

1 - Title Slide: Flip The Robot Monkey Section 2



Game:IT
Unit 3

Flip The Robot Monkey Section 2

Powered by
STEM & FUSE
A Revolution in STEM Education

2 - Section Objective

Section Objectives

During this presentation you'll cover the following objectives

- Adding a Knight Enemy
- Adding Lives, Ammo, and Score
- Adding a Coconut Gun
- Adding Flip and Knight Collisions

3 - Objective 1: Adding A Knight Enemy

Objective 1

Adding a Knight Enemy

The next step in your project is to add a Knight enemy to the game. This will be a character that will walk left and right and on collision with Flip will remove a life. The Knight will be able to be defeated by Flip "stunning" the Knight with a coconut gun and then jumping on the Knight's head.

Lives and the coconut gun will be added to the project later.

4 - Knight Sprite

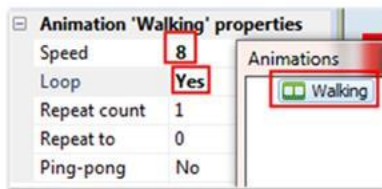
The first thing you'll want to do is add the Knight object and its animations.

Go to your **Level1** layout and add a new object. This object's type will be a **Sprite** and name it **Knight**. When your cross-hairs appear place the object above the Ground object.

Objective 1 – Adding a Knight Enemy

5 - Knight Animation Walking

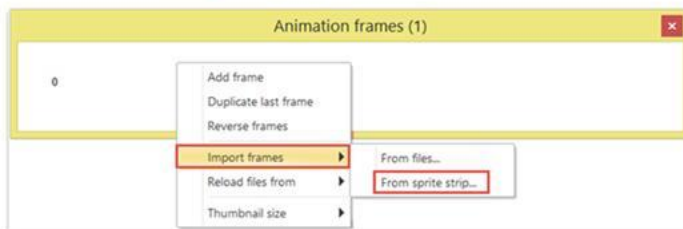
Your Knight object will have three animations: Walking, Swing, and Stunned. The first animation you'll add is the walking animation. Go to the Animations window and rename Default to **Walking**. With Walking selected go to the Properties Bar and set the Speed to **8** and Loop to **Yes**.



Objective 1 – Adding a Knight Enemy

6 - KnightWalk Import Frames

Now you'll add the images for the animation. In the Animation frames window right click, hover over Import frames and select **From sprite strip...** For the file select **KnightWalk**, which will have 8 horizontal cells and 1 vertical cell.



Objective 1 – Adding a Knight Enemy

7 - Knight Swing Animation

Delete the first blank frame from the Animation frames window.



Go to the Animations window, right-click and select **Add animation**. Name this animation **Swing** and you won't have to change any of its properties.

Objective 1 – Adding a Knight Enemy

8 - Import Sprite Strip Stunned

Go to the Animations frames window and import the sprite strip named **KnightSword**. This strip will have 4 horizontal cells and 1 vertical cell. **Delete** the first blank frame.



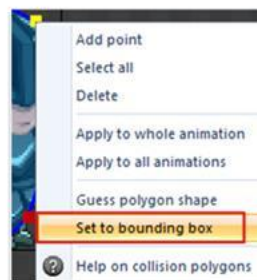
Add the last animation named **Stunned**. This animation will only have a single frame so you can go to the image editor and click the **open** button and select the **KnightStun** file.

Objective 1 – Adding a Knight Enemy

9 - Knight Collision Polygon

The collision polygon for the Knight object isn't as important as the one for Flip as the Knight won't be jumping and moving around as much. For this reason, you can simply set the collision polygon to the bounding box.

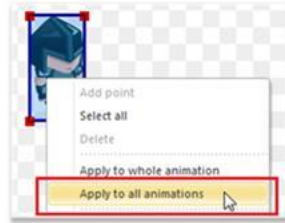
To do this, click the **Collision polygon** button on the left toolbar. When that is selected, right click on the image and select **Set to bounding box**.



Objective 1 – Adding a Knight Enemy

10 - Apply To All Animations

Now you'll want to apply this change to all the animations. Right click on the image again and select **Apply to all animations**. Click **Yes** to confirm your choice and then you can close the image editor.

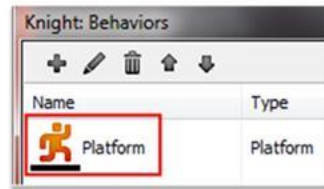


Objective 1 – Adding a Knight Enemy

11 - Platform Behavior

With the object animations set you're ready to add the functionality to the object. In order to get the object to move left and right you're going to add the Platform behavior to it.

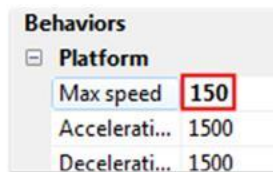
Make sure the Knight object is selected and then go to the Properties Bar and select the **Behaviors** link. Add the **Platform** behavior to the object.



Objective 1 – Adding a Knight Enemy

12 - Platform Max Speed

You'll now make some changes to the Platform properties. The first change is to the Max speed to make the Knight move slower. Go to the Behaviors section of the Properties Bar and change the Max speed to 150.



Objective 1 – Adding a Knight Enemy

13 - Platform Default Controls

The next change you want to make is to turn the default controls off. This will allow you to set the movements of the Knight through events.

Jump strength	650
Gravity	1500
Max fall speed	1000
Default controls	No
Initial state	Enabled

In the Platform properties set the Default controls setting to **No**.

Objective 1 – Adding a Knight Enemy

14 - Instance Variables

The next thing you'll want to do is add a few instance variables to the object. These variables will help control the Knight's movements as well as store a value for the amount of time it's stunned.

Click the **Instance variables** link in the Properties Bar.

Objective 1 – Adding a Knight Enemy

15 - Action Text Variable

Click the **Plus** button to add a new instance variable. The name of this variable will be **Action** and the Type will be **Text**. For the Initial value type in **left**.

New instance variable

Name	Action
Type	Text
Initial value	left
Description (optional)	

Help OK Cancel

Objective 1 – Adding a Knight Enemy

16.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

- ◇ Add another instance variable named PrevAction with a Type of Text and no initial value
- ◇ Add an instance variable named StunTime that is a Number with an Initial value of 0

Show Instance Variables



16.2 - Instance Variables

On Your Own

Complete all the tasks before continuing to the next slide

Name	Type	Initial value
Action	Text	left
PrevAction	Text	
StunTime	Number	0




17 - EdgeMarker Sprite

Before you start adding events to control the Knight you'll want to add another object. This object will be an edge marker for the Knight. You'll use it to indicate when a knight should change directions.

