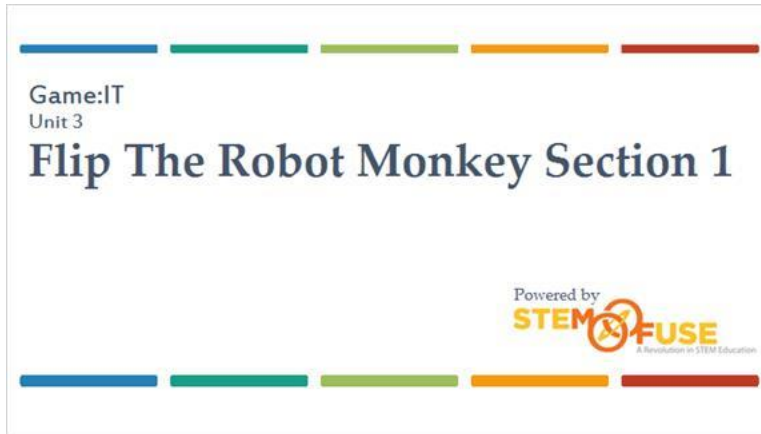


1 - Title Slide: Flip The Robot Monkey Section 1



Game:IT
Unit 3

Flip The Robot Monkey Section 1

Powered by
STEM & FUSE
A Revolution in STEM Education

The slide features a decorative border at the top and bottom consisting of five colored horizontal bars: blue, green, light green, orange, and red. The text is centered and includes the game title, unit information, and the STEM & FUSE logo.

2 - Introduction

For the first full Construct 2 game, you're going to create a platform game called Flip the Robot Monkey.

Platform games involve a character moving across different levels of platforms, avoiding obstacles to reach an end goal. A couple of examples of this type of game are Super Mario Bros. and Ratchet & Clank.

For your game, you'll have a main character named Flip that goes through a castle avoiding knights and archers while he frees his monkey friends.

3 - Project Start

You will use the Flip Intro as your starting point and build onto that game, so open that game now.

Make sure you have completed the Flip Intro.

Section 1 is dependent on the fact that you have completed those instructions.

4 - Section Objective

Section Objectives

During this presentation you'll cover the following objectives

- Setting Collision Points
- Create a Level
- Adding a Super Jumper

5 - Objective 1: Set Collision Points

Objective 1

Set Collision Points


Collision points and polygons allow you to indicate where the collisions will occur on a sprite. Setting these will be helpful to prevent objects from "catching" on platforms and will make animations appear smooth .

To learn more about Collision Points [Click Here](#)

6 - Collision Polygon

We will first set the collision polygon for Flip. In the Objects Bar, double click on the **Flip** object to open the image editor.

When the image editor appears, click the **Collision polygon** button. This button is the bottom button on the left toolbar.



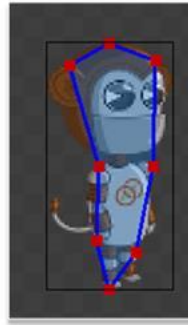
Objective 1 – Set Collision Points

7 - Flip Collision Polygon

A blue box with red points will now appear on the image. The edge of the blue box is where collisions with solid objects will occur. Anything outside of the box, like the tail, will ignore collisions.

By default, Construct 2 will guess the shape of the polygon but you can change that. For the Flip image, you'll change it to be a rectangular box.

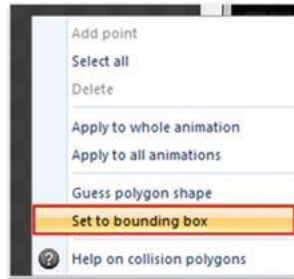
If you're having trouble seeing Flip, use the zoom in feature to make him larger.



Objective 1 – Set Collision Points

8 - Set To Bounding Box

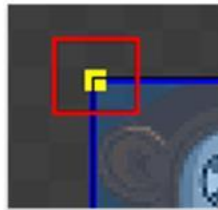
For the collision box for Flip, you'll only need to set four points. To easily do this, right click on the image and select **Set to bounding box**. This will set the collision polygon to the entire image.



Objective 1 – Set Collision Points

9 - Active Points

Click on the top left point to select that point. It will change it from red to yellow to indicate this is the point you've selected. You can click and drag these points to their desired position or input the X and Y value right below the top toolbar. For these points, you will input the X and Y values.



Objective 1 – Set Collision Points

10 - Flip Collision Diagram

For the top left point, set the X value to 17 and the Y keep at 0.

X 17 Y 0

Set the X and Y value for the other 3 points starting at the top right and moving clockwise to: 45, 0; 45, 100; 17, 100

Flip should look like this when done.



Objective 1 – Set Collision Points

11.1 - Apply To All Animations

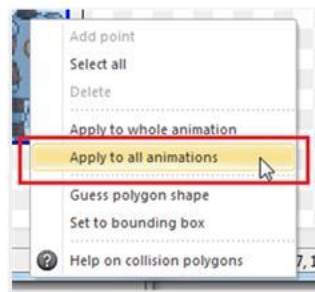
To keep things consistent, you'll want to set the same collision box for all of Flip's animations. Instead of doing this individually, you can use the one you already set and apply it to the rest of the animations.

To do that, with the Set collision polygon still selected, right click in the image editor and then select **Apply to all animations**. When the warning popup appears, click **Yes**. **Show Apply To All Animations**

Each animation has a different width, so the other collision polygons will not have the exact same X and Y values because points are calculated relatively to the width.

