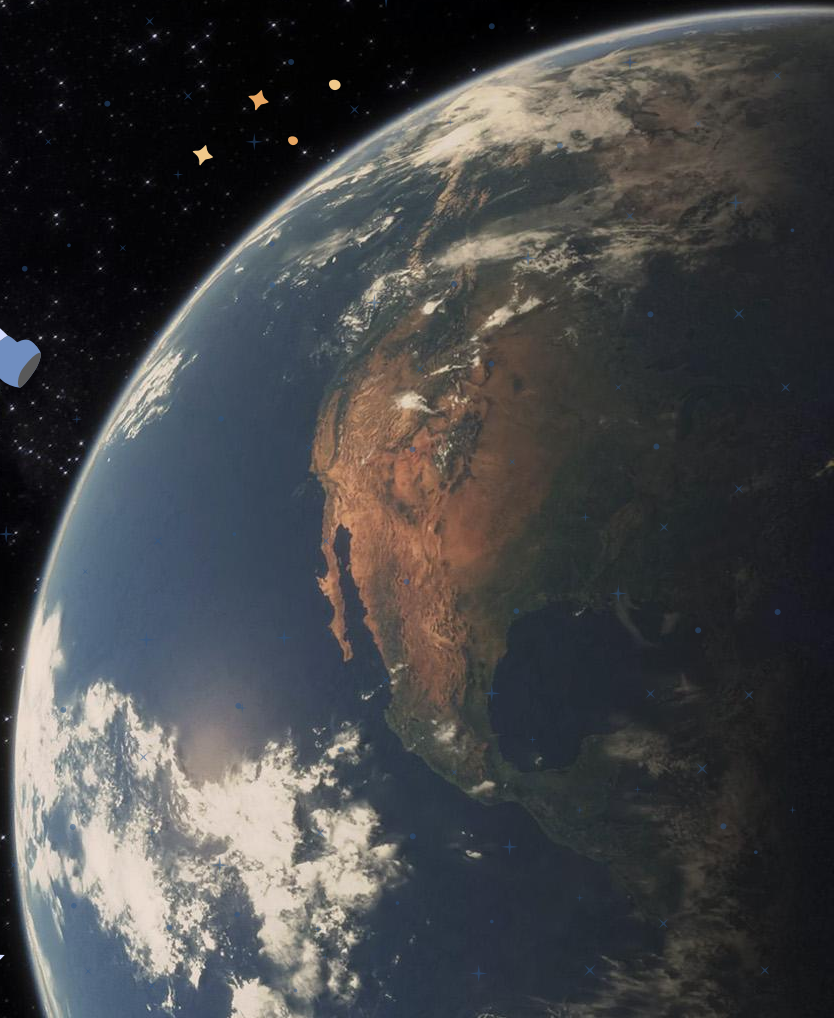




Scroll down until you see the Comments section.

Click on the link, press the green flag, and take note of our new bugs (errors)

Character Movement



Character Movement

Objective: Turn keyboard inputs to character movements on the screen. One way to program character movement is to use the code blocks from the Sensing category (blue blocks).

- The Sensing code blocks allow for keyboard inputs to be converted to integer value 1. Therefore, when the right arrow key is pressed, the character can move to the right 1 unit.

- To enable character left movement, subtract the left arrow key (which has an integer value of 1) from 0, which yields -1.
- Now, the character can will be able to move to the left 1 unit.
- The same logic applies to vertical movement.

TASK:
CURRENTLY, THE CHARACTER PROGRAMMED TO MOVE LEFT AND RIGHT. PLEASE COMPLETE THE CODE FOR UP AND DOWN MOVEMENT.

POST SURVEY

If there's time left, we'll walk through the collision model!

COLLISION MODELS

Wall Collisions in Scratch

Requirements:

- Movement
- Object that is moving
- Object that is not moving

Why do we code collisions?

When coding a game model/program the computer cannot differentiate one line in a layer from another. Effectively, your computer does not know that a ball should bounce or a person can walk on flat surfaces. This could result in your character walking straight through the map and the objects created in it.

A picture is worth a thousand words

How do we solve this?

One of the simplest ways to solve this is to tell the computer the object that is moving cannot move through an object from a specific layer or the map sprite. This works so well in scratch because the creation process involves creating multiple objects that you can program separately. That ability allows you to tell the computer “This object cannot overlap solid lines of this object”.

Presentation Notes

- Algorithms: A set of instructions that must be followed step-by-step.
- Variables: A container that holds a value. This value can change throughout the program, hence the name “variable”.
- If-else statement: Conditional statement used for decision-making.
- Loop- Used to repeat a block of code until some specified conditions are met.
- Animation is a process in which still images appear in rapid succession to create the illusion of movement.
- One way to program character movement is to use the code blocks from the Sensing category (blue blocks).
- The Sensing code blocks allow for keyboard inputs to be converted to integer value 1. Therefore, when the right arrow key is pressed, the character can move to the right 1 unit.

THANK YOU FOR PARTICIPATING!

Any questions about TAP or our project?