

Using the IF and WHILE Control Statements

🎯 What Will I Learn?

Objectives

In this lesson, you will learn how to:

- Use the IF control statement to effect execution of instructions
- Use the WHILE control statement to create a conditional loop for repetitive behavior





Why Learn It?

Purpose

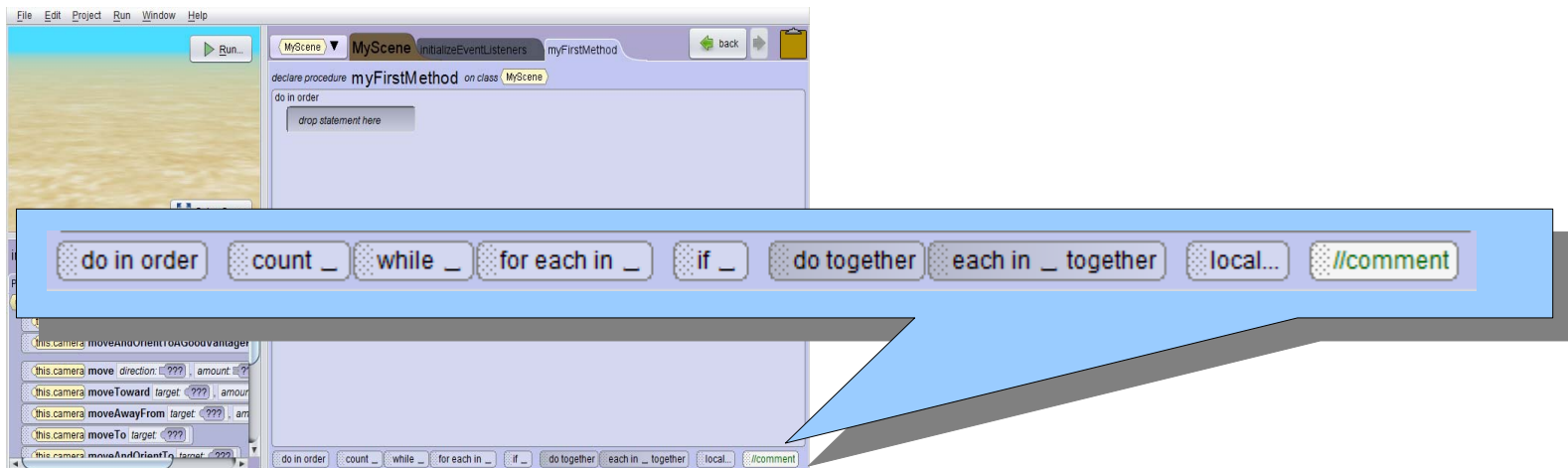
When you create an animation or a game you may want an object to execute an action based upon a decision. You can use the IF control structure to help an object make a decision. “If you encounter a rock turn left, otherwise continue going forward”

You may also want objects in your animation to execute an action while some condition is true. “While the boat moves left wave the person’s hand”. You can use the WHILE control structure to continuously loop an action while a condition is true.



Control Statements

Control statements are pre-defined statements that determine the order of execution for instructions.



For example:

1. Do In Order, where procedures are executed sequentially
2. Do Together, where procedures are executed simultaneously



Control Statements Nested

Control statements may be nested, meaning that one or many are embedded within another.

```
declare procedure myFirstMethod on class MyScene
do in order
do together
  this.myAdultPerson turn LEFT, 0.25 add detail
do together
  this.myAdultPerson move FORWARD, 2.0, duration 0.5 add detail
  do together
    this.myAdultPerson getLeftKnee turn FORWARD, 0.125, duration 1.0 add detail
    this.myAdultPerson getLeftHip turn FORWARD, 0.125, duration 1.0 add detail
    this.myAdultPerson getRightShoulder turn LEFT, 0.25 add detail
  do together
    this.myAdultPerson move FORWARD, 2.0, duration 0.5 add detail
    this.myAdultPerson getRightKnee turn FORWARD, 0.125, duration 1.0 add detail
    this.myAdultPerson getRightHip turn FORWARD, 0.125, duration 1.0 add detail
    this.myAdultPerson getLeftShoulder turn RIGHT, 0.25 add detail
    this.myAdultPerson turnToFace this.asteroid5Gray add detail
    this.myAdultPerson moveAndOrientTo this.asteroid5Gray add detail
```



Control Statements IF and WHILE

Additional control statements are:

1. IF control statement
 - A statement executes, or does not execute, dependent on the result of a condition
 - Also known as “conditional execution”
2. WHILE control statement
 - A statement, or block of statements, are run repeatedly
 - Also known as “repetition”



IF Statement

If it rains today, then I will wear a rain coat. Otherwise, I will wear a sweater.