Objective 1 – Set Collision Points

11.2 - Apply To All Animations Image



Objective 1 – Set Collision Points

12 - Ground Object Settings

Next, double click on the Ground object, in the Layout window to open up the image editor. You will notice that the Ground object has a front face and a top portion. We want it to appear that Flip is walking on the top portion of the ground, so let's set the collision polygon as follows:

Top left = 0, 9

Top right = 572, 9

The bottom points do not need to be changed.

Close the image editor when this is done. Run the layout now and see that Flip appears to walk on top of the Ground object.

Objective 1 – Set Collision Points

13 - Ground Object Set To Bounding Box

Now double click on the Platform object to open the image editor. Click the Set collision polygon icon if necessary. Then right click on the image and select **Set to the bounding box** to get the four points.

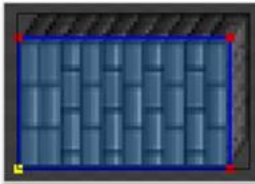
Set the points as follows:

Top Left = 0, 6

Top Right = 55, 6

Bottom Right = 55, 40

Bottom Left = 0, 40



If you run the layout now, you'll see that Flip appears to be walking on the top portion of the platform.

Objective 1 – Set Collision Points

14 - Objective 2: Create A Level

Objective 2

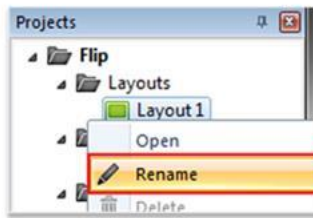
Create a Level

Before you start to build out your level, you'll want to rename your layout and event sheets. Maintaining a good naming convention makes programming easier and cleaner, especially when working in teams.

Don't forget to save your game, you wouldn't want to lose all your sweet stuff.

15 - Layout Rename

To rename the Layout, go to the Project Bar and in the Layouts folder right click on **Game** (or whatever you named the Layout in the Flip Intro) and click **Rename**.

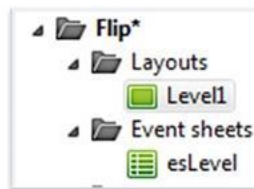


The name will now appear as a text box, and you can change the name. Rename it **Level1** (no spaces).

Objective 2 – Create a Level

16 - Event Sheet Renaming

Next you'll rename the event sheet by right-clicking on **Event sheet 1** in the Projects Bar and selecting **Rename**. Change its name to **esLevel**.



Most projects will use multiple event sheets, so it's helpful to rename them to correspond with the layout for which they contain events. Use "es" at the beginning to distinguish that it's an event sheet and use the layout name after to help tie it to the layout it corresponds with.

Objective 2 – Create a Level

17 - Layout Properties

Now that you have renamed the Layout and Event Sheet, you're going to change the size of the layout. Click on **Level1** in the Projects Bar and then go to the Properties Bar on the left. In the Layout, properties change the Layout Size to **2400, 720**.

Layout properties	
Name	Level1
Event sheet	esLevel
Active layer	Main
Unbounded s...	No
Layout Size	2400, 720
Margins	500, 500

Objective 2 – Create a Level

18 - Layout Size

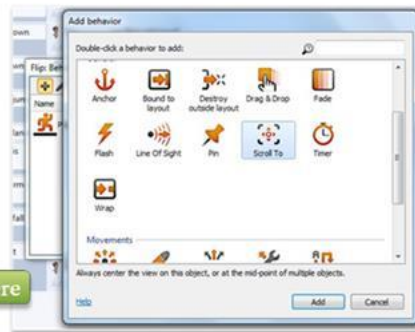
Increasing the Layout Size will allow you to create a bigger level with more obstacles and enemies. But, if you remember back to when you first started the project, you set the window size to 1280, 720. This means that only the first 1280 pixels of your level will be displayed. To see the rest of the level, you'll need to add a behavior to the Flip object so the window will follow him as he moves across the level.

Objective 2 – Create a Level

19 - Scroll To Behavior

Select your Flip object in the Objects Bar and in the Properties Bar click the Behaviors link. Click the Plus button to add a new behavior and select the Scroll To behavior.

To learn more about the Scroll To Behavior [Click Here](#)



Objective 2 – Create a Level

20 - Layout Grids

Next, you'll want to make sure that there's a surface across the entire layout. To do this, you'll add copies of your Ground object. To help place these objects, you're going to enable a grid on the layout. Grids are helpful for placing objects precisely on a layout.

Objective 2 – Create a Level

21 - View Menu

At the top of the window, click **View** to bring up the view tab.

The View options will appear, and you'll see a section for Grid options and Grid size. Here is where you'll set the width and height of the grid as well as its visibility and an option for an object to be only set on grid intersection points.



Make sure that the Layout tab is active. If you're on the Event Sheet, you will not be able to change the grid options section.

Objective 2 – Create a Level

22 - Snap To Grid

Check both boxes in the Grid options.

Since the Ground object is 572 x 60, you'll use these values for the width and height. Set the Grid width to 572 and the Grid height to 60.



After changing the Grid width or height, click anywhere in the View tab area to save your values. If your first click after changing is on the Layout, your values will not save.