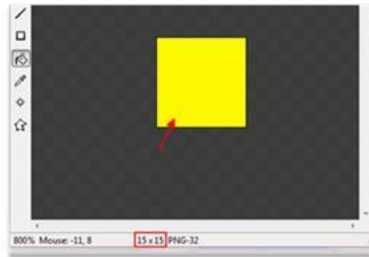
Make sure the Main layer is active and insert a new object that is a **Sprite** named **EdgeMarker**. Click anywhere with the cross-hairs to open the image editor.

Objective 1 – Adding a Knight Enemy



18 - EdgeMarker Image Editor

In the image editor, **Resize** the image to a Width and Height of 15 pixels. Now use the **Fill** tool to make the image a bright color to make it more visible in the layout editor (a bright yellow is suggested).



Close the image editor when this is done.

Objective 1 – Adding a Knight Enemy

19 - EdgeMarker Visibility

There's one property you'll need to change for this object. Go to the Properties section of the Properties Bar and set the Initial visibility to **Invisible**. This will prevent the object from being seen when you run the layout, but collisions and events will still happen with the object.



Objective 1 – Adding a Knight Enemy

20 - Knight EdgeMarker

Now, you'll want to make some changes to your layout so you can properly test the Knight object's event when you need. Create two instances of EdgeMarker object and place them on either side of the Knight object. This will indicate where the Knight will change direction.



Your Knight and EdgeMaker objects should look similar to this.

Objective 1 – Adding a Knight Enemy

21 - Knight Actions Group

You're ready to set up the events for your Knight. To start go to your event sheet and add a new group. Set the name to **Knight Actions** and for the Description put **Set the Knight enemy actions and movements**. Press **Ok** to add the group.



Objective 1 – Adding a Knight Enemy

22 - Compare Knight Actions

Next you'll add the events for the different values of the Action variable. There will be four values possible for Action: left, right, swing, and stun. First you'll set the events for left and right.

Add a sub-event to the **Knight Actions** group. Use Knight as the object and for the condition select **Compare instance variable** in the Instance variables section.

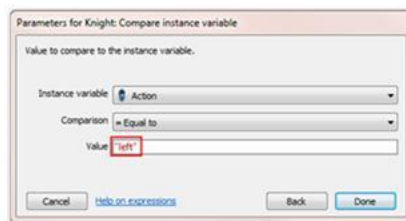
Since the Action variable is text, you'll have to use a string in your expression. Strings are just text used in a programming language. Text that makes up the string has to be included in double-quotes, e.g. "left".

Objective 1 – Adding a Knight Enemy

23 - Compare Action Variable

So in the Parameters window have the Instance variable set to **Action** and the Comparison to **Equal to**.

For the value you'll be testing for the left action, so in this field input "left".



Objective 1 – Adding a Knight Enemy

24 - Simulate Knight Control

For the actions on this event you'll want to have the Knight walk left. To do this you'll use three actions. Add an action to this event that uses the **Knight** object.

For the action, select **Simulate control** under the Platform section. Keep the Control parameter as **Left** and click **Done**.



Platform

- Fall through
- Set angle of gravity
- Set enabled
- Set ignoring input
- Set max fall speed
- Set vector X
- Simulate control**

Objective 1 – Adding a Knight Enemy

25.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

- ◆ Add an action that will set the Knight's animation to Walking
- ◆ Add an action that sets the Knight to Mirrored



Show Knight Actions

25.2 - Knight Actions Image

On Your Own

Complete all the tasks before continuing to the next slide



Knight	Action = "left"	Knight	Simulate Platform pressing Left
		Knight	Set animation to "Walking" (play from beginning)
		Knight	Set Mirrored

Add action

Hide Knight Actions

26 - Next Steps

Next you'll set up an event with actions for when the Knight's Actions is set to right. It will use the same three actions as the event you just set up but will simulate the right control instead of left and will set the knight to not mirrored.

Objective 1 – Adding a Knight Enemy

27.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

◇ Create an event off of the Knight Actions group that will test if the Knight Action variable is "right" and have the three appropriate actions.



Show Knight Actions



27.2 - Show Knight Actions

On Your Own

Complete all the tasks before continuing to the next slide


 Knight	Action = "right"	 Knight	Simulate  Platform pressing Right
 Knight		 Knight	Set animation to "Walking" (play from beginning)
 Knight		 Knight	Set Not mirrored
<small>Add action</small>			

Hide Knight Actions




28 - Knight Stun And Stunned Animation


With the left and right events setup you can add events for the stun and swing actions. Add a new sub-event off of the Knight Actions group that tests the Knight object Action variable is equal to `stun`.



You'll later add a sub-event to this, but for now just add an action that will set the Knight's animation to `Stunned`.



Objective 1 – Adding a Knight Enemy



29 - Knight Swing Animation

For the swing action just add an event that has a condition that tests if Action is equal to `"swing"`. You'll add actions to this event later.



Objective 1 – Adding a Knight Enemy



30 - Knight In Action

If you run the layout, you'll see that the knight will start walking left right away since the default value of the action is left, and keep going until it hits the edge of the layout. To fix this, you'll have to add some events that control what the knight does when it hits the EdgeMarker object.

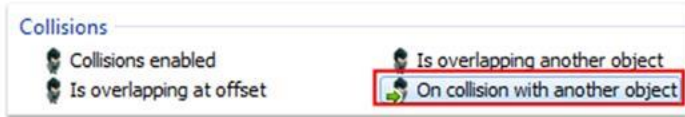
SAVE. SAVE. SAVE. Ahh, that feels better.

Objective 1 – Adding a Knight Enemy



31 - Knight On Collision

Off the of the Knight Actions group add another sub-event that uses the **Knight** object. For the condition select **On collision with another object** under the Collisions section.

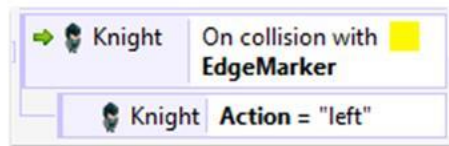


Objective 1 – Adding a Knight Enemy

32 - Knight Action Left

In the Parameters window, set the Object field to **EdgeMarker** and click **Done** to add the event.

Next you'll want to test what the current Action is and you'll do this in a sub-event. Add a sub-event to your new event that uses the **Knight** object and tests if Action is equal to left.



Objective 1 – Adding a Knight Enemy

33 - Knight Set Value Right

If the Knight collides with the EdgeMarker and it's moving left you want the Knight to reverse direction and move right.

Off of the last sub-event, you created add an action that uses the **Knight** object and the **Set value** action under the Instance variables section.

In the Parameters window, set the Action variable to **"right"**. Make sure you input the text as a string using the double-quotes. Click **Done** to insert the action.

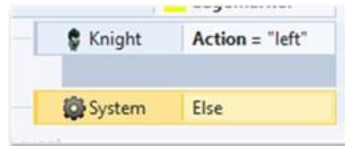


Objective 1 – Adding a Knight Enemy

34 - Knight Else Event

You'll need to add another event that sets the action to left. You could do this by adding an event that tests if the action is equal to left but to make things easy you're going to use an else event.