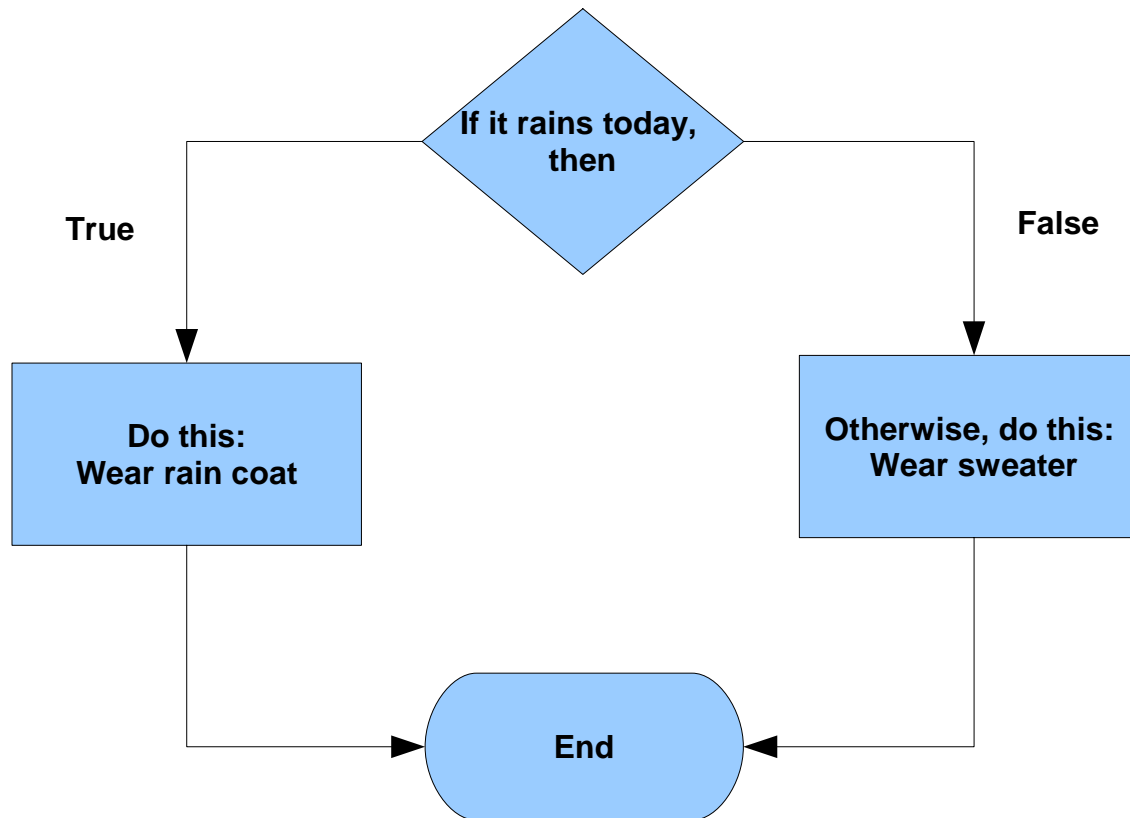
The above may appear to be a simple decision, but to a programmer this is a decision based on a condition.

Let's interpret this condition as a process flow.

A process flow is a graphical representation of a process model. Process flows use shapes to represent the actions in the model.

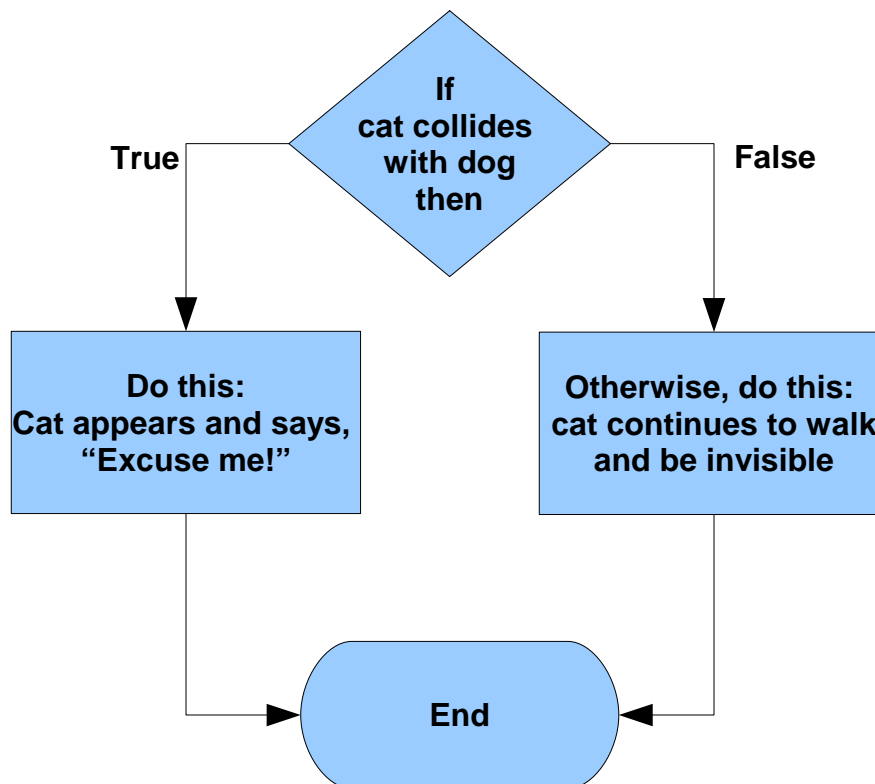
IF Statement Flowchart

If it rains today, then I will wear a rain coat. Otherwise, I will wear a sweater.



