Things to keep in mind before class

1. Create your account on repl.it an p5 editor and be ready with the user id and password.
2. Explore the platform thoroughly.
3. Keep the user id and password simple, because you might have to share the same with the student.
4. Open all the teacher activity links in different tabs of your browser.
5. While sharing screen please keep in mind that the tab containing this script must not be visible to the student.
6. Share one tab at a time and be very careful during this process.
7. Don't keep any extra tabs in your browser.

Things to keep in mind while using this script

1. Ask for the grade of the student and gauge the calibre of the student while in "Warm-up." Ask the student if he/she has prior knowledge of coding.
2. All the activities are mandatory.
3. Don't just read the dialogue, build upon it and have a conversation with the student.
4. Keep an eye on the action section for cues on how to proceed the conversation.
5. Please be as descriptive as possible in your teaching.
6. If you feel something is missing in the script about a concept, you can very well build upon it and explain the student according to you.
7. Just keep in mind the flow of the script, build upon it, not deviate from it.

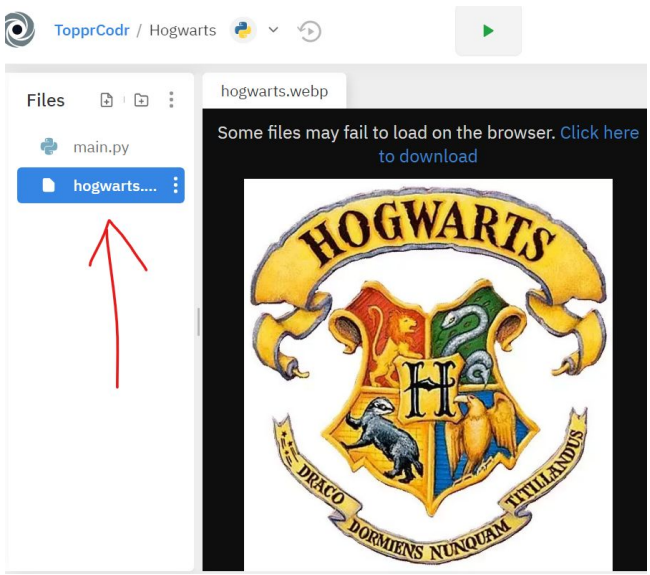
Session Code	CODR-912-TC-000
Module	Demo
Teaching Unit	Python and Game Development using JS
Learning Outcome	Coding in Python and creating a game using JS
Resources	Teacher: 1. Login credentials to the repl.it 2. Login credentials to the p5 Editor 3. A laptop along with audio and video exchange. 4. Notebook and Pen (To note any development from the session) Student Resources 1. Login credentials to the activity platform.

	2. Laptop along with audio and video exchange 3. Notebook and Pen(To keep note of important parts in the session)
Duration	50 Mins

Structure	**Warm-up** **Pace-up Activity** **Knowledge Transfer I** **Knowledge Transfer II** **Student Led Activity II** **Wrap up**	2 Mins 3 Mins 10 Mins 10 Mins 20 Mins 5 Mins

Section	Content	Prompt
Warm up (2 Mins)	Hi, my name is xxxxxxxxx. What is your name? How old are you? In which grade are you studying? Are you interested in taking programming as a career option? Is there anything you would want to create once you learn programming? Have you done programming before ?	Confirm if the student is able to see and hear you clearly. Motivate the student & explain how coding is the need of the hour.
Interaction (3 Mins)	What is programming or coding? Can you give me some examples of programs that we use in everyday life? Starting from a small calculator to satellites and rockets all are programmed to perform their task.	*Sample example: Banking apps, e-commerce apps, social media apps, games all are a few examples of programs.*

