Topic	**CONDITIONAL PROGRAMMING**	
Class Description	**Students use conditional programming to add control to the ball's movements. Student builds a little game using the ball's movements and adds some challenge to it.**	
Class	**PRO-C3**	
Class time	**45 mins**	
Goal	• Use conditional programming to add control to the ball's movements. • Make a challenging game using the ball's movements.	

Resources Required	• Teacher Resources
	  ○ Code.org login
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen
	• Student Resources
	  ○ Code.org login
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen	

Class structure	**Warm Up - Slide show option** **Teacher-Led Activity** **Student-Led Activity** **Wrap Up - Slide show option**	**15 Mins** **8 Mins** **30 Mins** **5 Mins**

WARM UP SESSION - 15mins

Teacher starts slideshow  **from slides 1 to 12**

Refer to speaker notes and follow the instructions on each slide.

Activity details	Solution/Guidelines
Hi, how are you? How was your day? Do you remember what we did in the previous class? **Run the presentation from slide 1 to slide 2.** **Following are the warm up session deliverables:** • Connecting students to the previous class. • Explaining conditional programming through real life connect. • Definition of control and conditional programming.	**ESR:** I am good. Yes. We made the ball move in the last class. Click on the slide show tab and present the slides.
QnA Session	
Question	**Answer**
Consider two balls ball1 and ball2 in the Pong game. Which of the following is the correct code to make the balls bounce off each other? A. Aball1.bounce(ball2); B. ball1.bounceOff(ball2); C. ball1.jumpoff(ball2); D. ball2.collide(ball1);	**B**
Continue the warm up session	
Activity details	**Solution/Guidelines**

<table>
<tr><td colspan="2">Run the presentation from slide 2 to slide 13 to set the problem statement.

Following are the warm up session deliverables:
• Introduce students to the coding environment - Workspace, blocks and output.
• Steps to write and run the code.</td><td>Narrate the story by using hand gestures and voice modulation methods to bring in more interest in students.</td></tr>
<tr><td colspan="3">Teacher ends slideshow  </td></tr>
<tr><td colspan="3">TEACHER-LED ACTIVITY - 8mins</td></tr>
<tr><td colspan="3">Teacher Initiates Screen Share</td></tr>
<tr><td colspan="3"><u>CHALLENGE</u>
• Create a ball whose movements can be controlled by the arrow keys.</td></tr>
<tr><td>Step 2:
Teacher-led Activity
(10 min)</td><td>Review last class.

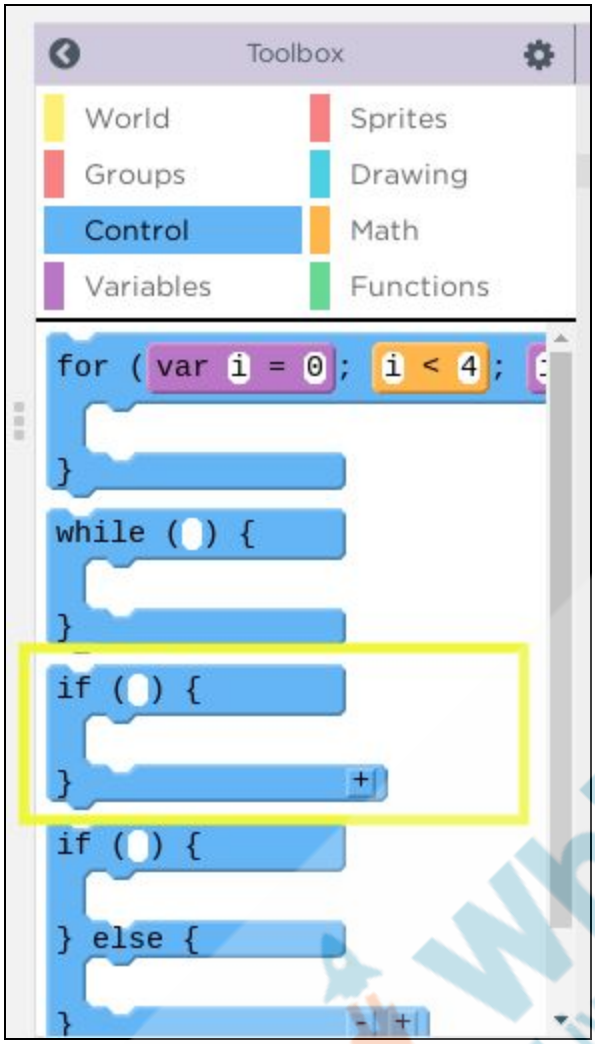
Let's open the playground project and play around a little more.