IF Statement Flowchart (cont.)

To apply the process flow to Alice 3 consider the following:
If the mysterious, invisible cheshire cat walks into the dog, then the cheshire cat appears and says “Excuse me!”;
otherwise, the cheshire cat continues walking.



 IF Statement Flowchart (cont.)

The important factor to consider when using the IF construct in Alice is that you must specify a *true* or *false* value for the IF construct when you first drag the tile to the code editor. You can choose either argument as a placeholder because you will change the value to be an actual statement.

Consider this portion of the example:

If the mysterious, invisible cheshire cat walks into the dog....

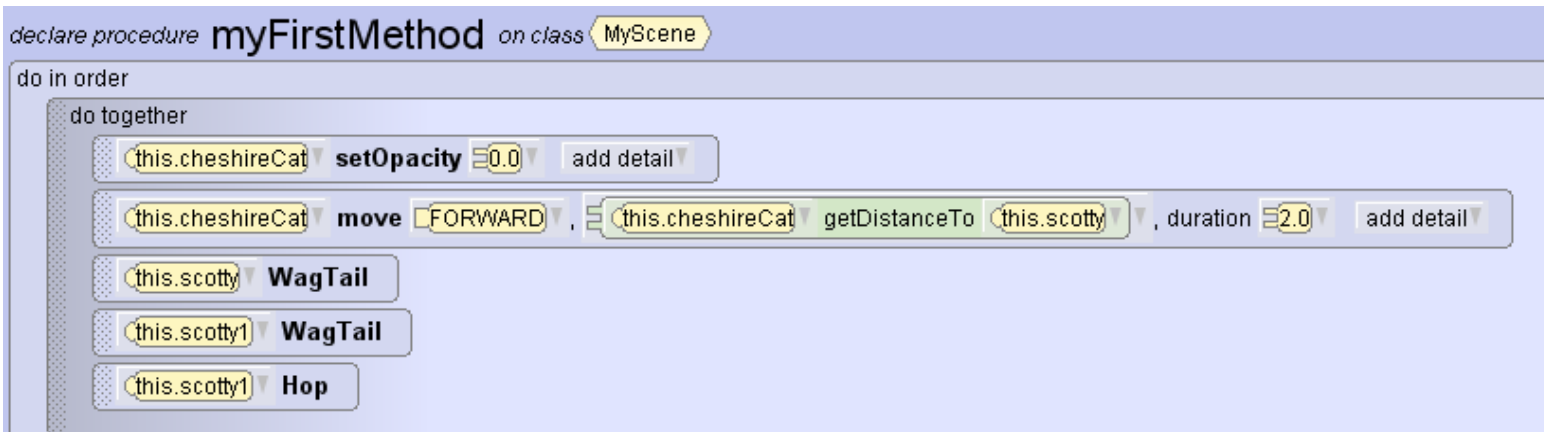
You create an IF construct and choose the true placeholder value. You then replace the true value with a programming statement that determines if the collision between the cat and the dog did in fact occur. If the collision does occur then the IF value is true and the IF statements are executed. Otherwise the Else statements are executed.

Creating an IF Statement

Steps to use the IF statement include:

1. Insert the initial motions that happen before the collision. In this example:
 1. The cat is invisible (set opacity = 0.0).
 2. The cat moves forward the distance to the dog.
 3. The dogs wag their tails and hop.

```
declare procedure myFirstMethod on class MyScene
do in order
do together
  this.cheshireCat setOpacity 0.0 add detail
  this.cheshireCat move FORWARD, this.cheshireCat getDistanceTo this.scotty, duration 2.0 add detail
  this.scotty WagTail
  this.scotty1 WagTail
  this.scotty1 Hop
```



Creating an IF Statement (cont.)

Steps to program IF statement (cont.):

2. Drag and drop the IF statement tile into code editor.
Notice the format of the control statement.
3. Select True condition.



The screenshot displays a Scratch code editor window. At the top, a 'do in order' loop is visible. Inside this loop, there is a 'do together' block containing five actions: 'this.cheshireCat setOpacity 0.0 add detail', 'this.cheshireCat move FORWARD, this.cheshireCat getDistanceTo this.scotty, duration 2.0 add detail', 'this.scotty WagTail', 'this.scotty1 WagTail', and 'this.scotty1 Hop'. Below the 'do in order' loop, an 'if true is true then' conditional statement is present. The 'if true' block is currently selected, and its 'true' condition is highlighted. The 'if true' block contains two empty 'drop statement here' boxes, one for the 'if true' branch and one for the 'else' branch.