Objective 2 – Create a Level

23 - Copying Ground Objects

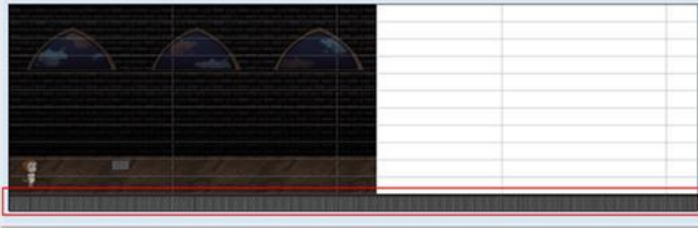
If you click on the layout, you'll see the grid now appears, and you can easily add more ground sections in their proper places.

To make more Ground objects, you'll create new Instances of the object. Instances will share the same events, images, and behaviors. To make a new instance of the Ground object hold the **Ctrl** key and then click and drag the **Ground** object in the layout. If you drag to the right, a new instance of the Ground object will appear in the next grid space. If you let go of the mouse, the instance will be created.

Objective 2 – Create a Level

24 - Creating Multiple Ground Instances

Create 3 more instances (4 total) of the Ground object, placing each one in the grid space to the right of the last.



Your last ground object will overflow off the layout. This is ok as just the part in the layout will appear when you run the layout.

Objective 2 – Create a Level

25 - Background Size

Once you have ground objects across the entire layout run the layout. You can see that the window will now follow Flip as he moves across the level.

You'll also see that the background stops. To fix this, you'll have to increase the size of the Background object.

Select the **Background** object from the Object Bar. In the Properties, Bar set its Size to 2400, 720. The Background will now extend the entire layout.

Common	
Layer	Background
Angle	0
Opacity	100
Position	0, 0
Size	2400, 720

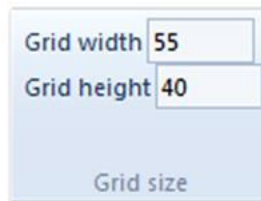
Objective 2 – Create a Level

26 - Platform Grid Settings

Next you'll want to add more instances of the Platform object to build out your level. To help with this, you can set the grid size to a different size. In the top toolbar set the Grid width to 55 and the Grid height to 40.

The width is set to 55 pixels instead of the image width of 60 to ensure that there are no gaps in the collision polygons.

Remember the collision polygon for the Platform object has a width of 55 pixels.



Objective 2 – Create a Level

27.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

◇ Using instances of the Platform object to build out the level. Create some groups using 2 or 3 instances of the Platform object.

[Click Here To See A LVL Example](#)





When you create a level, you may test it and find that you cannot reach platforms. You'll address this issue in the next objective.



27.2 - Created Lvl

On Your Own

Complete all the tasks before continuing to the next slide




28 - Edge Sprite

The last thing you'll want to do for the level is prevent Flip from going off the left and right of the layout. To do this, you're going to add a solid object to left and right edges of the level.

On your layout add a new object. For the object type select **Sprite** and name the object **Edge**. When the cross-hairs appear, click on the left edge of the layout, to bring up the image editor.

Objective 2 – Create a Level



29 - Edge Resizing

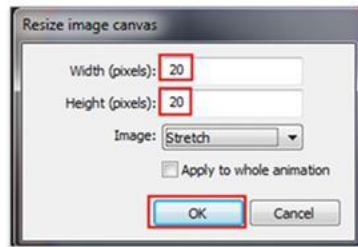
For this object, you're just going to use a box with a filler color to make it easy to see in the editor. First you'll want to resize the object smaller. In the top toolbar click the **resize** button. It is a diagonal line with arrows on each end.



Objective 2 – Create a Level

30 - Resize Dimensions

In the Resize image canvas, set the Width and Height both to 20. Click the **OK** button to resize the canvas.

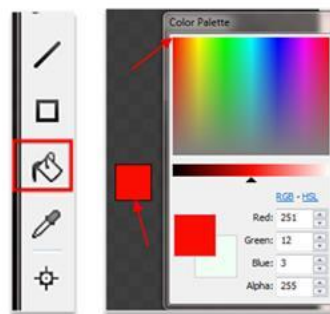


Objective 2 – Create a Level

31 - Color Palette

Next you're going to fill tool to give the object a color and make it easily visible in the editor. On the left toolbar click the **Fill** button. It's the button that looks like a paint can.

This will bring up a Color Palette, click the **upper left corner** of the palette to select a red color. Then click the inside of the canvas to make your object a red box.

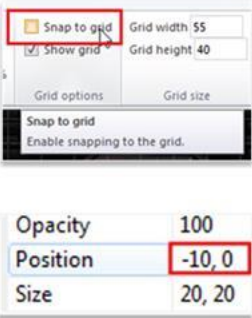


Objective 2 – Create a Level

32 - Edge Snap To Grid

Close the image editor to get to the layout. If still active, uncheck the **Snap to grid** box in the View toolbar.

You'll now want to set the position of the Edge object, so it's just outside of the layout. With the Edge, object selected go to the Properties Bar and set its Position to (-10, 0).




Opacity	100
Position	-10, 0
Size	20, 20

Objective 2 – Create a Level

33 - Resizing The Edge Object

Next, make sure the Edge object is still selected and resize the image, so it covers the entire left edge of the layout. To do this click the bottom middle box surrounding the object and drag to the bottom of the layout.