Teacher opens the Playground project (Teacher Activity 1) from the activity tab, presses REMIX (top left) and runs the code.</td><td></td></tr>
<tr><td colspan="3">Code:</td></tr>
</table>

```
var ball = createSprite(200,200,10,10);

//ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");


  createEdgeSprites();
  ball.bounceOff(edges);
  drawSprites();
}

```

	Motivation to add control to the ball movements. It is quite fun watching the ball bounce against the walls. But right now, it is going in any direction. We don't have any control over the ball. Wouldn't it be wonderful if we could control the ball's direction by pressing the arrow keys? Games also require control over the character. Do you remember any game which requires you to control the character in the game?	**ESR:** [varied]
	Let's add control to our ball in this playground. How do you think you would want to map the arrow keys to ball movements?	**ESR:** - left key: ball goes left - right key: ball goes right - up key: ball goes up - down key: ball goes down
	So we want to give instructions to the computer in the following way. IF the left key is pressed, THEN move	

	the ball left. IF the right key is pressed, THEN move the ball right. IF the up key is pressed, THEN move the ball up. IF the down key is pressed, THEN move the ball down. We just talked about giving such instructions to the computer. Do you remember what it's called?	**ESR:** CONDITIONAL PROGRAMMING
	Code to add "Up" control to the ball. Let's learn how to do conditional programming and add control to the ball. There is an instruction in our toolbox which will help us do that. *Teacher locates if() block inside Control in the toolbox.*	*The student listens and watches.*



	Teacher drags and drops the if() block inside draw(). We use this instruction to tell the computer - if this condition happens, do these things. We put the condition inside "( )" and statements/instructions we want the computer to run if that condition is met inside "{ }"	*The student listens and watches.*

Code:

```
var ball = createSprite(200,200,10,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if () {

  }


  createEdgeSprites();
  ball.bounceOff(edges);
  drawSprites();
}
```

```
var ball = createSprite(200,200,10,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if (condition) {
    //do this
    //do this
  }


  createEdgeSprites();
  ball.bounceOff(edges);
  drawSprites();
}
```

	What is the condition for the ball to move up?	**ESR:** Pressing the up arrow key

	Let's write it down inside the if () as a condition. What do we want the computer to do if the condition happens? Let's put this inside the curly brackets (**{ }**) as instructions/statements which will be run if the condition is satisfied.	**ESR:** Move the ball up.

Code:

```
1  var ball = createSprite(200,200,10,10);
2
3  ball.velocityY = 2;
4  ball.velocityX = 2;
5
6  function draw() {
7    background("white");
8
9    if (up arrow key is pressed) {
10     //do this
11     move the ball up
12   }
13
14
15   createEdgeSprites();
16   ball.bounceOff(edges);
17   drawSprites();
18 }
```

	But we have written the condition and the statements in English. Do you think the computer would understand them? The computer would need more precise language for the condition and the instructions.	**ESR:** No

	For the condition, we use the **keyDown()** instruction from the toolbox. *Teacher navigates to the keyDown() instruction inside World in the toolbox.* What instruction can we use to move the ball up? How did we move the ball in the last game?" Let's modify the condition and the statements inside the if-block so that it is in a language the computer can understand. *Teacher makes the change.*	**ESR:** ball.velocityX = 0; ball.velocityY = -2; *The student observes and learns.*
Code:		

```
var ball = createSprite(200,200,10,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if (keyDown("UP_ARROW")) {
    //do this
    ball.velocityX = 0;
    ball.velocityY = -2;
  }


  createEdgeSprites();
  ball.bounceOff(edges);
  drawSprites();
}

```

	Let's test if our code does what we want it to do. *Teacher runs the code.* What should we do to test if the code is correct? What do you think will happen if we press the up arrow key? Let's try. *Teacher presses Up Arrow to make the ball go up.* It's working!	**ESR:** Press the up arrow key. **ESR:** The ball should move up.