	How can you explain coding in layman language? Sample Answer: Set of instructions or commands that you give to the computer to perform a specific task or activity. We give these instructions in a specific language. Can you give an example of programming languages ? Sample example: Java, Javascript, C, C++, Python etc. In today's class you are going to create 2 programs using the top 2 programming languages in the world: Python and Javascript. Let's start with Python.	If the student can't answer, you may prompt using the sample answers.
Knowledge Transfer I (10 mins)	Ok before that, tell me do you have houses in your school? Houses are like teams in which you belong. All competitions are played between these houses or teams, mostly on sports day or annual day. As long as you are part of the school you will be part of that house too. Do you know how a student is given a house? Most often the house is chosen randomly, so that each house has different types of students.	Talk to the student about their school and house system. Let the student share their story. If the student is not aware of the house then tell them that your school had a house system, with red, yellow, green and blue houses.
• **Share your screen and open teacher activity 1**		
	Let's write a small program in python to sort students in different houses. Ok, the houses I am going to create are quite famous. Have you heard of Hogwarts?? It is a school of Magic in the Harry Potter movies. There are four houses in Hogwarts: Gryffindor, Slytherin, Ravenclaw and Hufflepuf.	Teacher Activity 1: Hogwarts Ask if they have seen the movie, if not then send the name of houses in the chat box while saying them.



Use the hogwarts tab and show the logo and the four animals representing the four houses respectively.

Explain the platform a bit: place where you code, the output screen and the run button.

We have to code such that whenever we ask for a house, Python selects any house name out of the four randomly.

For that we will first store the name of four houses in the computer's memory.

To store something we usually need boxes, right?

Explain the meaning of variables very patiently.

Similarly, to store something in a computer's memory we need a box or space with some name to recognize it.

Such a box is known as a variable.

I have created a variable named **hogwarts** and have stored the names of the four houses in it.

Tell them not to worry about the brackets and the quotes, we will study about them gradually.

```
hogwarts=["Gryffindor","Slytherin",
"Ravenclaw","Hufflepuf"]
```

The four names are items or elements of variable **hogwarts.**

Now, to select a house randomly, we first import the random library by using the command.

Explain how easy the syntax is as it sounds almost like english.

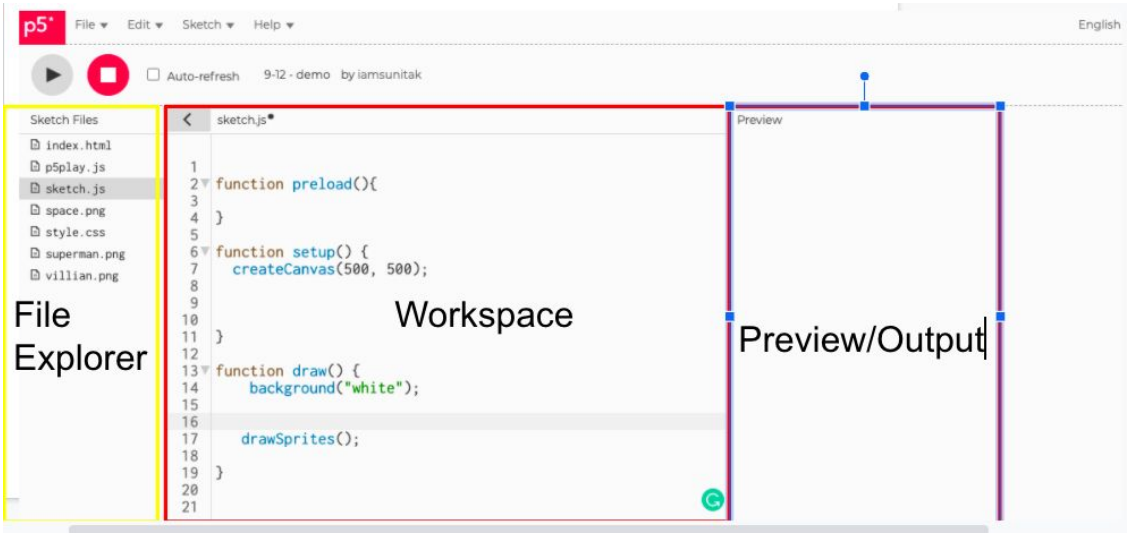
```
import random
```

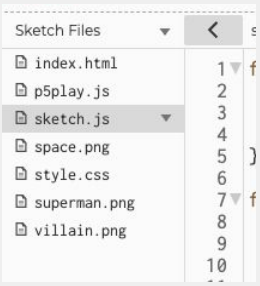
This will tell Python that we have to do things

