## **Tutorial 7 – Loops and Object Creation & Destruction**

Estimated time: 90 minutes

### **Prerequisites**

If, else, conditional statements

### **Learning Objectives**

Students will learn about loops, using the while loop as an example. Students will also learn about creating and destroying objects.

### **API Methods Covered**

* createNewObject()
* destroyThisObject()
* destroySingleObject()
* destroyAllObjects()
* setObjectName()
* getRandomNumber()

### **Activity**

1. Describe the while loop
2. Describe the new API methods
3. Create plane, add firewood (Fantasy, then Miscellaneous Category)
4. Add script to firewood (firewood.cs)
   1. Set object name at runtime to: firewoodThrow
   2. Set the object as throwable
5. Create prefab of firewood
   1. Rename firewood object: firewoodThrow
   2. Drag object to the resources folder “\_customPrefabs” to create a prefab
6. Create empty object, add script (createFirewood.cs) that:
   1. Creates several firewoodThrow objects when the left trigger is pressed (at a random locations)
   2. Destroys one of the firewood prefabs when the right trigger is pressed
   3. Destroys all of the firewood prefabs when both of the triggers are pressed

### **Scripts**

**createFirewood.cs**

void updateGame()

{

if (isControllerRightTriggerClicked())

{

int count = 0;

while (count < 5)

{

int xPos = getRandomNumber(10) - 5;

int zPos = getRandomNumber(10);

createNewObject(xPos, 5, zPos, "firewoodThrow");

count++;

}

}

if (isControllerLeftTriggerClicked())

{

destroySingleObject("firewoodThrow");

}

if (isControllerLeftGripDown() && isControllerRightGripDown())

{

destroyAllObjects("firewoodThrow");

}

}

}

**firewood.cs**

void buildGame () {

setObjectThrowable(true);

setObjectName("firewoodThrow");

### }

### **Optional Tutorial 7B**

Estimated Time: 45 minutes

1. Create wall object (Wall category)
2. Create script on object (wall.cs)
   1. Change name to: “collideWall”
   2. Change size of object so that it covers the x-axis of the play area and splits the game area in half.
3. Create Table object (Furniture Category)
4. Add script to table, (table.cs) change name in hierarchy to: “throwTable”
   1. Change name at runtime to: “throwTable”
   2. Make table throwable
   3. When it passes the z-axis location of “collideWall”, destroy the object
5. Create empty object, add script (createTable.cs)
   1. When a controller grip is pressed, create new table at player location
      1. Use timer so that only one object may be created per second

#### ***Source Code***

**collideWall.cs**

void buildGame () {

setObjectName("collideWall");

}

**createTable.cs**

bool timeout = false;

void buildGame()

{

startTimer(1);

}

void updateGame()

{

if (isTimerFinished())

{

timeout = true;

}

if(isControllerGripDown() && timeout){

createNewObject(getPlayerPositionX(), getPlayerPositionY(), getPlayerPositionZ() + 1, "throwTable");

timeout = false;

startTimer(1);

}

}

}

**table.cs**

void buildGame () {

setObjectThrowable(true);

}

void updateGame () {

if (getObjectPositionZ() >= 10)

{

destroyThisObject();

}

}

### **Optional Tutorial 7C**

Estimated Time: 30 minutes

1. Create cactus object (BasicAssets Catagory)
2. Add script (grabObject.cs)
   1. Make object throwable
   2. When the object is grabbed, destroy the object once it is released

#### ***Source Code***

**grabObject.cs**

bool wasGrabbed = false;

void buildGame () {

setObjectThrowable(true);

}

void updateGame () {

if (isObjectGrabbed())

{

wasGrabbed = true;

}

else if (wasGrabbed)

{

destroyThisObject();

}

}