Objective 2 – Create a Level

34 - Creating Duplicate Edge Objects

You'll want to create a new instance of the Edge object to cover the right edge of the layout. Hold the **Ctrl** key and click and drag the edge object to the right edge of the layout to create a new instance. Make sure it vertically covers the right edge and in its Properties Bar set its X value to 2410.

Position	2410, 339
X	2410
Y	339

The size and Y value of your edge will probably vary from the image. The X value ensures it stays right off the screen, but as long as your object covers the edge the other values don't need a specific value.

Objective 2 – Create a Level


35 - Edge Behaviors

Next, you'll need to add a solid behavior to the object. Click on the Edge object in the Objects Bar and add the Solid behavior to the object.

Play your game now and notice that you no longer can walk off the left and right edge of the layout.

You have now added all that you need for your level at this time. Next, you're going to add a super jump for Flip.

Objective 2 – Create a Level



Behaviors	
Solid	
Initial state	Enabled

36 - Objective 3: Adding A Super Jump

Objective 3

Adding a Super Jump

You'll now add a super jump to Flip. You'll make it so when you hold the down key the super jump will charge and when the up key is pressed it will enable the super jump.


To do this, you'll have to use an instance variable and some events. To start, click on **Flip** in the Objects Bar.

37 - Platform Jump Strength

Before you add the instance variable, you'll want to change one of the values in the Platform Behavior. In the Properties, Bar go down to the Behaviors section and in the Platform behavior change the Jump strength to 750.

Jump strength is measured in pixels per second. The value of 750 was chosen because it was found that most low platforms could be reached without a super jump with that value.

Objective 3 – Adding a Super Jump



Behaviors	
Platform	
Max speed	330
Acceleration	1500
Deceleration	1500
Jump strength	750
Gravity	1500
Max fall speed	1000

38 - Flip Instance Variables

Next you're going to add an instance variable to Flip. An instance variable is a variable that is used in all the instances of that object, but each instance can have a different value for its instance variable.

In this case, you're going to add a variable to store jump boost value. This value will be added to the Jump strength property to increase Flip's jumping ability.

To learn more about Instance Variables [Click Here](#)

Objective 3 – Adding a Super Jump



39 - Flip Instance Variables

With Flip still selected go to the Properties Bar and click the **Instance variables** link.



Objective 3 – Adding a Super Jump



40.1 - Jump Boost Instance Variable

In the Instance Variables window, click the **Plus** button to add a new variable.

This will bring up the New instance variable window. For the name type in **JumpBoost**. Keep the Type as **Number** and the Initial value as **0**. Click **OK** to insert the variable.

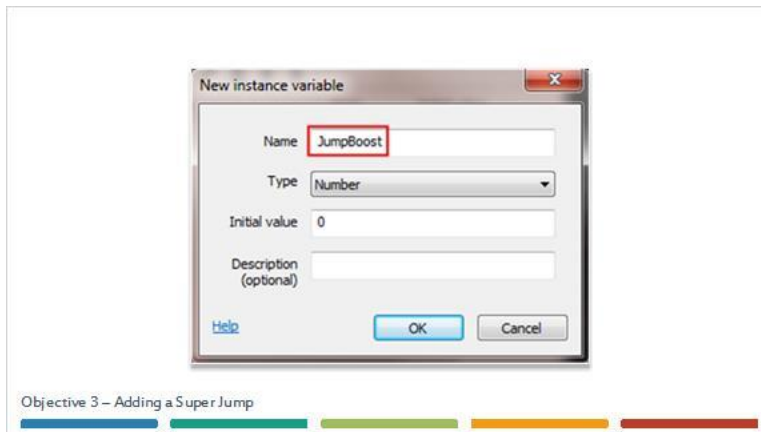
The other types of variables are Boolean that will either be a true or false value. The other type is Text, which stores a string value.

[Show Instance Variable](#)

Objective 3 – Adding a Super Jump



40.2 - Instance Variable JumpBoost



41 - Grouping

You can close the Instance variable window and go to your event sheet. Before you start to add events for the super jump you're going to start to organize your events using groups.

Groups are helpful to keep your event sheet clean and organized. Groups are also able to be activated or deactivated which is helpful to have events trigger at only specific times.

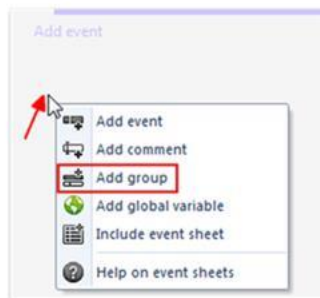
Objective 3 – Adding a Super Jump

42 - Adding A Group

To add a group, go to the bottom of the event sheet and right click on the **empty white space**. In the list that appears click on **Add group**.

To learn more about Groups

[Click Here](#)

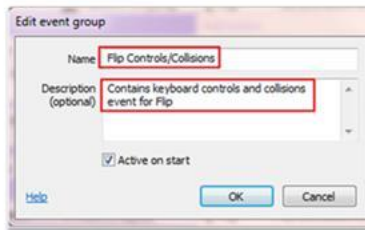


Objective 3 – Adding a Super Jump

43 - Group Settings

In the Edit event group window that appears set the Name field to **Flip Controls/Collisions** and the Description to **Contains keyboard controls and collision events for Flip**.