	randomly. **Import random** is actually loading the ability of selecting things at random. Now we need another variable to store the **randomly selected house**. Let's name it **house** only for ease. `house=` To select a random element from the variable **hogwarts,** we need the function **random.choice()** `house=random.choice(hogwarts)` In the brackets of the function we put in the items from which we need to make a random choice. So when we run this then an item from the variable **hogwarts (**i.e, four house names stored in it) will be chosen at random and stored in the variable **house**. Let's check the output. To show the value of house, we need another statement called print `print(house)` Every time we run this program, we get a different house. We can even print a message along with the house to make it better. `print("Welcome to Hogwarts! Your house is")` `print(house)` Now let's see if you were to go to Hogwarts to learn magic, which house would this program put you in. We will edit the program to ask the student's name in the beginning. Let's see how easy it is to do so. `name=input("What is your name? ")` Here, again we have created a variable **name** to store the name of the student and in the variable we have the **input function**. So let me tell you what the **input function** does.	Refer to the variable **hogwarts** that they have created earlier. Again, point out how easy the syntax sounds in english.  Click on the run button 5-6 times to show that the output is different every time. Remind creating a variable and using a function to put some value in it.

	It makes the program **collect value from the user**. And the **message in the quotes** will be displayed on the screen as instruction as to what to input. Now that we know the student name we can display both the **name** and the **house** predicted by the program as output. Let me show you how. The variable storing predicted house is **house** and the variable storing student name is **name.** We can simply use the print functions in the end to display the value of these variables. `print("Welcome to Hogwarts!",name)` `print("You are in",house)`	Ask what the student wants in the message, giving them this example. Actively use student's suggestions in the message
	Sample Program `name=input("What is your name? ")` `import random` `hogwarts=["Gryffindor","Slytherin", "Ravenclaw","Hufflepuf"]` `house=random.choice(hogwarts)` `print("Welcome to Hogwarts!",name)` `print("You are in",house)`	
	Let's finally run the program and see the output. Congrats! *StudentName* You're in the house of *HouseName* of Hogwarts! Did you realize how easy it is to code in Python. Now let's create a fun game using Javascript. Before that, let us see the game we will create today.	Put in the student name in the console and show the final output. Congrats the student on whatever house they get.

• Stop your screen sharing • Help the student to share their screen • Take the student to chat box		
Student Led Activity (1-2 mins)	I am sharing a link with you on the chat box. I want you to click on the link and play the game. *Guide the student to play the game.* We have Superman chasing the villain in space. Both of them are flying upward in the sky. Or are they? Actually both of them are fixed and the moving background is creating the illusion of them moving upward. Now, when you click on the upper arrow key on the keyboard the superman flies upward, faster. If you click on the right arrow key along with the upper arrow key you can make Superman catch the villain. See what happens when the villain is caught. Boom! It disappears.	Share the link with the student in the chat option. Teacher Activity 2: Final Superman Game The arrow keys on the keyboard are used to control the superman.
• Help the student stop sharing screen • Share your screen and open Teacher Activity 3		
Knowledge Transfer II (10 mins)	Are you excited to create your own game? Great let us start. We will be using an online IDE called as p5 editor for this. When we click on the arrow below the play button, the third window appears on the left.	Teacher Activity 3 Show the three sections of the IDE 



	The three sections of p5 editor are :- ❖ **File explorer** : To see all the files in the program. You will understand these files as and when you learn more coding. ❖ **Workspace** : To write the program code. ❖ **Preview or output window** : Space to see the output.	
This section is for Teacher's Information. Explain this only if student asks about it	Here the file Explorer has a lot of files that we are going to use, I have added them earlier. ● The **png** files that you see here are the image files that we will be using for our game like the background, the superman and the villian. ● The **p5Play.js** is a supporting library file. Because of this file, we are able to use graphics and sound functionality for our program. ● **index.html** is the main file where all files are connected and it makes the whole code work. ● **style.css** is a file, where we can code to add in more style and design to our program, we don't need it today, we will study about them later. ● The main code of the game is written in the file **sketch.js**	
	Let us start from the beginning to code the game. Here, you can see three main functions, let's learn	

