## **Tutorial 8 – Arrays and Object Collisions**

Estimated Time: 90 minutes

### **Prerequisites**

If, else, conditionals, loop

### **Learning Objectives**

Students will learn how to use arrays and how to detect object collisions. Students will also learn how to use the object timer.

### **API Methods Covered**

* hasCollisionWithOtherObject()
* hasCollisionBetweenObjects()
* startObjectTimer()
* isObjectTimerFinished()

### **Activity**

1. Explain Arrays
2. Create a plane, 1 sofa (Furniture Category), and 1 bottle (Fantasy, Misc, On Table Category)
3. Put a script on the sofa (sofa.cs)
   1. Create an array of text strings
   2. Set name of sofa at runtime to: “sofa”
   3. Create a timer
   4. When the timer is finished, move the object to a new position
   5. If the sphere collides with the cube, display new text
4. Put script the rock (bottle.cs)
   1. Make it throwable
   2. Change name to: “bottle”

### **Scripts**

**sofa.cs**

string [] text = {"Way to go", "Good Job", "Awesome", "Cool"};

void buildGame()

{

setObjectName("sofa");

startTimer(5);

}

void updateGame()

{

if (hasCollisionWithOtherObject("bottle")){

int textIndex = getRandomNumber(4);

setObjectText(text[textIndex]);

}

if(isTimerFinished())

{

int newX = getRandomNumber(5);

int newZ = getRandomNumber(7);

setObjectPositionAbsolute(newX, 0, newZ);

startTimer(5);

}

}

**bottle.cs**

void buildGame () {

setObjectName("bottle");

setObjectThrowable(true);

}

### **Optional Tutorial 8B**

Estimated Time: 30 minutes

1. Create a machine object (SciFi, Machines Category)
   1. Add script (machineDestroy.cs)
      1. Change name to: “machine”
2. Create bed object (Furniture Category)
   1. Add script (bed.cs)
      1. Change name to: “bed”
      2. Set as throwable
3. When the bed collides with the machine
   1. Destroy both objects
   2. Create new objects at locations predetermined in arrays held by a script on an empty object (arrayObject.cs)

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

**machine.cs**

void buildGame()

{

setObjectName("machine");

setObjectThrowable(true);

}

void updateGame()

{

if (hasCollisionWithOtherObject("bed"))

{

destroyThisObject();

}

}

**bed.cs**

void buildGame () {

setObjectName("bed");

setObjectThrowable(true);

}

void updateGame () {

if(hasCollisionWithOtherObject("machine"))

{

destroyThisObject();

}

}

**arrayObject.cs**

int[] bedX = { 3, 6, -2 };

int[] bedZ = { 4, 7, 1 };

int[] machineX = { 6, 3, -2 };

int[] machineZ = { 7, 4, 4 };

int locationNumber = 0;

void updateGame()

{

if (hasCollisionBetweenObjects("bed", "machine") && locationNumber < 3)

{

Debug.Log("Collision with bed and machine");

createNewObject(bedX[locationNumber], 1, bedZ[locationNumber], "bed");

createNewObject(machineX[locationNumber], 1, machineZ[locationNumber], "machine");

locationNumber = locationNumber + 1;

}

}

### **Optional Tutorial 8C**

Estimated Time: 30 minutes

1. Create plant object (Vegetation Category) with script plant.cs
   1. Set as throwable
   2. Set name as “plant
2. Create tree object (Vegetation Category with script tree.cs
   1. Set name of tree as “tree”
   2. If A collides with B
      1. Set timer for 3 seconds then destroy tree

### **Source Code**

|  |
| --- |
| plant.cs  **void** **buildGame** () {  setObjectName("plant");  setObjectThrowable(true);  }  tree.cs  **void** **updateGame**() {  **if** (hasCollisionWithOtherObject("plant")){  startTimer(3);  }  **if** (isTimerFinished())  {  destroyThisObject();  }  } |