Creating an IF Statement (cont.)

Steps to program IF statement (cont.):

4. Select the object to collide (cheshire cat) from Instance menu.
5. In the Functions tab, drag the isCollidingWith function onto the True condition.
6. Select the target object to collide with (dog).

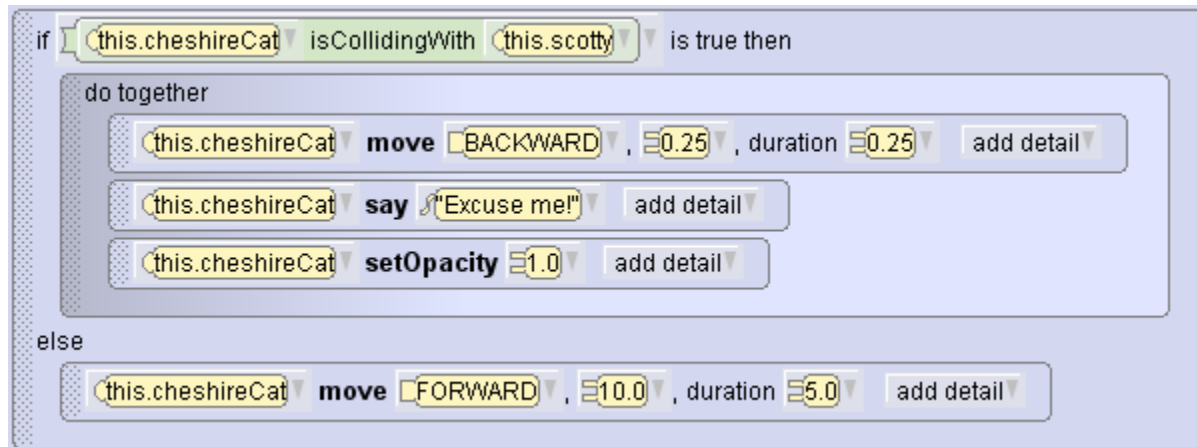


The screenshot displays a visual programming interface with a "do in order" container. Inside, a "do together" block contains several actions: "this.cheshireCat setOpacity 0.0 add detail", "this.cheshireCat move FORWARD, this.cheshireCat getDistanceTo this.scotty, duration 2.0 add detail", "this.scotty WagTail", "this.scotty1 WagTail", and "this.scotty1 Hop". Below this, an "if" statement is shown, with the condition "this.cheshireCat isCollidingWith this.scotty is true then" highlighted in a yellow box. The "if" block has two "drop statement here" slots for the true and false paths.

Creating an IF Statement (cont.)

Steps to program IF statement (cont.):

7. Insert the procedures that will be executed if the condition is true (if) and if the condition is false (else).



```
if [this.cheshireCat] isCollidingWith [this.scotty] is true then
  do together
    [this.cheshireCat] move [BACKWARD], [0.25], duration [0.25]
    [this.cheshireCat] say "Excuse me!"
    [this.cheshireCat] setOpacity [1.0]
  else
    [this.cheshireCat] move [FORWARD], [10.0], duration [5.0]
```

8. Run the animation and debug as necessary.



IF Statement Steps Summarized

To summarize the IF control structure steps:

1. You select the control structure (IF) tile
2. You choose the True conditional option
3. You specify the True condition
4. You specify the actions to occur if the condition is true
5. You specify the actions to occur if the condition is false
6. You execute the program to observe the conditional behavior
7. You debug and test until the conditional actions execute as desired

WHILE Statement

Loops are used when a program requires one or more procedures to execute repeatedly. A loop can be infinite (continue forever) or conditional (stops upon a condition).

Examples

Infinite loop: The hour and minute hands on a clock continue rolling (unless the batteries run out!).



Conditional loop: The propeller of an airplane turns while the airplane is moving or flying. Once the airplane stops, the propeller stops turning.