	about them one by one. `function preload(){` `}` `function setup() {` `  createCanvas(500, 500);` `}` `function draw() {` `  drawSprites();` `}` • **preload**() is for loading any image or sound files to be used in the program . • **setup**() is for creating the objects or characters of our game. • **draw**() is for adding properties to the objects or characters of our game. Inside the **setup()** function you can see another function *createCanvas(500, 500);* • ***createCanvas()*** creates a canvas on the output window. • Size of the canvas can be changed by editing the width and height value. • *This is the syntax of the function:* ***createCanvas(width,height)*** • 500, 500 are the width and height of the canvas. You will be able to see the canvas clearly once we put an image for the canvas background.	Preloading it saves it in the memory of the browser, which helps the game run faster.
	Inside the **draw()** function you can see another function drawSprites(); • drawSprites() is a function to draw the sprites on the output or preview screen. Okay now what is this sprite, you must be thinking, right? Simply speaking, you can assume sprites to be objects or components of a game. Like in this game, the background, superman and villain are all components of the game, so we need to create sprites for all of them,	Sprite is a 2D bitmap graphic used in animations & video games.
	The elements and characters of our game are as	Ask if the student

	follows: 1. **Background-** If you have watched the superman movie, superman was from a planet **Krypton.** So, we will create the space background you saw. 2. **Superman-** Main Player 3. **Villain-** because what will a superhero do without a villain.	has seen any superman movies and let the student guess the home planet of superman.
	For every object we need to do 3 things:- 1. **Load an image for the object** - This will be done in **preload()** 2. **Create a sprite for the object** - This will be done in **setup()** 3. **Add the image on the sprite object** - This will be done in **draw()**	Point out to the portion of the program where these actions will be done.
	To load an image, we can use a function **loadImage()** inside the **preload()** function. We will create a variable to store the loaded image. *Syntax:-* *image_var_name = loadImage("file_name");* The image we have for background is the *space.png* in the file explorer. *bgImage = loadImage("space.png");* We have loaded the space image in a variable named **bgImage**.	We are doing this in preload so that the game loads faster. Recall the concept of naming variables in python. Show the file in the file explorer section.
	Now in the **setup()** function, **To create a sprite**, we use a function **createSprite()**. Every sprite that we create needs to be given a name, again we will create a variable to store the sprite. *Syntax:* *name_of_sprite = createSprite(x,y,width,height);* "x" shows the horizontal position of the sprite on canvas. "y" shows the vertical position of the sprite on	Use this image to explain coordinate system of canvas: Canvas Coordinates

	canvas. Width and height as you would have guessed, are the width and height of the sprite. So we give different values in this function to decide the position and size of the sprite.	
	Let us create a background sprite object. *bg = createSprite(250,250,50,50);* Let's increase the size of the bg sprite to cover the whole screen. *bg = createSprite(250,250,500,500);*	The upper-left corner of the canvas has the coordinates (0,0) *Note: Click on the preview window for playing the game.*
	Now we don't want this grey background here, right? We have loaded a background image that we want on this sprite.	Refer to the preload function, and point out that we are going to use this image.
	To add an image on the sprite object, we use the function **addImage.** *Syntax:* *sprite_name.addImage(image_var_name);* Here, *image_var_name* refers to the variable in which the image was loaded.	
	Let's add the image on the background sprite named **bg** *bg.addImage(bgImage);* Here, *bgImage* refers to the variable in which the **space.png** was loaded.	

<table>
<tr><td colspan="3">

```
function preload(){
 bgImage = loadImage("space.png");
}
function setup() {
  createCanvas(500, 500);
  bg = createSprite(250,250,500,500);

}
function draw() {
   bg.addImage(bgImage);
  drawSprites();
}
```


</td></tr>
<tr><td></td><td>Looks beautiful right!
But it is not moving.

You must have played games. Have you noticed that most of the games have endless space or a moving background.

Have you played any runner game?

If you play any runner game like mario, temple run, subway surfer, you never run out of space. You always have an endless road to run on.

That’s an illusion. Shall we create an endless space for our Superman to fly ?</td><td>Let the student give an example of any runner game he/she has played.

If the student doesn’t know the Runner game, give examples.</td></tr>
<tr><td></td><td>So for the illusion we need to make the background move.