Right, click on your last sub-event (Knight Action="left"), go to add, and select **Add 'Else'**.



Objective 1 – Adding a Knight Enemy

35 - Knight System Else

In your new Else sub-event add an action that set the Knight's Action variable to left.



If you run the layout, you'll now see that the Knight will move back and forth between where the EdgeMarkers are placed.

Currently, there is no way for Flip and the Knight to interact. To fix this, you're going to add a Coconut gun that Flip can use to stun the Knight and then defeat them by jumping on the Knight.

Objective 1 – Adding a Knight Enemy

36 - Objective 2: Adding A Coconut Gun

Objective 2

Adding a Coconut Gun

To get a coconut gun added to game you'll only have to add one new object and a new animation for Flip. To get started go to your layout.

37 - FlipShoot Animation

You'll first add the animation to Flip. Double click Flip in the Objects Bar to bring up the image editor. In the Animations window, add a new animation and name it **Shoot**. Click on the new animation to make it active.



The shoot animation will be single frame so click the **Open** button and select the **FlipShoot** file.

Objective 2 – Adding a Coconut Gun

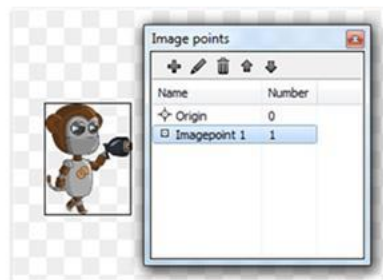
38 - Additional Image Points

For this animation, you're going to add a second image point to the image. Additional image points can be reference in events to be used in a variety of ways. For this instance you're going to use this image point as a place where your coconut will appear.

Objective 2 – Adding a Coconut Gun

39 - Adding Image Points

Click on the **Image points** (cross-hairs) button in the left toolbar. In the Image points window that appears click the **Plus** button to add a new image point. Make sure the new **Imagepoint 1** image point is selected and with the cross-hairs click just in front of the gun in the image to set the point.



Objective 2 – Adding a Coconut Gun

40 - Setting Default Collision Polygon

The last thing to do with Flip is to set the collision polygon for the new animation. In the Animations window, select the **Default** animation. Now in the left toolbar select the **Collision Polygon** tool. Right-click on the image and select **Apply to all animations**. Click **Yes** to confirm and then close the image editor.

Objective 2 – Adding a Coconut Gun

41 - Coconut Sprite

Next, you'll add the coconut object to your layout. Add a new object that is a **Sprite** object type and name it **Coconut**. When the cross-hairs appear, click somewhere outside of the layout to bring up the image editor.

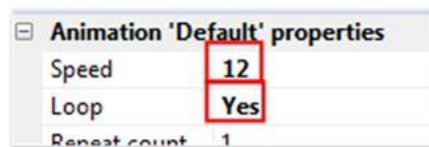
In the Animation frames window, right click, hover over Import frames and select **From sprite strip**. Select the **Coconut** file and it will consist of 6 horizontal and 1 vertical cell. Delete the first blank frame.



Objective 2 – Adding a Coconut Gun

42 - Coconut Animation Properties

Click on **Default** in the Animations window. In the Properties Bar, set the Speed to **12** and Loop to **Yes**. This will give the coconut the illusion of spinning fast. Close the image editor when this is done.



Objective 2 – Adding a Coconut Gun

43 - Bullet Behavior

You'll now want to add a behavior to the Coconut. Since the Coconut will act as a bullet, you'll add the Bullet behavior.

In the Properties Bar, click the **Behaviors** link and add a new behavior. Under the Movements section, select the **Bullet** behavior.

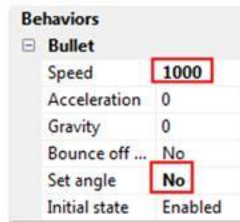


To learn more about Bullet Behavior [Click Here](#)

Objective 2 – Adding a Coconut Gun

44 - Bullet Behavior Settings

With the behavior added, go to the Bullet properties. Set the Speed to 1000 and change Set angle to **No**.

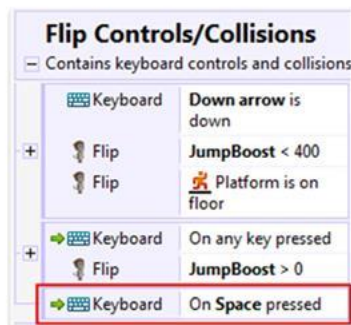


Objective 2 – Adding a Coconut Gun

45 - Keyboard On Space Pressed

You're now ready to add the events for the coconut gun. Go to your event sheet and add a new sub-event to the Flip Controls group.

For this event use the **Keyboard** object and the **On key pressed** condition. Set the Key to **Space** and click **Done** to insert your event.



Objective 2 – Adding a Coconut Gun

46 - Flip Spawn Another Object

For this event add an action that sets Flips animation to "Shoot".



You'll also add an action to this event that will create the Coconut. Add an action that uses the Flip object. Now under the Misc section select the Spawn another object action.



Objective 2 – Adding a Coconut Gun

47 - Spawning Coconut Object

For the Object parameter, select the Coconut object. For the Layer, you'll want to set it to 1, which is the Main layer. Set Image point to 1 also to have the object spawn at the image point you set at the end of the gun.



Objective 2 – Adding a Coconut Gun

48 - Flip Is Mirrored

The bullet behavior will cause the Coconut to travel in a straight line, just like a bullet would. By default the coconut will travel to the right, so you'll have to set an event to have it travel to the left when Flip is facing left.

Create a new sub-event off of the On Space pressed event. For the object select Flip and your condition select Is mirrored under the Appearance section.

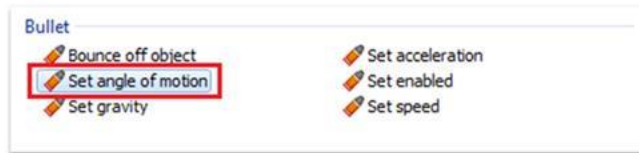


Objective 2 – Adding a Coconut Gun

49 - Coconut Set Angle Of Motion

There are two methods you could use to have the coconut travel left. One method would be to set the speed to -1000. The other method is the one you'll use, and that is to set the angle of motion.

Add an action your sub-event that uses the **Coconut** object and the **Set angle of motion** action.



Objective 2 – Adding a Coconut Gun

50 - Coconut Angle Parameters

In Construct, an angle of 0 degrees is facing right and the angle will increase as you move clockwise. The angle you want is when the coconut travels directly left or 180 degrees.

In the Parameters window set the Angle to 180 and click **Done** to insert the action.