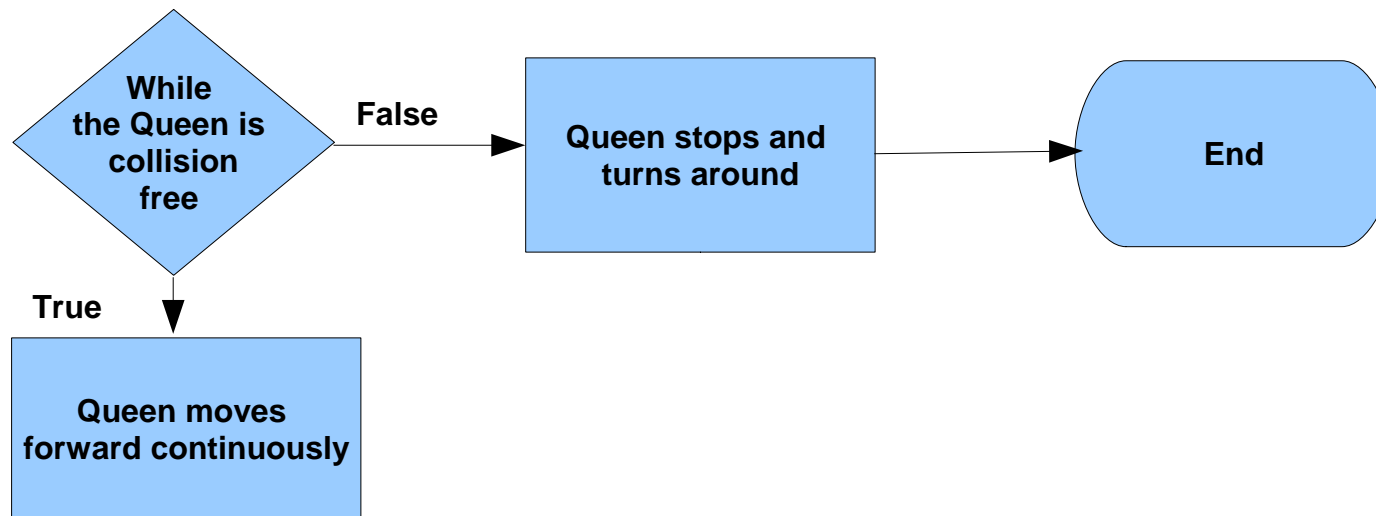


WHILE Statement Flowchart

The While control statement performs conditional loops.

The While control will perform instructions while a condition is true; otherwise it will stop the instructions.

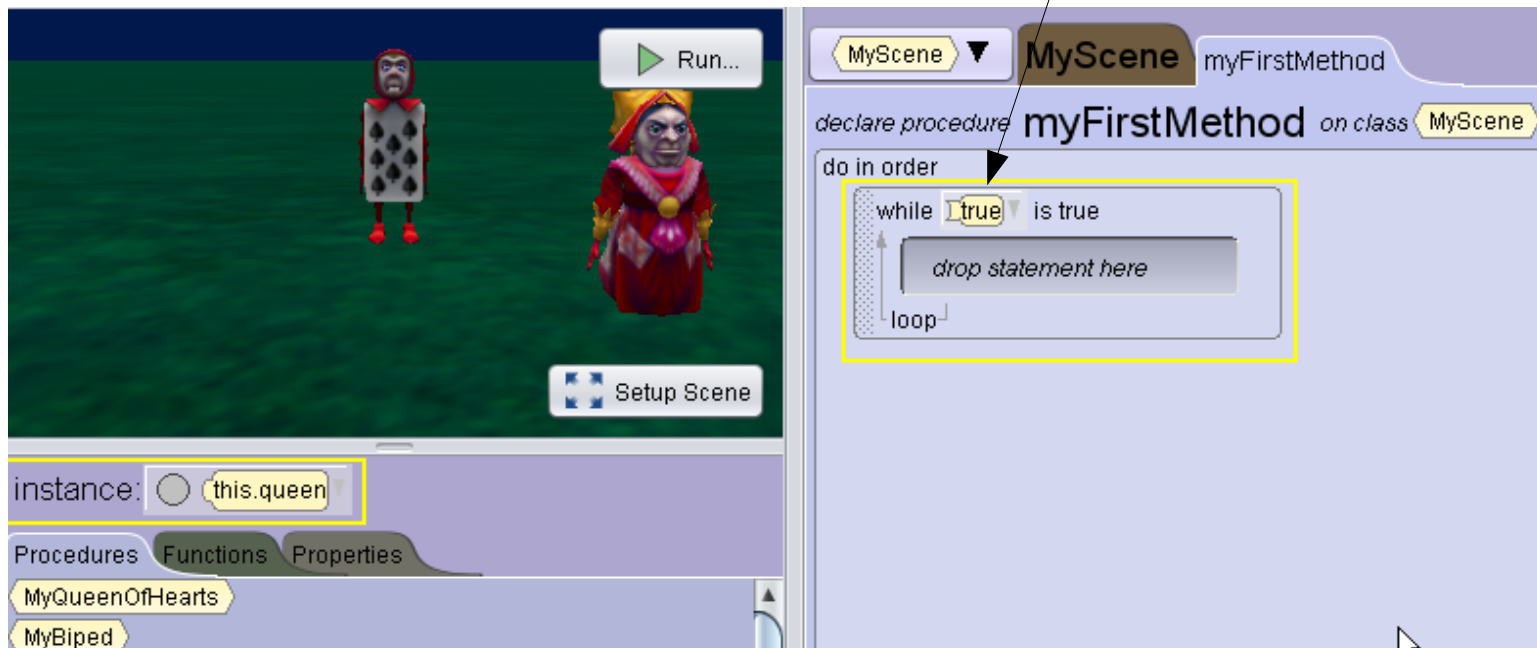
Example: The Queen continues to move forward, unless it collides with the Playing Card. If the Queen collides with the Playing Card, she stops and turns around to face the camera.