I.e. making our background sprite bg move.

We have to give speed to an object to make it move.

But what if we want it to be moving in a particular direction?

Speed with direction is called velocity.

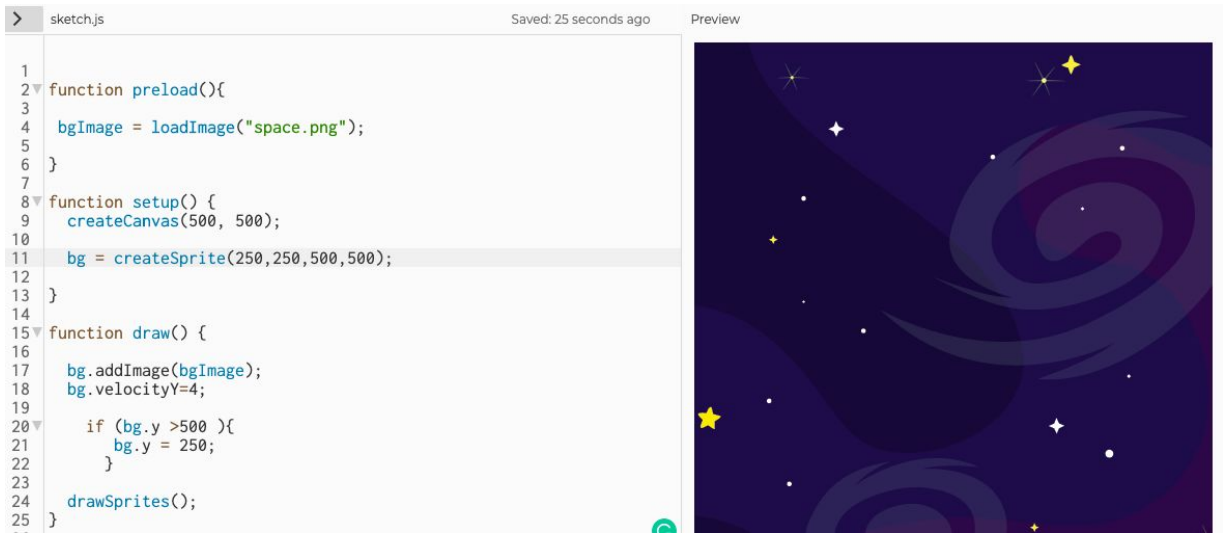
We will give velocity to the sprite bg in the function draw() using the command velocityY

Can you guess what velocityY is?</td><td>Run and show the output to the student.

Make the student</td></tr>
</table>

	Yeah you guessed it right, it's the speed in **Y** directions i.e. vertical direction. Let's write the code then, *bg.velocityY=4;* Now, see we have put in 4 as the velocity Y for the sprite **bg**. This will make the background move down. Let's run the code and see what happens. Oh no! This is not what we wanted right? We want the background to move downwards and then come back again to form an endless space. Let us bring back the background once it goes out. We want to continuously bring it back. So, we see the background going out of the canvas, but how does the computer understand where it is? Do you remember the position of the sprite is stored in the computer memory? Also we know the size of the canvas to be 500. If the position of the sprite is greater than 500, that means it has gone out of the canvas, right? So we will put a condition that whenever the sprite moves out of the canvas, the sprite is brought back into it. *if (bg.y >500){* *bg.y = 250;* *}* *bg.y* is the y position of the sprite. As the value of y position becomes greater than 500 we change it to half of it, i.e. 250, so it comes back to the centre of the canvas. Now, let's run the program and see what	recall that x refers to the horizontal line and y refers to the vertical line. Run the code and show how the background moves down and disappears. Point to the create canvas command. *Note:- x, y are the predefined properties of a sprite object. "x" is the x position and "y" is the y position of the bg sprite object.* Run the code and

	happens. We created an infinite background using just a few lines of code. We can do a lot more in this powerful programming language. Now it's your turn. Can you create sprites for the player and villain?	show the endless moving space.