Objective 2 – Adding a Coconut Gun

51 - Quiz

What angle would you use if you wanted the coconut to travel straight up?

- 180
- 270
- 0
- 90

Quiz Time

52 - Coconut Collisions Group

You'll next want to create a new group and events for the coconut collisions. Create a new Group named **Coconut Collisions** and click **OK** to add the group.



Objective 2 – Adding a Coconut Gun

53 - Coconut Is On Screen

The first event you'll add to this group will cause a Coconut to be destroyed when it leaves the screen.

Add a new sub-event to the group that uses the **Coconut** object and for the condition scroll down to the Size & Position section and select **Is on-screen**.



Objective 2 – Adding a Coconut Gun

54 - Inverting Coconut Event

Since you want the event to fire when the Coconut is off the screen, right click on the condition and select **Invert**.

Now add an action to the event that uses the **Coconut** object and for the action select **Destroy** under the Misc section.



Objective 2 – Adding a Coconut Gun

55 - Coconut On Collision

The next event you'll add to this group will destroy the Coconut when it hits a platform. Add a new sub-event to the Coconut Collisions group that uses the **Coconut** object and the **On collision with another object** condition. For the Object select **Platform**.

Add an action to this event that will destroy the Coconut object.

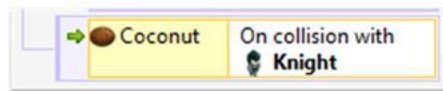


Objective 2 – Adding a Coconut Gun

56 - Coconut Knight Collision

Next you'll add the events for when the Coconut stuns the Knight. When this happens, the Knight will become stunned for 3 seconds, during this time Flip can jump on the Knight to defeat him. If the 3 seconds are up and the Knight hasn't been defeated it will go back to its previous action.

First, add a new sub-event to the Coconut Collisions group that has a condition for the Coconut colliding with the Knight.



Objective 2 – Adding a Coconut Gun

57 - Knight Comparison

To set up the actions for the collision, you'll need two sub-events, one for when the Knight is stunned that will reset the stun time to 3 and the other if he is not stunned that will store the previous action as well as set the stun time.

First add a sub-event to the Coconut on collision with Knight event that uses the **Knight** object and the **Compare instance variable** condition.

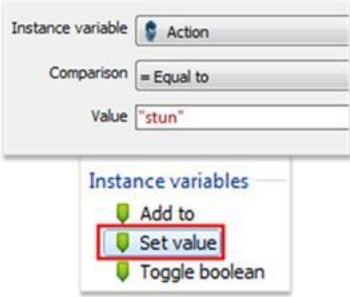
When a collision event occurs in Construct and there are multiple instances of an object only the instance where the collision occurs will be selected in events.

Objective 2 – Adding a Coconut Gun

58 - Instance Variable Set Value

For the Parameters make sure the Instance variable is set to **Action**, the Comparison to **Equal to**, and the Value set to "stun".

Now add a new action to the sub-event and use the **Knight** object and under the Instance variables section select **Set value**.



Objective 2 – Adding a Coconut Gun

59 - Knight Set StunTime


In the Parameters window select **StunTime** for the Instance variable and in the Value field insert 3. Click **Done** to add the action and then add another action to the event that will destroy the Coconut object.



Objective 2 – Adding a Coconut Gun

60 - System Else StunTime

Right click on your sub-event and add an **'Else'** sub-event. Copy the two actions for your sub-event above that set the StunTime and destroy to Coconut to your new Else sub-event.



Objective 2 – Adding a Coconut Gun

61 - Knight Set Value PrevAction

Next you'll want to add an action to store the Knight current Action in the PrevAction variable.

Add an action that uses the Knight object and the Set value action.

For the Parameters set the Instance variable to PrevAction. For the value, you'll want the Knight's current value in its Action variable. To get this the value will be Knight.Action.



Objective 2 – Adding a Coconut Gun

62 - Knight Set Action Stun

Now you'll want to add another action that will use the Knight object and set the Action variable to "stun".

Since Construct will run events from top to bottom as they are on the event sheet, it's important that the action that sets the Knight's action to stun is below that action that set the PrevAction variable.

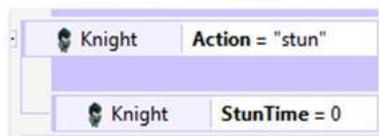


Objective 2 – Adding a Coconut Gun

63 - Knight Compare Instance Variable

You'll now set up the events that will countdown the StunTime of the Knight and reset its action when it's no longer stunned.

In the Knight Actions group, find the event that tests if Action is equal to stun. Off of this event add a new sub-event that uses the Knight object and the Compare instance variable condition. For the Parameters set Instance variable to StunTime and click Done.



Objective 2 – Adding a Coconut Gun

64 - Knight Sete Value Action

On this sub-event add an action that uses the **Knight** object. For the action select **Set value** in the Instance variables section.

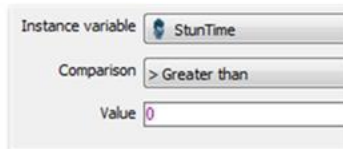
For the Parameters you'll want Instance variable to be set to **Action**. For the value field, you'll use the PrevAction variable. To access the variable input **Knight.PrevAction** in the Value field.

Objective 2 – Adding a Coconut Gun

65 - Knight Compare StunTime

Next you'll add an event that will count down the stun time of the Knight. Add a new sub-event off of the Knight Actions group that uses the **Knight** object and the **Compare instance variable** condition.


For the Parameters set Instance variable to **StunTime**, Comparison to **Greater than**, and keep Value at 0.



Objective 2 – Adding a Coconut Gun

66 - Every X Seconds

Next you'll add another condition to this event that will cause it to run every second. Right click on the sub-event, go to Add and select **Add another condition**. For the object select System and for the condition scroll down to the Time section and select **Every X seconds**.



To learn more about the Every X Seconds [Click Here](#)

Objective 2 – Adding a Coconut Gun

67 - StunTime Countdown

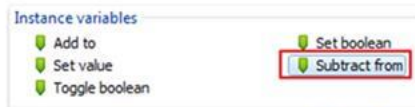
With this condition, you can specify how often, in seconds, you would like the event to run. For this instance, you need to event to run once per second to countdown the StunTime.

Keep the Initial value as 1.0 and click **Done**.

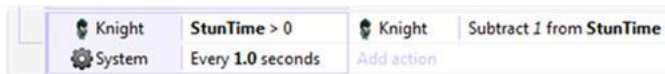
Objective 2 – Adding a Coconut Gun

68 - Knight Subtract From

Now you can add the action to the event. Use the **Knight** object and for the action select **Subtract from** under the Instance variables section.



For the Instance variable parameter select **StunTime** and keep the Value as 1. Click **Done** to complete the action.