	Do you want to write code to add other controls for the ball? Making it go left, right and down? We will also convert this into a small interesting game. Share your screen so that I can guide you while you are working on it.	**ESR:** Yes
Teacher Stops Screen Share		
Teacher starts slideshow :Slide 14-16 (Only 1 slide for this Activity)		
Run the presentation for slide 14-16 to set the student activity context.		
	Now it's your turn. Please share your screen with me.	
Teacher ends slideshow		
• **Ask Student to press ESC key to come back to panel** • **Guide Student to start Screen Share** • **Teacher gets into Fullscreen**		
ACTIVITY • **Create a small game where the ball bounces off a target off the screen to display a victory message.**		
Step 3: Student-Led Activity (20 min)	*Guide the student to add complete control of the ball's movements - up, down, left, right.* *Guide the student to write another IF block to make the ball go down when the down key is pressed.* *Observe the student's code for typos.*	*Student opens Student Activity 1 and writes code to make the ball go down when the down key is pressed.* *Student runs the code.*

Code:

```
var ball = createSprite(200,200,10,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if (keyDown("UP_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = -2;
  }

  if (keyDown("DOWN_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = 2;
  }


  createEdgeSprites();
  ball.bounceOff(edges);
  drawSprites();
}

```

	Guide the student to write another IF block to make the ball go left when the left key is pressed. *Observe the student's code for typos.*	*The student writes code to make the ball go left when the left key is pressed.*

Code:

```
var ball = createSprite(200,200,10,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if (keyDown("UP_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = -2;
  }

  if (keyDown("DOWN_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = 2;
  }

  if (keyDown("LEFT_ARROW")) {
    ball.velocityX = -2;
    ball.velocityY = 0;
  }


  createEdgeSprites();
  ball.bounceOff(edges);
  drawSprites();
}

```

	Guide the student to write another IF block to make the ball go right when the right key is pressed. *Observe the student's code for typos.*	*The student writes code to make the ball go left when the left key is pressed.*

Code:

```
var ball = createSprite(200,200,10,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if (keyDown("UP_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = -2;
  }

  if (keyDown("DOWN_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = 2;
  }

  if (keyDown("LEFT_ARROW")) {
    ball.velocityX = -2;
    ball.velocityY = 0;
  }

  if (keyDown("RIGHT_ARROW")) {
    ball.velocityX = 2;
    ball.velocityY = 0;
  }


  createEdgeSprites();
  ball.bounceOff(edges);
```

	Create a small maze. Let's turn this into a small game. Let's create a small maze through which the ball has to be navigated by the player. IF the ball hits any of the walls of the maze, the ball should go back to its starting position. Do you think you can do this?	**ESR**: Yes!

	Guide the student to create a maze of her/his own style.	*The student writes code to create a maze by creating rectangular wall sprites.* *He/She runs the code.*

```
var ball = createSprite(200,200,10,10);
ball.velocityX = 2;
ball.velocityY = 3;

ball.shapeColor = 'blue';

var wall1 = createSprite(10,50,20,100);
wall1.shapeColor = 'red';

var wall2 = createSprite(50,50,20,100);
wall2.shapeColor = 'green';


var wall3 = createSprite(50,130,100,20);
wall3.shapeColor = 'brown';

function draw() {
    background("white");

    if(keyDown("up")){
      ball.velocityX = 0;
      ball.velocityY = -2;
    }

    if(keyDown("down")){
      ball.velocityX = 0;
      ball.velocityY = 2;
    }
```

	Guide the student to use sprite.isTouching property to check if the ball is touching one of the walls. If the ball is touching one of the walls, change the position of the ball to its starting point. Repeat this for all the walls in the maze.	*The student writes code to check if the ball touches any of the walls.*

```
1  var ball = createSprite(200,200,10,10);
2  ball.velocityX = 2;
3  ball.velocityY = 3;
4
5  ball.shapeColor = 'blue';
6
7  var wall1 = createSprite(10,50,20,100);
8  wall1.shapeColor = 'red';
9
10 var wall2 = createSprite(50,50,20,100);
11 wall2.shapeColor = 'green';
12
13 var wall3 = createSprite(50,130,100,20);
14 wall3.shapeColor = 'brown';
15
16
17 function draw() {
18     background("white");
19
20     if(ball.isTouching(wall1)){
21       ball.x = 200;
22       ball.y = 200;
23     }
24
25
26     if(keyDown("up")){
27       ball.velocityX = 0;
28       ball.velocityY = -2;
29     }
```

```
11  wall2.shapeColor = 'green';
12
13  var wall3 = createSprite(50,130,100,20);
14  wall3.shapeColor = 'brown';
15
16
17  function draw() {
18      background("white");
19
20      if(ball.isTouching(wall1)){
21        ball.x = 200;
22        ball.y = 200;
23      }
24
25       if(ball.isTouching(wall2)){
26        ball.x = 200;
27        ball.y = 200;
28      }
29
30       if(ball.isTouching(wall3)){
31        ball.x = 200;
32        ball.y = 200;
33      }
34
35
36
37      if(keyDown("up")){
38        ball.velocityX = 0;
39        ball.velocityY = -2;
```

If you check, we are essentially giving the same instruction to the computer - move the ball back to the starting position - for different conditions.