Note :- Background is scrolling top to down

• **Stop your screen sharing** • **Help the student to share their screen** • **Take the student on student activity platform**

Student Led Activity II (20 mins)	Now it's your turn to create the player and the villain. Can you tell me the 3 steps you will have to do? Yes, well done. First we will load images for the character. Then we will create the sprite for that character And finally we will add the image we loaded in the first step to the sprite.	Share the link in the chat box if required. Student Activity 1: Player and Villain Prompt the answer if required.
	Student loads the images for superman and villain.	Guide the student

	Student creates sprites for Superman and the villain. Student adds the images for superman and villain. sketch.js `function preload(){` `  playerImage =  loadImage("superman.png");` `  bgImage = loadImage("space.png");` `  villainImage = loadImage("villain.png");` `}` `function setup() {` `  createCanvas(500, 500);` `  bg = createSprite(200,200,400,400);` `  player = createSprite(50,300,20,20);` `  villain =createSprite(460,50,20,20);` `}` `function draw() {` `  bg.addImage(bgImage);` `  bg.velocityY=4;` `  player.addImage(playerImage);` `  villain.addImage(villainImage);` `  if (bg.y >500 ){` `  bg.y = 250;` `  }` `  drawSprites();` `}`	in creating the sprites with the size and x,y coordinates. Mention that the upper left portion of the canvas is 0,0 
	• All the activities are mandatory till here. • If the time permits then proceed further with one of the two options given below. • Option 1 gives movement to sprite using mouse control • Option 2 gives movement to sprite using keyboard control • The section after option 2 is compulsory to destroy the villain.	
Option1: Mouse Control	Let us move the player with our mouse pointer. To make the player move with the mouse, we will assign the mouse's x and y position value to our player's x and y position. As we are not creating any new sprite and we are not loading any new image, where should we write the code ? Yes, we should write the assignment code in the draw function. Draw function is called continuously leading to continuous assignment of mouse position to player's x and y position. *player.x=mouseX;*	Prompt the student to answer. Share the code via

	player.y=mouseY; ***mouseX*** *and* ***mouseY*** *are the commands to get the x and y position of the mouse.* `function draw() {` `bg.addImage(bgImage);` `bg.velocityY=4;` `player.x=mouseX;` `player.y=mouseY;` `player.addImage(playerImage);` `villain.addImage(villainImage);` `if (bg.y >500 ){` `bg.y = 250;` `}` `drawSprites();` `}`	chat or dictate it. You can make the student recall the **velocityY** command used earlier.
Option 2:- Keyboard Control	Let us move the player with the arrow keys. We need to tell our program that If the left arrow key is pressed, move the player's position to the left. If the right arrow key is pressed, move the player's position to the right. if the up arrow key is pressed, move the player up. If the down arrow key is pressed, move the player down. As the syntax is a little tricky for the first class, I will help you. *Pass the below code in the chat window* *if(keyDown("left")) {* *player.x = player.x-4;* *}* As we are not creating any new sprite and we are	Guide the student to the chat window. Ask the student to copy the code and paste it in the correct portion of the code.

	not loading any new image, where should we write the code ? Yes, we should write the code in the draw function. Copy the code from chat into the workspace `if(keyDown("left") ) {` `player.x = player.x-4;` `}` This code first checks if the left arrow key of the keyboard is pressed using *keyDown("left")* When the condition is true, then the line inside is executed. *player.x=player.x-4* will decrease the x position value by 4 pixels every time the left arrow key is pressed so it will move leftward. Now, we need to duplicate the same code three times for all the remaining arrow keys. In place of **left** we need to put **right** and instead of **x-4** we will put **x+4** `if(keyDown("right") ) {` `player.x = player.x+4;` `}` Similarly, we need to edit the remaining two if blocks for **up** and **down.** But for up and down direction, what we need to keep in mind is that we will use **player.y** instead of **player.x** Can you tell me why?? Because **left** and **right** are the **horizontal** directions and **x** denotes **horizontal** direction. But **up** and **down** are the **vertical** directions and **y** denotes **vertical** direction. Again keep in mind the canvas direction, to move up we need **y-4**	Remind the student about the x and y value on the canvas.