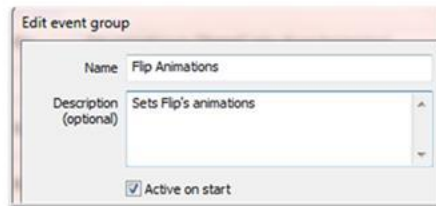
Keep the box checked for **Active on start** and click **OK** to add your group.



Objective 3 – Adding a Super Jump

44 - Add Another Group

Using the same method, add another group. This one should be named **Flip Animations** with a description that says **Sets Flip's animations**. This one will also be active on start.



Objective 3 – Adding a Super Jump

45.1 - Flip Animations Group

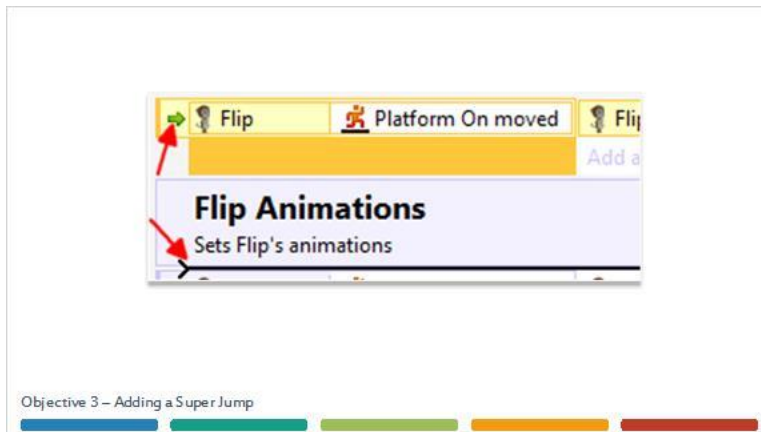
Events in the event sheet can be moved and rearranged by dragging and dropping. To do this, click on the left edge of your first event and drag it down to the Flip Animations group. When you move the event below the group, you'll see an indented black line appear. This indicates, that the event, you're dragging become a sub-event of the event above that line. Release the mouse to move your event to the Flip Animations group.

In the provided image, the Flip Animations group was moved to reduce the image size. Your group will still be at the bottom of your event sheet.

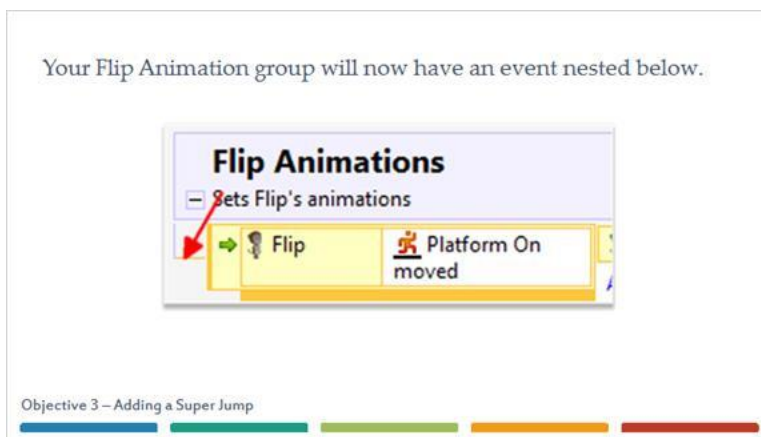
Show Group

Objective 3 – Adding a Super Jump

45.2 - Flip Animations Group Image



46 - Flip Platform On Moved



47.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

◇ Move your other events, except the start of layout event and Flip Controls group, into the Flip Animations group. Make sure as you move the event that they become nested under the Flip Animations group and not as a sub-event of another event.

Show Events

A slide titled "On Your Own" with a subtitle "Complete all the tasks before continuing to the next slide". It contains a task instruction with a diamond icon, a badge with a thumbs up icon and the text "IT'S YOUR TURN" and "On Your Own", and a green button labeled "Show Events". Below the slide, there is a progress bar with five colored segments (blue, green, light green, orange, red).

47.2 - New Event Group

The box on the left edge of the Group will allow you to collapse and expand the events below.



System On start of layout Flip Set animation to "Default" (play from beginning)
Flip Move to top of layer
Add action

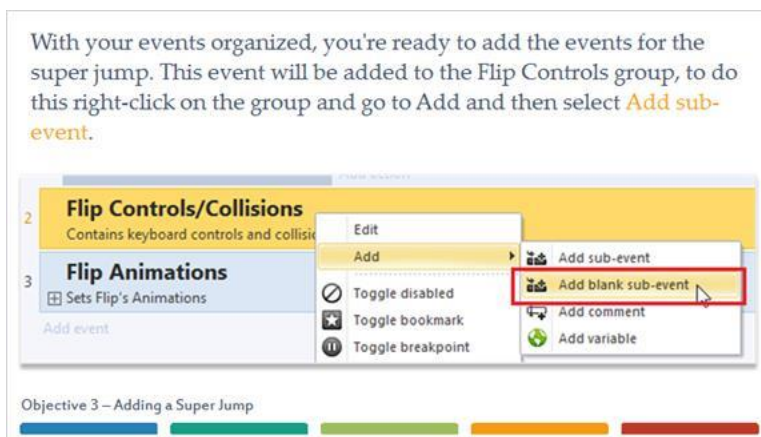
Flip Controls/Collisions
Contains keyboard controls and collisions event for Flip