Objective 2 – Adding a Coconut Gun

69 - Coconut In Action

If you run the layout now and shoot a coconut at the Knight by hitting Space, you'll see that the knight will be "stunned". After the three seconds expire, the Knight will begin to walk again.

Hold your horses. It's time to save, partner.

Objective 2 – Adding a Coconut Gun

70 - Objective 3: Adding Flip And Knight Collisions

Objective 3

Adding Flip and Knight Collisions

The final thing you'll do is set up the events for when Flip and the Knight collide.

Add a new sub-event onto the Flip Controls group. Select **Flip** as the object and for the condition select **On collision with another object**. For the Object parameter, select **Knight**.

71.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

◇ You'll first set up the event for if the Knight is stunned. Add a sub-event to your new event that checks if the Knight's Action variable is equal to "stun".





71.2 - Knight Stun Action Image

On Your Own

Complete all the tasks before continuing to the next slide

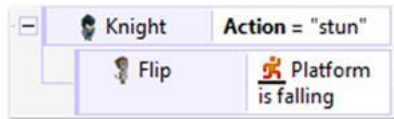




72 - Flip Is Falling

Next you'll want to test if Flip is falling when the collision happens. You'll do this as a sub-event of the last sub-event you added.

Add a sub-event to the last sub-event that uses the **Flip** object and the **Is falling** condition under the Platform section.



Objective 3 – Adding Flip and Knight Collisions

73 - Flip Compare Y

Now you'll add a condition onto this event to test if Flip is above the Knight when the collision occurs. Remember the Y-values in Construct start with 0 and increase as you move down.

Add a condition to the sub-event that uses the **Flip** object and the **Compare Y** condition under the Size & Position section. For the Comparison select **Less than**.

Objective 3 – Adding Flip and Knight Collisions

74 - Flip Compare Y

For the Y coordinate type in **Knight.Y-40**. Since the Y coordinate of the Knight is based on its origin point, you need to remove 40 to get the co-ordinate near the top of the image.

Click **Done** to add the condition.



Objective 3 – Adding Flip and Knight Collisions

75 - Knight Set Vector Y

Next you'll add actions to the event where your condition was just added. The first action will destroy the Knight. Add this action by using the **Knight** object and the **Destroy** action.

Next you'll want to add an action that will simulate Flip bouncing up off the Knight. To do this, add an action that uses the **Flip** object and the **Set vector Y** action in the Platform section.

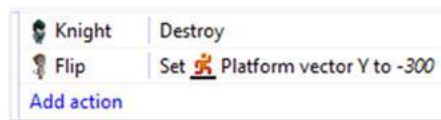


Objective 3 – Adding Flip and Knight Collisions

76 - Flip Set Platform Vector Y

The Y vector will set the vertical motion of Flip, with positive values setting the motion down and negative values setting the motion up.

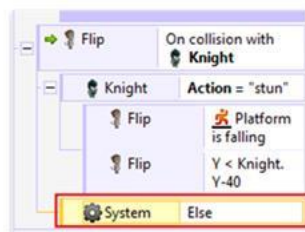
For your Vector Y field set the value to -300. Press **Done** to add the action.



Objective 3 – Adding Flip and Knight Collisions

77 - System Else


Next you'll set up the events for when Flip collides with the Knight when it's not stunned. Right-click on the event that tests if Action is equal to stun, go to add, and select **Add 'Else'**.



Objective 3 – Adding Flip and Knight Collisions

78 - Flip Set Ignoring Input

The first action you'll want to set on this event is one what will disable Flip from moving. So add a new action to your Else event that uses the **Flip** object and under the Platform section select the **Set ignoring input** action.




Change the Input parameter to **Start ignoring** and click **Done**.

Objective 3 – Adding Flip and Knight Collisions

79 - Knight Set Max Speed

The next thing you'll want to do is stop the Knight from moving. Add an action that uses the **Knight** object and the **Set max speed** action under the Platform section. Set the Max Speed parameter to **0** and click **Done**.



Objective 3 – Adding Flip and Knight Collisions

80.1 - Knight Actions

On Your Own

Complete all the tasks before continuing to the next slide

- ◆ Add an action to the Else event to set the Knight's Action to "swing"
- ◆ Add an action that sets the Knight's animations to "Swing"

Show Knight Actions



Objective 3 – Adding Flip and Knight Collisions

80.2 - Knight Action Image

On Your Own
Complete all the tasks before continuing to the next slide

 Knight	Set Action to "swing"
 Knight	Set animation to "Swing" (play from beginning)

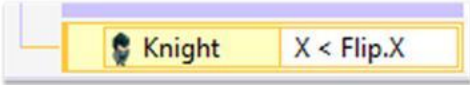
Hide Knight Actions




81 - Knight Compare X

The last thing you'll want to do is set up a couple sub-events to make sure the Knight is facing the direction of Flip. Add a sub-event to the Else event that uses the **Knight** object and the **Compare X** condition.

In the Parameters window, set Comparison to **Less than** and the X coordinate to **Flip.X**.



Objective 3 – Adding Flip and Knight Collisions



82.1 - On Your Own

On Your Own
Complete all the tasks before continuing to the next slide

-  Add an action to the last create sub-event that sets the Knight to Not mirrored
-  Add a sub-event off of the Else event that test is the Knight's X value is greater or equal to Flip's X
-  Add an action to this sub-event that sets the Knight to Mirrored


Show Knight Actions



82.2 - Knight Actions Image

On Your Own

Complete all the tasks before continuing to the next slide

Add action	
Knight	X < Flip.X
Add action	
Knight	Set Not mirrored
Add action	
Knight	X ≥ Flip.X
Knight	Set Mirrored
Add action	

[Hide Knight Actions](#)




83.1 - Example Level

Go to your layout and add more instances of the Knight. It is recommended to have 4 or 5 instances. Make sure to include EdgeMarkers to contain your Knights.

The images below will show an example of an image.

[Click To Show Level](#)

Objective 3 – Adding Flip and Knight Collisions



83.2 - Level



Objective 3 – Adding Flip and Knight Collisions



84 - Testing Your Layout

Run your layout a few times to see how you can now defeat Knights and how they will defeat Flip.

The next thing you'll want to do is add a few key features to the game. These would be a score, lives, and ammo for the coconut gun. As we did previously, we will add the necessary objects to the game and then move on to the events.

Objective 3 – Adding Flip and Knight Collisions

85 - Objective 4: Adding Lives, Ammo, And Score

Objective 4

Adding Lives, Ammo, and Score

In many platform games the main character will collect coins as a way to earn points. But being as Flip is Robot Monkey, and would have no use for coins, you're going to use bananas. As Flip moves through the level he will collect bananas and earn points.

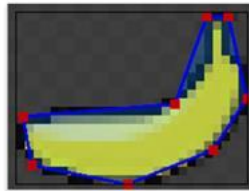
On your layout add a new Object that is a **Sprite** and name it **Banana**.