We don't have to repeat the instructions. We can combine all the conditions using OR.

We can say if the ball is touching wall1 OR ball is touching wall2 OR ball is touching wall3, follow these instructions.

	In computer language OR is written as || (two PIPE symbols) *Guide the student to use || operator to combine all the conditions.*	*The student writes code to join all the conditions using the OR symbol.*

```
11  wall2.shapeColor = 'green';
12
13  var wall3 = createSprite(50,130,100,20);
14  wall3.shapeColor = 'brown';
15
16
17  function draw() {
18      background("white");
19
20      if(ball.isTouching(wall1) || ball.isTouching(wall2) || ball.isTouching(wall3)){
21        ball.x = 200;
22        ball.y = 200;
23      }
24
25
26      if(keyDown("up")){
27        ball.velocityX = 0;
28        ball.velocityY = -2;
29      }
30
31      if(keyDown("down")){
32        ball.velocityX = 0;
33        ball.velocityY = 2;
34      }
35
36      if(keyDown("left")){
37        ball.velocityX = -2;
38        ball.velocityY = 0;
39      }
```

	Scramble the keys to make it challenging. This game is too simple, isn't it? What is a game without a little challenge? Do you have any ideas on how to make the game more challenging?	**ESR:** Varied

	Let's try to make one simple change. Let's scramble the keys so that - "left" goes "up" "up" goes "right" "right" goes "down" and "down" goes "left". Let's see if the game becomes more challenging then. *Guide the student towards scrambling the keys and the direction in which they take the ball.*	*The student writes code to scramble the keys and the direction in which they take the ball.* *He/She runs the code and plays the game.*

```
var ball = createSprite(200,200,10,10);
var target = createSprite(330,10,80,10);

ball.velocityY = 2;
ball.velocityX = 2;

function draw() {
  background("white");

  if (keyDown("LEFT_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = -2;
  }

  if (keyDown("RIGHT_ARROW")) {
    ball.velocityX = 0;
    ball.velocityY = 2;
  }

  if (keyDown("DOWN_ARROW")) {
    ball.velocityX = -2;
    ball.velocityY = 0;
  }

  if (keyDown("UP_ARROW")) {
    ball.velocityX = 2;
    ball.velocityY = 0;
  }

  text("Push me out of screen", 270, 30);

```

	Was it challenging? You can make this your own game and make it more challenging. Let's wrap up the class for now.	

Teacher Guides Student to Stop Screen Share

Quiz time - Click on in-class quiz

Question	Answer

Consider the following code for adding controls to a ball movement. What is the purpose of writing "ball.velocityX = 0"?
 	
var ball = createSprite(200,200,10,10);
ball.velocityX = 2;
ball.velocityY = 3;
 	
function draw() {
background("white");
 	
If(keyDown(UP_ARROW) {
ball.velocityX = 0;
ball.velocityY = -3;
}
 	
drawSprites();
}
 	
A. To make sure the ball moves upwards when the up arrow is pressed and does not move in a diagonal direction due to velocityX value
B. To make sure the ball moves upwards when the up arrow is pressed and moves in a diagonal direction due to velocityX value
	A

C. To make sure that ball is moving upwards when the up arrow is pressed D. To make sure that ball has some movement when the up arrow is pressed	
Which operator can be used to combine multiple conditions in an if statement? A. AND (&&) operator B. OR (II) operator C. Both 1 and 2 D. It is not possible to combine multiple conditions in an if statement	**C.**
Sam wants to move an object while playing a game. Which function should be used for writing the corresponding code? A. Inside the conditions() function B. Inside setup() function C. Inside draw() function D. All of the above	**C.**
End the quiz panel	
WRAP UP SESSION - 5 Mins	
Teacher starts slideshow  **Slide 18-24**	
Activity details	**Solution/Guidelines**
Run the presentation from slide 18 to slide 24 **Following are the warm up session deliverables:**	

<table>
<tr><td colspan="2">

- **Explain the facts and trivias**
- **Next class challenge**
- **Project for the day**
- **Additional Activity**

</td><td>Guide the student to develop the project and share with us.</td></tr>
<tr><td colspan="3">Teacher ends slideshow  </td></tr>
<tr><td>Project Overview</td><td>Note: This is a tiered project with multiple tasks. All students must do the main task. The main task is very similar to the projects that are already live. Each tiered project has two or more additional tasks which are optional.