WHILE Statement Placeholder

Consider the portion of the While statement that reflects the true condition – while the queen is collision free.

To create the While loop construct using the While tile you must specify an initial true or false placeholder value that is then replaced with the actual condition.



The screenshot displays the Oracle Academy IDE interface. On the left, a 3D scene shows a queen character and a card character. A 'Run...' button is visible. Below the scene, an instance of 'this.queen' is selected. The right pane shows the code editor for 'MyScene.myFirstMethod'. The code is as follows:

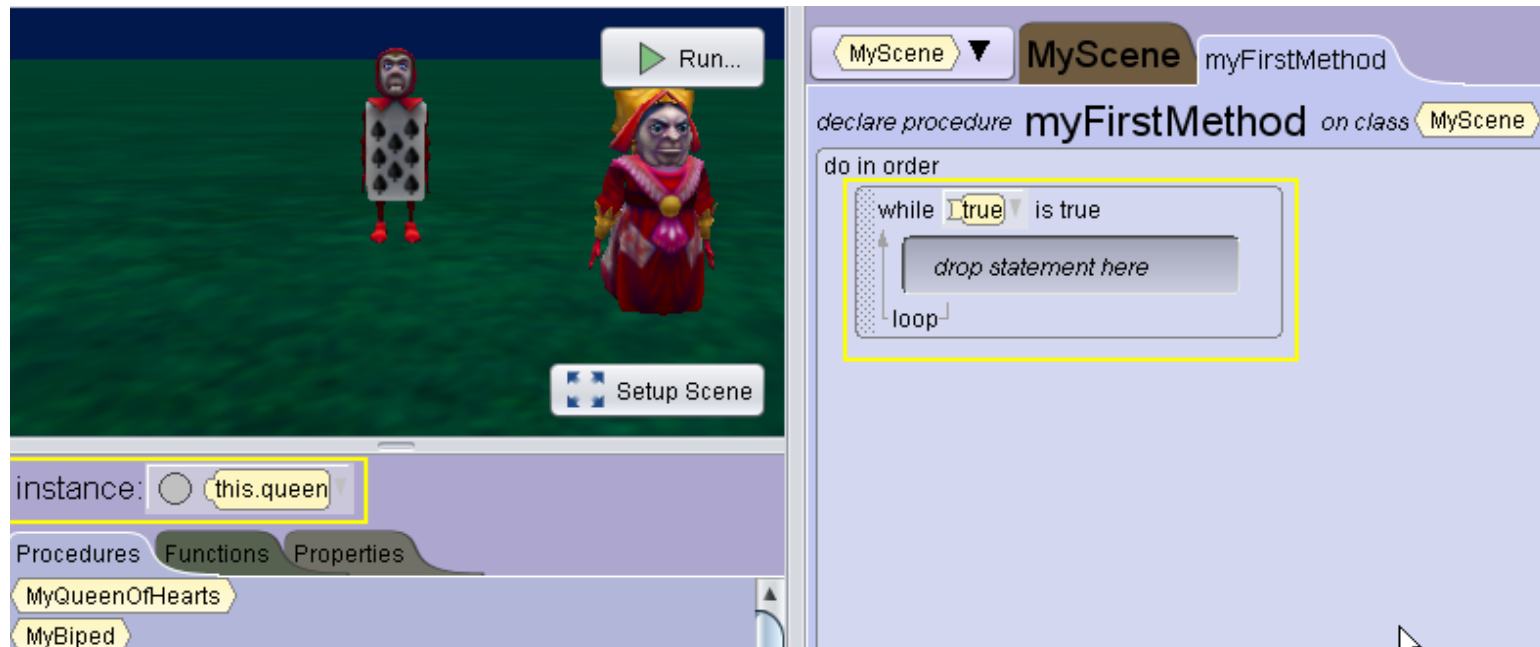
```
declare procedure myFirstMethod on class MyScene
do in order
  while true is true
    drop statement here
  loop
```

A yellow box highlights the 'while true is true' line, and an arrow points from the text above to this line.

WHILE Statement Steps

Steps to program the While control statement:

1. Select the moving object from the Instance menu.
2. Drag the While control statement tile into the code editor. Select the True condition.