86 - Banana Collision Polygon

When the cross-hairs appear click on above the first platform in your level to bring up the image editor. In the image editor click the **Open** button and select the **Banana** file to import your banana image.

The only other thing you'll need to do in the image editor is fix the collision polygon. Click on the **Collision Polygon** button in the left toolbar. Click and drag the points until you have the polygon roughly around the banana. Close the image editor when this is done.



Objective 4 – Adding Lives, Ammo, And Score

87 - Ammo Sprite

The Banana object won't need any other behavior or property changes. So, in your layout add more instances of your Banana throughout the level.

Next you'll add the object for the coconut gun ammo. Add a new object that is a **Sprite** and name it **Ammo**. Click above the first platform in your level with the cross-hairs to bring up the image editor.

Another example of the layout with Ammo and Banana objects placed will be provided after more elements are added to the layout.

Objective 4 – Adding Lives, Ammo, And Score

88 - Adding Ammo

In the image editor, click the **Open** button and select the **Ammo** file. No other steps are needed so close the image editor when the ammo image is imported.

In the layout add one or two more instances of the Ammo object to your level.

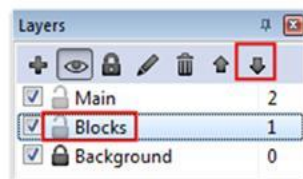
When you add the score, lives, and ammo to your game you're going to want to display them on the screen. To do this, you're going to add a new layer to your game. This layer will be for an HUD, or heads up display. This is found and many games and is used to display relevant information for the player.

To begin click on the **Layers** tab on the right side of the screen.

Objective 4 – Adding Lives, Ammo, And Score

89 - New Layer Blocks

Click the **plus** button to add a new layer. **Rename** the layer to **Blocks** and click it to make it active. Next click the **Down arrow** button in the Layers Bar to move this layer below the Main layer.



Objective 4 – Adding Lives, Ammo, And Score

90.1 - Parallax Settings

Now we're going to add a heads-up display (aka HUD). It's the interface that shows the player's health, score and other information in-game. First, add another new layer and **Rename** it HUD. This layer can stay on the top. Click it to make it active and go to the Properties Bar. Change the value of Parallax to 0, 0.

Parallax allows different layers to scroll at different rates for a sort of semi-3D effect. If we set the parallax to zero, though, the layer won't scroll at all, which is ideal for an HUD.

Show Parallax Settings

To learn more about Parallax Scrolling [Click Here](#)

Objective 4 – Adding Lives, Ammo, And Score

90.2 - Parallax Settings Image

Layer properties	
Name	HUD
Initial visibility	Visible
Background co...	<input type="checkbox"/> 255, 255, 255
Transparent	Yes
Opacity	100
Force own text...	No
Scale rate	100
Parallax	0, 0

Objective 4 – Adding Lives, Ammo, And Score

91 - Background Parallax

In the Layers Bar, click on the **Background** layer and then in the Properties Bar change its Parallax value to 75, 75.

The higher the value you enter, the faster that layer appears to scroll.

Opacity	100
Force own text...	No
Scale rate	100
Parallax	75, 75
Editor properties	

Objective 4 – Adding Lives, Ammo, And Score

92 - Moving Objects To Different Layers

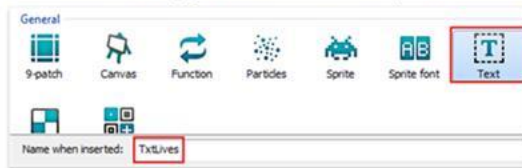
Before you add the objects for your HUD, you're going to change the layer of a couple of objects. With a layout tab selected (not an event sheet tab) click on the **Ground** object in the Objects Bar or the layout itself. In the Properties Bar, change its Layer to **Blocks**. This will make sure the Flip and Knight objects appear on top of the ground. Do the same for the **Platform** object as well.

Common	
Layer	Blocks
Angle	0
Opacity	100

Objective 4 – Adding Lives, Ammo, And Score

93 - Text Objects

Make sure your active layer is HUD and add a new object to your layout. Select Text as the type and name the object TxtLives.



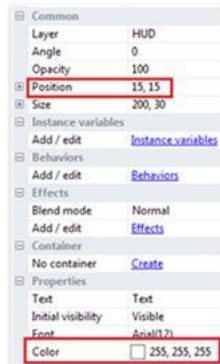
Click in the upper left of your layout when the cross-hairs appear to insert the object.

To learn more about the Text Object [Click Here](#)

Objective 4 – Adding Lives, Ammo, And Score

94 - Text Object Settings

With the object selected go to the Properties Bar and change its Position to (15, 15). Next go to the Properties section and change the Color setting to **White**.



The image shows the Properties Bar for a Text object. The 'Position' field is set to '15, 15' and the 'Color' field is set to '255, 255, 255'. Both fields are highlighted with red boxes.

Common	
Layer	HUD
Angle	0
Opacity	100
Position	15, 15
Size	200, 30
Instance variables	
Add / edit	Instance variables
Behaviors	
Add / edit	Behaviors
Effects	
Blend mode	Normal
Add / edit	Effects
Container	
No container	Create
Properties	
Text	Text
Initial visibility	Visible
Font	Arial(12)
Color	255, 255, 255

Objective 4 – Adding Lives, Ammo, And Score

95.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

- ◇ Add two new Text objects. One named TxtAmmo, the other TxtScore
- ◇ Set the position of TxtAmmo to (1100, 15) and TxtScore to (1100, 50)
- ◇ Change both of their Color properties to White



[Show Level Example](#)



95.2 - Level Example

On Your Own

Complete all the tasks before continuing to the next slide



[Hide Level Example](#)



96.1 - Level Example


Since your newly created TxtAmmo and TxtScore object will be displayed on the right side of the screen, you'll want to change their text alignment to the right.

In the Properties Bar, under the Properties section, set the Horizontal alignment of both to **Right**.

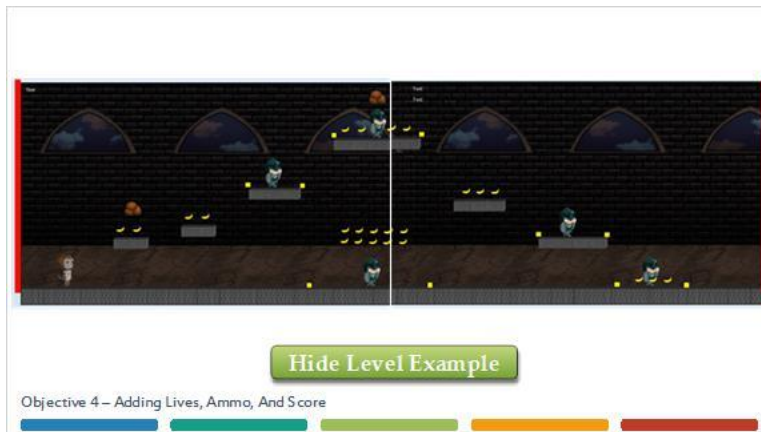
Font	Arial(12)
Color	<input type="checkbox"/> 255, 255...
Horizontal alignment	Right
Vertical alignment	Top

[Show Level Example](#)

Objective 4 – Adding Lives, Ammo, And Score



96.2 - Level Example Image



97 - TxtAmmo Object

Now you're ready to set up your events. The first thing you'll want to do is set it so the TxtAmmo and TxtScore objects are on the right side of the screen. Since many screens are not the same size you'll want to set it so these objects will be positioned based off the screen size.