SOPHIA IN MAZE LAND

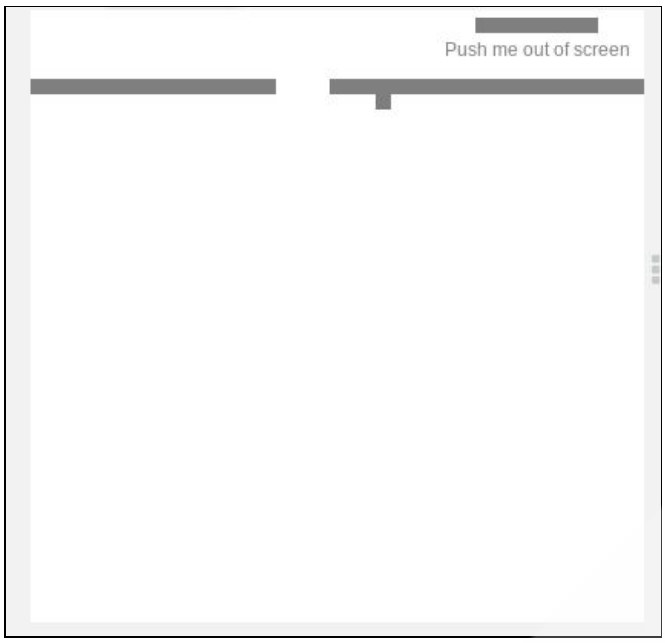
Goal of the Project:
Today, you learned to use Conditional Programming (if statements) to add control to the game elements.

In this project, you will have to practice and apply what you have learnt in the class and create a game of Maze for Sophia to play with.

Story:
Sophia loves to play maze games, where her mother helps her create actual mazes using cardboard pieces at home. Sophia has just embarked on a coding learning journey and she is eager to try her hand at creating a virtual game of maze.</td><td>Students engage with the teacher over the project.</td></tr>
</table>

	Help Sophia build a such a game with a path that leads to a golden cup. I am very excited to see your project solution and I know you both will do really well. Bye Bye!	
Teacher Clicks 		
Additional Activities	*Encourage the student to make the game more challenging by creating a more complicated maze with obstacles (fixed or moving)*	*The student writes code to create a more complicated maze.*

```
1  var ball = createSprite(200,200,10,10);
2  var target = createSprite(330,10,80,10);
3  var obstacle1 = createSprite(10,50,300,10);
4  var obstacle2 = createSprite(320,50,250,10);
5
6  ball.velocityY = 2;
7  ball.velocityX = 2;
8
9  function draw() {
10   background("white");
11
12   ball.collide(obstacle1);
13   ball.collide(obstacle2);
14
15   if (keyDown("LEFT_ARROW")) {
16     ball.velocityX = 0;
17     ball.velocityY = -2;
18   }
19
20   if (keyDown("RIGHT_ARROW")) {
21     ball.velocityX = 0;
22     ball.velocityY = 2;
23   }
24
25   if (keyDown("DOWN_ARROW")) {
26     ball.velocityX = -2;
27     ball.velocityY = 0;
28   }
29
```



	Encourage the student to write the same code in the p5 editor. *Guide the student to write the same game code in p5 editor*	*Student writes the same code in the p5 editor.*

<Check final Teacher Reference code for solution>

Activity	Activity Name	Links
Teacher Activity 0	Complete code for pong game	https://studio.code.org/projects/gamelab/l8gg2ID9B0WlHEEJqGS0jzX3nZ6XMF7Nz6oNYaaqcy4
Teacher Activity 1	Playground	https://studio.code.org/projects/gamelab/icbPvTKK7eCRwFAQCM3wTVsrk3eJrFM88PKHWepGcJo/edit
Teacher Activity 2	Playground 2 (Teacher Reference)	https://studio.code.org/projects/gamelab/FgXkmmHIJC2cadr4d1MSGg1cAcUPV_JHsZlE1yWg9EM/edit
Student Activity 1	Playground	https://studio.code.org/projects/gamelab/HIbZy1Iby5j54c64Tb2JMq1oTjsBPyOjG6hqltYDLDU/edit
Teacher Reference	Additional Activity Solution	https://editor.p5js.org/whitehatjr/sketches/8cCIhXB_
Teacher Reference visual aid link	Visual aid link	https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/C3+without+cues.html
Teacher Reference In-class quiz	In-class quiz	https://curriculum.whitehatjr.com/Visual+Project+Asset/PRO_VD/PRO-C3_Jayshree.docx.pdf