Flip Animations
Sets Flip's animations

- Flip Platform On stopped Flip Set animation to "Default" (play from beginning)
Add action
- Keyboard Right arrow is down Flip Set Not mirrored
Add action
- Keyboard Left arrow is down Flip Set Mirrored
Add action
- Flip Platform On jump Flip Set animation to "Jump" (play from beginning)
Add action
- Flip Platform On landed Flip Set animation to "Walk" (play from beginning)
Add action
- Flip Platform is moving Flip Set animation to "Default" (play from beginning)
Add action
- Flip Platform is moving Flip Set animation to "Default" (play from beginning)
Add action
- Flip Platform On fall Flip Set animation to "Fall" (play from beginning)
Add action
- Flip Platform On moved Flip Set animation to "Walk" (play from beginning)
Add action

48 - Adding Blank Sub-event

With your events organized, you're ready to add the events for the super jump. This event will be added to the Flip Controls group, to do this right-click on the group and go to Add and then select **Add sub-event**.



Flip Controls/Collisions
Contains keyboard controls and collisions event for Flip

Flip Animations
Sets Flip's Animations
Add event

Objective 3 – Adding a Super Jump

2 Flip Controls/Collisions
3 Flip Animations

Edit
Add
Toggle disabled
Toggle bookmark
Toggle breakpoint

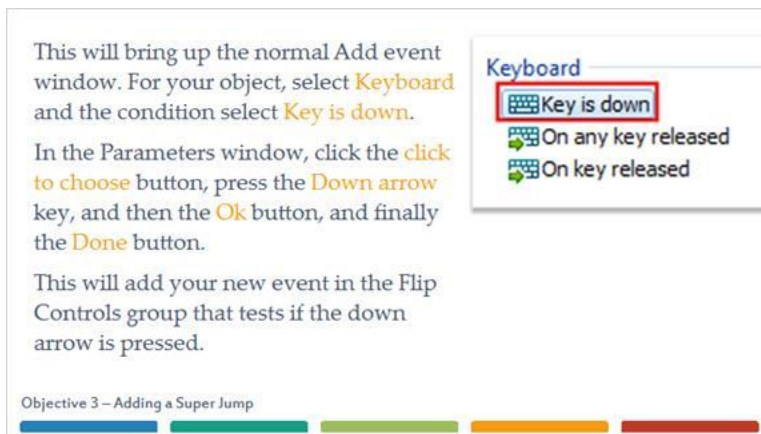
Add sub-event
Add blank sub-event
Add comment
Add variable

49 - Keyboard Key Is Down

This will bring up the normal Add event window. For your object, select **Keyboard** and the condition select **Key is down**.

In the Parameters window, click the **click to choose** button, press the **Down arrow** key, and then the **Ok** button, and finally the **Done** button.

This will add your new event in the Flip Controls group that tests if the down arrow is pressed.



Keyboard

- Key is down
- On any key released
- On key released

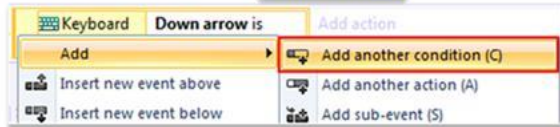
Objective 3 – Adding a Super Jump

50 - Add Another Condition

You'll next add two more conditions to this event. One will test if the JumpBoost variable is less than 400 to prevent Flip from being able to jump too high. The other will test if Flip is on the Ground or Platform object.

To add an additional condition to your event, right click on the event, go to Add, and then press **Add another condition**.

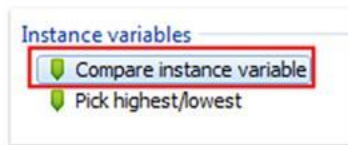
To learn more about Conditions [Click Here](#)



Objective 3 – Adding a Super Jump

51 - Comparing Instance Variables

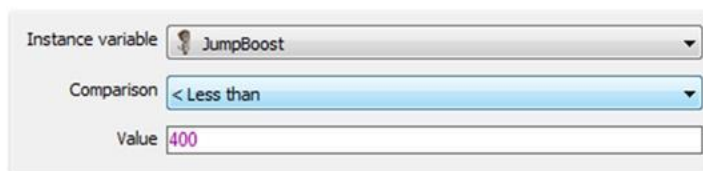
You'll use this condition to test the JumpBoost value. Select the **Flip** object and for the condition select **Compare instance variable** in the Instance variables section.



Objective 3 – Adding a Super Jump

52 - JumpBoost Comparison

Here you'll test that the JumpBoost Value isn't higher than 400. To do this keep the Instance variable field as **JumpBoost** change the Comparison to **Less than** and set to value as **400**. Click **Done** to add the condition.

A screenshot of a configuration panel for a condition. It has three fields: 'Instance variable' with a dropdown menu showing 'JumpBoost', 'Comparison' with a dropdown menu showing '< Less than', and 'Value' with a text input field containing '400'.

Objective 3 – Adding a Super Jump

53 - Flip Is On Floor

Add another new condition to the event and again use the **Flip** object. For the condition, select **Is on the floor** under the Platform section.



Objective 3 – Adding a Super Jump

54 - New Sub-event

Next, you'll add two sub-events to this event. One will test to see if Flip is not moving and if he isn't you'll increase the JumpBoost. The other will test if Flip is moving and set the JumpBoost to 0.