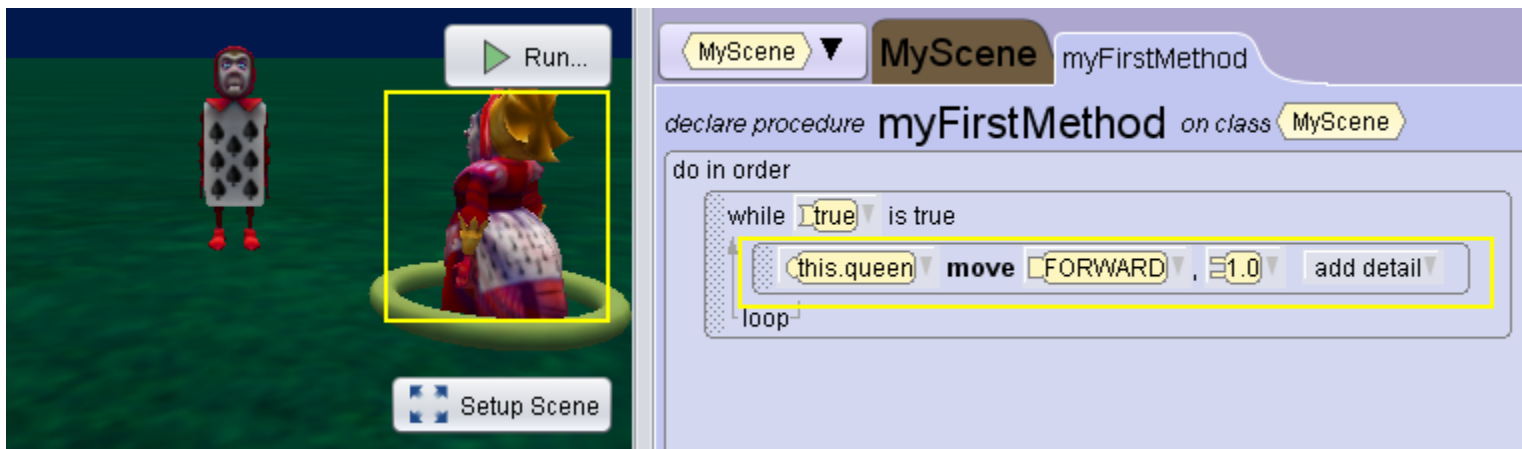
The screenshot displays the Oracle Academy programming environment. On the left, a 3D scene shows two characters: a king and a queen. A 'Run...' button is visible in the top right of the scene, and a 'Setup Scene' button is in the bottom right. Below the scene, the 'instance:' menu is open, showing 'this.queen' selected. The 'Procedures' tab is active, showing 'MyQueenOfHearts' and 'MyBiped'. On the right, the code editor shows the following code:

```
declare procedure myFirstMethod on class MyScene
do in order
  while true is true
    drop statement here
  loop
```

WHILE Statement Steps (cont.)

Steps to use the While statement (cont.):

3. Position the moving object so it is facing the target object to collide with (this will help us test if the while statement works when we run the animation).
4. Drag the move procedure into the While statement. Select forward and 1.0 for the distance.



The screenshot displays the Oracle Academy animation software interface. On the left, a 3D scene is shown with a character and a queen object. A yellow box highlights the queen object. On the right, the code editor for 'MyScene' is shown. The code is:

```
declare procedure myFirstMethod on class MyScene
do in order
while true is true
  this.queen move FORWARD, 1.0 add detail
loop
```

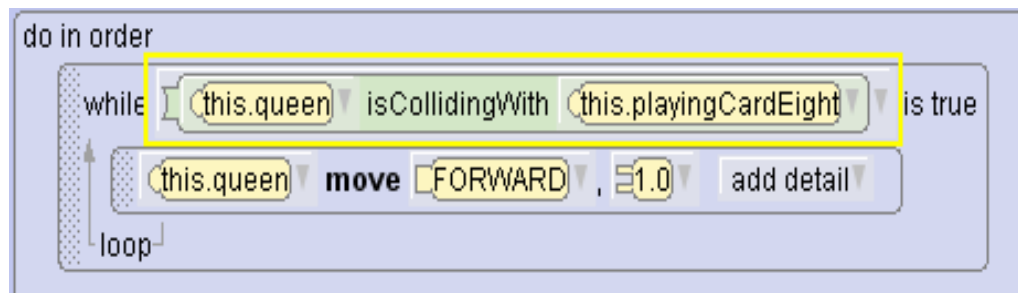
A yellow box highlights the 'while' loop and the 'move' statement.

WHILE Statement Steps (cont.)

Steps to use the While statement (cont.):

5. Drag the `isCollidingWith` function of the queen object onto the true argument and select the target object (`playingCardEight` in this example).