Go to the event sheet and add a new action to the Start of layout event. Select the **TxAmmo** object and for the action select **Set X** under Size & Position.

Objective 4 - Adding Lives, Ammo, And Score

98.1 - Viewport Right

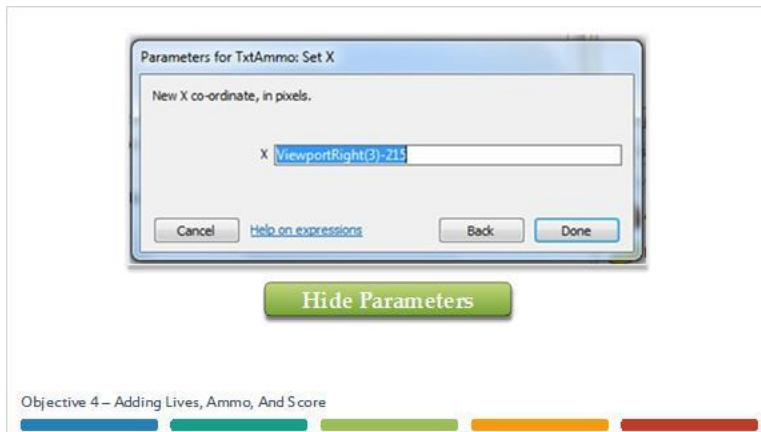
For the X value, you're going to use the expression ViewportRight. This will return the X value of the right side of the viewport boundaries of a layer. In this case, you'll use ViewportRight to set the X position to 215 pixels less than the right edge. The three within the parentheses refers to the layer number. In this case, the 3 refers to the HUD layer.

In the X field input `ViewportRight(3)-215`.

Show Parameters

Objective 4 - Adding Lives, Ammo, And Score

98.2 - Parameters Image



99 - Adding Global Variables

On the Start of layout event add another action that sets the X of TxtScore to `ViewportRight(3)-215`.

You're now going to add the score, ammo, and lives to the game as Global Variables. Global variables differ from instance variables in that they aren't tied to any certain object and can be used across your game. Unlike instance variables, they can only have one unique value whereas instance variables can be different for different instances of an object.

Objective 4 – Adding Lives, Ammo, And Score

100 - New Global Variable

To add an global variable, right click on the blank space of the bottom of your event sheet and select **Add global variable**.


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

Objective 4 – Adding Lives, Ammo, And Score

101 - Score Global Variable

Name your variable **Score**. Keep the Type as a **Number** and the Initial value as **0** and click **OK** to add your variable.

You'll see that the variable has been added to the top of the event sheet and is ready to be used.

 Global number **Score = 0**

Objective 4 – Adding Lives, Ammo, And Score

102.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

- ◇ Add a new global variable named AmmoAmt. It will be a Number with an Initial value of 3.
- ◇ Add another global variable this one named Lives. It will also be a number with an Initial value of 3.






Show Global Variables

102.2 - Global Variables Image

On Your Own

Complete all the tasks before continuing to the next slide

 Global number **Lives = 3**
 Global number **AmmoAmt = 3**
 Global number **Score = 0**

Hide Global Variables

103 - System Every Tick

With the variables added you can set the events that will display the values in the text objects. You'll do this using an event that will run every tick. A tick will occur about 60 times per second, so essentially this will run all the time.

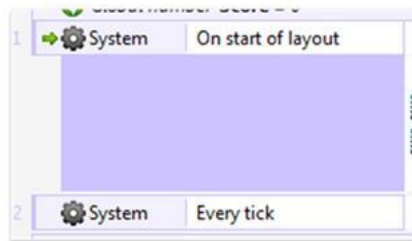
Add a new event that uses the **System** object and the **Every tick** condition.



Objective 4 – Adding Lives, Ammo, And Score

104 - Moving Events

To keep your event sheet organized, move the **Every tick** right below the **Start of Layout** event.



Objective 4 – Adding Lives, Ammo, And Score

105 - TxtLives Set Text

Now add an action to this event that uses the **TxtLives** object and the **Set Text** action.

In the parameters, you're going to use the first instance of combining strings and a number in an expression. These can be combined by using an **&** symbol.



Objective 4 – Adding Lives, Ammo, And Score

106 - Text Setting

In the Parameters window set the Text field to "Lives: " & Lives. Note the space after the colon before the quotes closes and that the second Lives used is the Lives variable. Click Done to insert the action.



Objective 4 – Adding Lives, Ammo, And Score

107.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

◇ Add two more actions to the every tick event. One will set the text of TxtAmmo to "Ammo: " & AmmoAmt. The other will set the text of TxtScore to "Score: " & Score.

A circular badge with a thumbs-up icon in the center. The text "IT'S YOUR TURN" is written around the top inner edge of the circle. Below the circle is a ribbon with the text "On Your Own" and two stars on either side.

Show Every Tick Event

107.2 - Every Tick Image

On Your Own

Complete all the tasks before continuing to the next slide

System	Every tick	TxtLives	Set text to "Lives: " & Lives
		TxtAmmo	Set text to "Ammo: " & AmmoAmt
		TxtScore	Set text to "Score: " & Score

Hide Every Tick Event

108.1 - Score Action

Now you'll just need to set up the event that will change the values of the global variables.

Go into the Flip Controls group and in the Flip on collision with Knight event find the action that Destroys the Knight. Add a new action that uses the **System** object and the **Add to** action under the Global & local variables section. Set the Variable to **Score** and the value to **100** and click **Done** to add the action.

Show Score Action

Objective 4 – Adding Lives, Ammo, And Score

108.2 - Score Action Image

Object	Event	Action
Flip	On collision with Knight	Add action
Knight	Action = "stun"	Add action
Flip	Platform is falling	Knight Destroy
Flip	Platform is falling	Flip Set Platform vector Y to -300
Flip	Y < Knight.Y-40	System Add 100 to Score

Hide Score Action

Objective 4 – Adding Lives, Ammo, And Score

109.1 - Ammo Event

Next add a new sub-event to the Flip Controls group. Use the **Flip** object and the **On collision with another object condition**. Select **Ammo** as the Object in the parameters.