Right click on your event, go to Add, and then select **Add sub-event**.

Objective 3 – Adding a Super Jump

55 - Inverting Flip Is Moving

For your object select **Flip** and for the condition select **Is moving** under the Platform section.

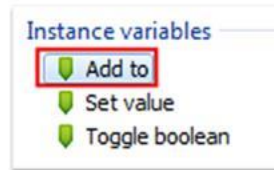
When your sub-event is inserted, right click on the condition and select **Invert**.



Objective 3 – Adding a Super Jump

56 - Adding To Instance Variables

Now you're ready to add the action to increase the JumpBoost. On the new sub-event add an action that uses the **Flip** object and the action **Add to** which is in the Instance variables section.



Objective 3 – Adding a Super Jump

57 - Flip Adding 5 To JumpBoost

When your Parameters window appears, keep the instance variable as **JumpBoost** and set the value to 5. Click **Done** to insert your action.



Objective 3 – Adding a Super Jump

58 - Sub-event Flip Is Moving

Add another new sub-event to the Keyboard Down arrow is down event. The condition for this one will again use the **Flip** object to test if it **Is moving**.



Objective 3 – Adding a Super Jump

59 - Flip Set Value For JumpBoost

With your sub-event added, add a new action to it. Use the **Flip** object and for the action select **Set value** in the Instance variables section.

Click **Done** when the Parameters window appears, because it's already setting the JumpBoost to 0.



Objective 3 – Adding a Super Jump

60 - Keyboard On Any Key Pressed

Next, you'll set up the events that will execute your super jump and remove the super jump if Flip moves before he jumps.

Add a new sub-event off of the Flip Controls group. This event will use the **Keyboard** object and the condition **On any key pressed**.



Objective 3 – Adding a Super Jump

61.1 - Compare Instance Variable

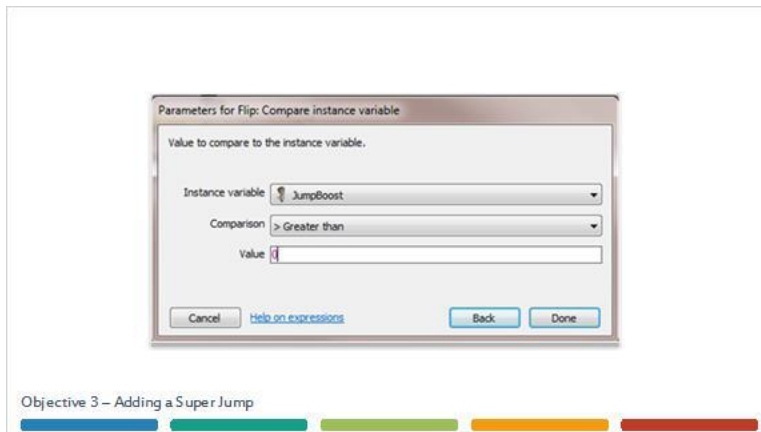
With this event created you'll add another condition to it that will test if JumpBoost has a value. To do this, add a new condition to your new event that uses the **Flip** object and the **Compare instance variable** condition.

For your Parameters keep Instance variable as **JumpBoost**, change Comparison to **Greater than**, and keep the Value at 0. Click **Done** to add your condition.

Show Compare Instance Variable

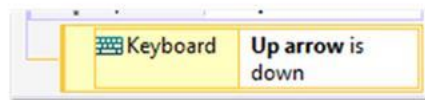
Objective 3 – Adding a Super Jump

61.2 - Compare Instance Variable Image



62 - Keyboard Up Arrow Is Down

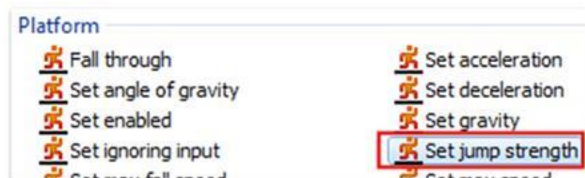
You'll trigger the actions for this event off of sub-events. Add a new sub-event to your newly created event that uses the **Keyboard** object and the **Key is down** condition. Set the Key value to **Up Arrow** and click **Done**.



Objective 3 – Adding a Super Jump

63 - Flip Set Jump Strength

This event will trigger your super jump so you'll add the actions to do that now. Add a new action that uses the **Flip** object and under the Platform section select the **Set jump strength** action.



Objective 3 – Adding a Super Jump

64 - Expressions In Parameters

In the parameters window you're going to use your instance variable for the expression. Expressions are used in parameters to calculate sums or to retrieve information from objects. In this case you're going to use expressions to get the value of the JumpBoost variable and add it to the Jump strength value.

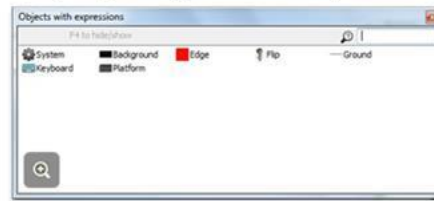
There are two ways to input object expressions into your parameter. Either through typing or through the Expressions panel. You'll use both for this parameter and from there you can decide which method you prefer.