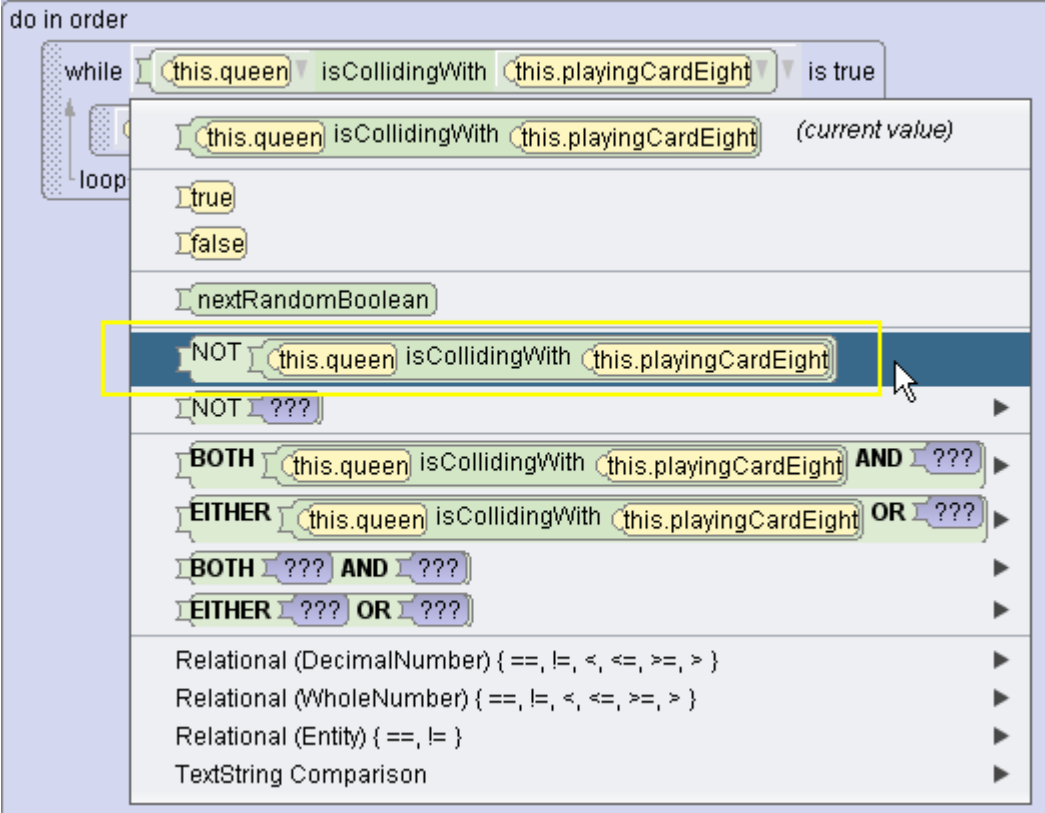
Note that in this example we are specifying the While condition after specifying the action to occur. It is perfectly fine to program either part of the While loop first.



WHILE Statement Steps (cont.)

Steps to use the While statement (cont.):

6. From IsCollidingWith menu, select the NOT isCollidingWith argument to invoke the opposite action.



do in order

while this.queen isCollidingWith this.playingCardEight is true

this.queen isCollidingWith this.playingCardEight (current value)

loop

true

false

nextRandomBoolean

NOT this.queen isCollidingWith this.playingCardEight

NOT ???

BOTH this.queen isCollidingWith this.playingCardEight AND ???

EITHER this.queen isCollidingWith this.playingCardEight OR ???

BOTH ??? AND ???

EITHER ??? OR ???

Relational (DecimalNumber) { ==, !=, <, <=, >=, > }

Relational (WholeNumber) { ==, !=, <, <=, >=, > }

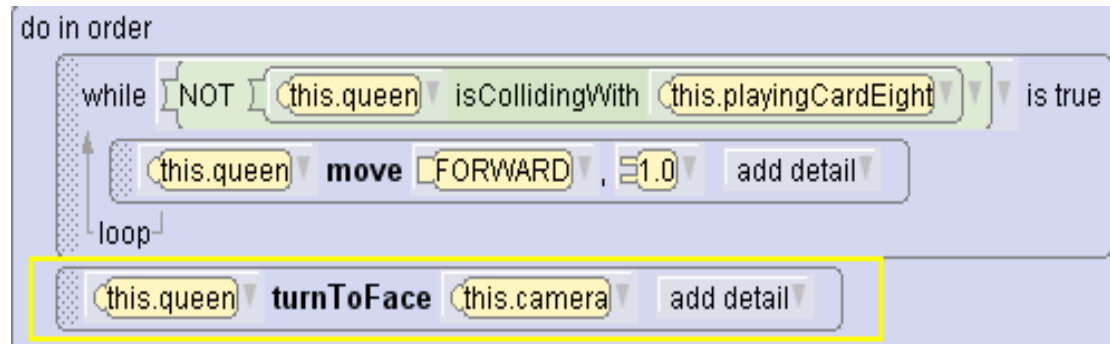
Relational (Entity) { ==, != }

TextString Comparison

WHILE Statement Steps (cont.)

Steps to use the While statement (cont.):

7. Drag the turnToFace procedure below the While condition. Select this.Camera as the target.



WHILE Statement Steps (cont.)

Steps to use the While statement (cont.):

8. To test this program, move the Queen to different positions. Include a path that collides with the playing card, and one that does not. Run the animation and debug as necessary.



Terminology

Key terms used in this lesson included:

Conditional loop

IF control statement

Process flow

WHILE control statement



Summary

In this lesson, you learned how to:

- Use the IF control statement to effect execution of instructions
- Use the WHILE control statement to create a conditional loop for repetitive behavior



Practice

The exercises for this lesson cover the following topics:

- Using conditional control structures
- Using conditional loops for repetition