Add another condition to this event that uses the **System** object and **Compare variable** condition. Set to the parameters to test if **AmmoAmt** is **Less than 9**.

This condition will only let Flip pick up ammo if he has less than 9.

Show Score Action

Objective 4 – Adding Lives, Ammo, And Score

109.2 - Ammo Event Image



On collision with **Ammo**

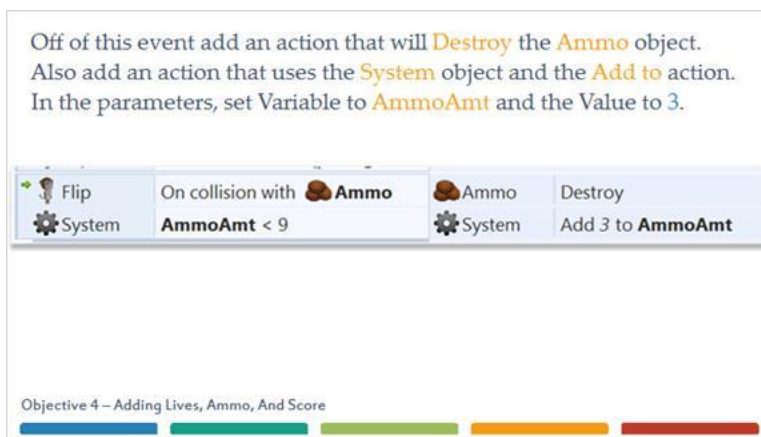
AmmoAmt < 9

Hide Score Action

Objective 4 – Adding Lives, Ammo, And Score

110 - System Add to AmmoAmt

Off of this event add an action that will **Destroy** the **Ammo** object. Also add an action that uses the **System** object and the **Add to** action. In the parameters, set Variable to **AmmoAmt** and the Value to 3.



On collision with **Ammo**

AmmoAmt < 9

Ammo Destroy

System Add 3 to **AmmoAmt**

Objective 4 – Adding Lives, Ammo, And Score

111 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

- Next you'll need to add a sub-event that will limit the amount of ammo that Flip can have at one time.
- Make this new event for object **System** with the condition of **Compare variable**. Set the Variable to **AmmoAmt**, Comparison to **Greater or equal** and the Value to 9.



Objective 4 – Adding Lives, Ammo, And Score

112.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

Next Add an action for object **System** to **Set value of AmmoAmt** to 9.

Show New Sub-event




112.2 - Sub-event Image

On Your Own

Complete all the tasks before continuing to the next slide

19	Flip	On collision with Ammo	Ammo	Destroy
	System	AmmoAmt < 9	System	Add 3 to AmmoAmt
20	System	AmmoAmt ≥ 9	System	Set AmmoAmt to 9

Hide New Sub-event




113 - System AmmoAmt Subtract From

With Flip now collecting Ammo, you'll want to set it up so when the gun is fired the AmmoAmt decreases. In Flip Controls group, find the event for when Space is pressed. On this event add an action that will use the **System** object and the **Subtract from** action. Set the Variable to **AmmoAmt** and keep the value as 1.

Keyboard	On Space pressed	Flip	Set animation to "Shoot" (play from beginning)
		Flip	Spawn Coconut on layer 1 (image point 1)
		System	Subtract 1 from AmmoAmt

Objective 4 – Adding Lives, Ammo, And Score



114 - AmmoAmt Greater Than

You'll also want to add a condition to test if Flip has ammo to fire. On the same Space pressed event add a condition that will use the **System** object to test if **AmmoAmt** is **Greater than 0**.



Objective 4 – Adding Lives, Ammo, And Score

115.1 - On Your Own

On Your Own

Complete all the tasks before continuing to the next slide

- ◆ Add a sub-event to the Flip Controls group that tests if Flip collides with a Banana object. Add actions to this event that will destroy the Banana and add 5 to the Score variable.



Show Banana Event

115.2 - Banana Event Image

On Your Own

Complete all the tasks before continuing to the next slide



Hide Banana Event

116 - Knight On Any Finished

Next you'll set the event that will remove a life when a Knight collides with Flip. In the Knight Actions group, locate the event that tests if Action is equal to "swing". Off of this event, add a sub-event that uses the **Knight object** and the **On any finished** condition in the Animations section.

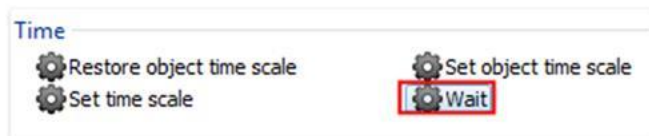


Objective 4 – Adding Lives, Ammo, And Score

117 - System Wait

Add an action to this sub-event that uses the **System** object and the **Subtract from** action. Set the Variable to **Lives** and keep the Value as 1.

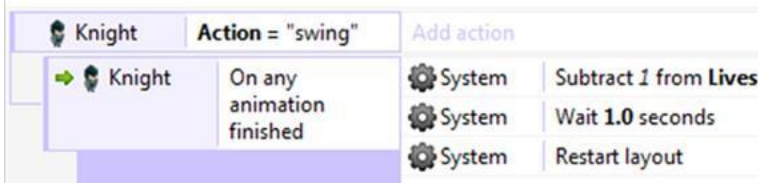
Next add an action that uses the **System** object and an action called **Wait** which is under the Time section at the bottom.



Objective 4 – Adding Lives, Ammo, And Score

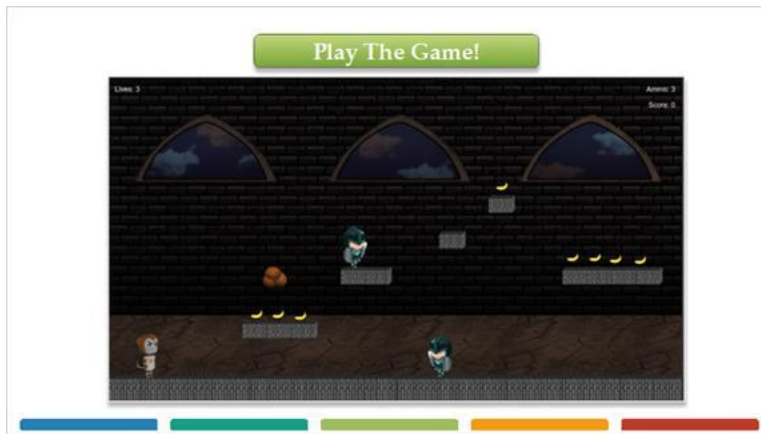
118 - Restart Layout

Keep the Seconds parameter as 1.0 and click **Done**. Now add a final action that uses the **System** object and the **Restart layout** action.



Objective 4 – Adding Lives, Ammo, And Score

119 - Play The Game!



120 - Success