	And to move down we need **y+4** *if(keyDown("up")) {* *player.y = player.y-4;* *}* *if(keyDown("down")) {* *player.y = player.y+4;* *}* `function draw() {` `bg.addImage(bgImage);` `bg.velocityY=4;` `if(keyDown("left") ) {` `player.x = player.x-4;` `}` `if(keyDown("right") ) {` `player.x = player.x+4;` `}` `if(keyDown("up") ) {` `player.y = player.y-4;` `}` `if(keyDown("down") ) {` `player.y = player.y+4;` `}` `player.addImage(playerImage);` `villain.addImage(villainImage);` `if (bg.y >500 ){` `bg.y = 250;` `}` `drawSprites();` `}`	
Compulsory section	Let's run the code now. Now the superman can easily catch the villain. But then what? Nothing happens. We want the villain to disappear,when the superman catches him.	Ask the student to click on the run button and move the superman to reach the villain.
	For that we have a very simple line of code. Using **if** and **.isTouching()** functions we can easily check if two sprites are touching each other or not and by using **.destroy()** function we can make a sprite disappear from canvas. See this code	Send the syntax in the chat box to and

	if(sprite1.isTouching(sprite2)){ *sprite2.destroy();* *}* Can you tell me what will be the value of sprite 1 and 2 here? Great! Sprite 1 is the **player** and sprite 2 is the **villain.** You can copy the code from chat and replace sprite 1 and sprite 2 in the code with player and villain. *if(player.isTouching(villain)){* *villain.destroy();* *}*	explain the structure. Refer to the sprite creation code, where the variable player and villain was used for the two characters of the game.
	Run and see the results. Did you enjoy creating your own game?	Make the student run the program and play for sometime.
In case the student mentions that the game which you showed at first was a little different, in that case first praise the student for attention to details. *Ask why do they think that was different?* *Mention that the game requires one more concept, i.e. gravity which is making the superman fall down when not controlled by the player.* *Few more lines of code will add that function into the game too, but we will learn later about that.*		
	Great! You did a great job in this class. You completed all your challenges and I am surprised to see your patience. So you deserve applause! One applause for completing all the activities 👏 One applause for your patience 👏	

Only if the student or parent enquires about the course.

Our course follows a project-based approach where the students learn the concepts of computer programming while developing games, websites and applications. They develop their logical and algorithmic thinking skills by solving complex problems which a game/app development can offer.

Our course is divided into 3 modules.
The first module consists of 8 classes. In this module your kid will be introduced to the core concepts of computer programming using Python.
The second module consists of 40 classes. In this module your kid will learn how to create Websites and Games. Also, they will be introduced to Application development where the students will apply their creative skills to develop advanced mobile and web-applications.
The third module consists of 96 classes, in which the students will reach the peak of their creativity and develop industry-standard native applications which can be publishable in playstore. They will also be introduced to Artificial Intelligence.
To provide more details about the course and to solve your questions, our counsellor will contact you very soon. I would really love to have your kid as my student as I can see a great potential in him/her. Thank you for your time sir/ma'am.

BID GOOD BYE & END CLASS

Resources:

Activity	Name	Links
Teacher Activity 1	Hogwarts	https://repl.it/@TopprCodr/Hogwarts#main.py
Teacher Activity 2	Sample of final SuperMan game- To be shared in chat box	https://editor.p5js.org/iamsunitak/present/dg0vnMNMw
Teacher Activity 3	SuperMan level 0	https://editor.p5js.org/TopprCodr/sketches/UGGahVFLa
Teacher Activity 4	Canvas Coordinates	https://drive.google.com/file/d/1uL6IQ138OYCa2zje-RIGwqFM68n2ubjB/view?usp=sharing

Student Activity 1	Player and Villain	https://editor.p5js.org/TopprCodr/sketches/4pwEcoWbc
Teacher Reference 1	SuperMan Game: Background and Players	https://editor.p5js.org/TopprCodr/sketches/BL3jQ57Z4
Teacher Reference 2	Reference: Mouse Control.	https://editor.p5js.org/TopprCodr/sketches/cfoW267Bm
Teacher Reference 3	Reference: Keyboard Control.	https://editor.p5js.org/TopprCodr/sketches/S3zXVeeps
Teacher Reference 4	Reference: Final Game	https://editor.p5js.org/TopprCodr/sketches/9qv8lfdMg