Objective 3 – Adding a Super Jump

65 - Setting Jump Strength

First, you'll use the expression panel to get the JumpBoost variable. The expressions panel will be transparent above the parameters window. You want to access the JumpBoost instance variable on the Flip object, so in the window double click **Flip** and scroll down to find the **JumpBoost** instance variable and double-click it. This will cause the expression to input into the Jump Strength field in the parameters window.

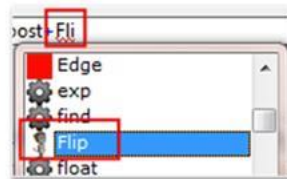
Jump strength `Flip.JumpBoost`



Objective 3 – Adding a Super Jump

66 - Flip Expressions

Next, you'll add the Jump strength value to this. Click on the Jump strength field, so your cursor is after the JumpBoost. Put a + after this and then start to type **Flip**. You'll see a window appear listing some expressions and as you type **Flip** will become highlighted. If you hit the **Enter** key, it will complete **Flip** for you.



Objective 3 – Adding a Super Jump

67 - Flip Jump Strength Expression

After Flip insert a . to be able to access expressions within the Flip object. Access the Jump strength by inputting `Platform.JumpStrength`. Your full completed expression will look like this.

Jump strength `Flip.JumpBoost+Flip.Platform.JumpStrength`

Click **Done** to insert your action.

Moving forward you'll be giving what your final expression will be. You can use either method describe to input that expression.

To learn more about System Expressions [Click Here](#)

Objective 3 – Adding a Super Jump

68 - Locating The JumpBoost Action

On this same sub-event, you're going to add an event that sets the JumpBoost to 0. Instead of adding an entirely new action you'll see how to copy an existing action from another event.

In your event above locate the action that sets JumpBoost to 0.

Objective 3 – Adding a Super Jump

69 - Copying Existing Events And Actions

Select the set JumpBoost to 0 event by clicking it. Now holding the **Ctrl** key, click and drag the action, so a black line appears below the action that sets the jump strength. Release the mouse button when this appears to copy the action.



It's helpful to use opportunities to copy action so save time as you create projects.

Objective 3 – Adding a Super Jump

70 - Super Jump Strength

You can now run your layout and test your super jump. You can see that pressing and holding down then pressing up will cause your jump to be greater than normal, but a few issues are seen.

Currently, the JumpBoost gets stacked, so each time you super jump it adds more to the jump strength. Also, you can currently press down to get a jump boost, move, and then press up and still get a super jump. You want it set, so the super jump goes away if Flip moves.

Objective 3 – Adding a Super Jump



71 - Keyboard Down Arrow Is Down

First, you're going to address the issue of the JumpBoost not resetting when you move. To do this, you'll add two new sub-events to the On any key pressed event.

Add a new sub-event this event that uses the **Keyboard** to test if the Down Arrow key is down.



Objective 3 – Adding a Super Jump



72 - Sub-events For Keyboard

Now you'll add another sub-event for if the left or right arrows or spacebar are pressed. Instead of adding separate conditions for each key you can use an Else event. Else events will fire if conditions in previous events are not met. In this case, if the JumpBoost is greater than 0 and a key is pressed that isn't the Up or Down arrow keys then this event will fire.


Objective 3 – Adding a Super Jump



73.1 - Adding An Else Event

To add the Else event right click on the last sub-event, you added (Down arrow is down) and go to Add and select **Add 'Else'**. You'll see that a new sub-event will be added that uses the System object and the else condition.

Add an action to this sub-event that sets the JumpBoost to 0.



The screenshot shows the event editor interface. On the left, there is a table with columns for 'Keyboard', 'Event', and 'Action'. The first row contains 'Down arrow is down' and 'Add action'. The second row contains 'System', 'Else', and 'Set JumpBoost to 0'. Below the table is a green button labeled 'Click Here To See More Info'. On the right, a context menu is open with the 'Add' option selected. The 'Add Else' option is highlighted in red. Below the menu is a progress bar with five colored segments (blue, green, yellow, orange, red) and the text 'Objective 3 - Adding a Super Jump'.

73.2 - More Information Image

You won't be adding an action to the Down arrow sub-event. This is because the up arrow will fire the super jump, and you want any other key to reset JumpBoost, except the Down arrow. This event will prevent the JumpBoost from resetting when the Down arrow is pressed.

[Click Here To Hide More Info](#)

Objective 3 - Adding a Super Jump



74 - Special Conditions

You have all the events and action set in the Flip Controls group for now. To address the issue of the jump strength not resetting you'll add an action to an event in the Flip Animations group.

To learn more about Special Conditions [Click Here](#)

Dinosaurs are extinct, but that doesn't mean saving has to be.

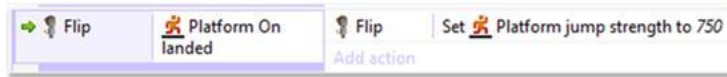
Objective 3 - Adding a Super Jump



75 - Flip Platform On Landed

Go to the Flip Animations group and find the event for Platform On landed (It will have two sub-events).

Add an action to this event that uses the Flip object and will Set jump strength to 750.



Objective 3 – Adding a Super Jump

76 - Super Jump In Action

If you run the layout, you'll now see that you have a super jump fully set up for Flip.

But jumping around can get boring, so next you'll add some bad guys for Flip to defeat!

This completes Flip the Robot Monkey Section 1.

[Click Here](#) to play the completed section.

Objective 3 – Adding a Super Jump

77 - Success

