

Java FX: Keyboard Events & a UFO Game

Learning Goal: Change your program state based on keyboard clicks!

- 1)
 - Copy and paste the file GameAsn.java from our website into a java class called GameAsn
 - Make sure that you have the following images in your source folder:
 - o earth.png
 - o sun.png
 - o space.png
 - o ufo_0.png
 - Run this program – it should look pretty familiar!
- 2) First we are going to create a rectangle that will mirror the movement of our ufo, and a circle that will mirror the movement of our earth (remember that images are difficult to manipulate but shapes are easy).
 - o Section 1 – Add the yellow highlighted code to your project.

```

43      GraphicsContext gc = canvas.getGraphicsContext2D();
44
45      // we need 3 images for this class
46      // remember to move them into the src folder!
47      Image earth = new Image( "earth.png" );
48      Image sun   = new Image( "sun.png" );
49      Image space = new Image( "space.png" );
50      Image ufo   = new Image("ufo_0.png");
51
52      //rectangle to hold place of ufo - starts near top left
53      Rectangle ufoData = new Rectangle(10,10,25,30);
54      Circle earthData = new Circle(10,10,22); //slightly smaller than the earth
55      earthData.setFill(Color.ALICEBLUE);
56
57      //only for testing
58      //root.getChildren().add(earthData);
59
60      final long startNanoTime = System.nanoTime(); //60 times per second

```

- Since these objects are (purposefully) hidden from view, we won't notice them when we run the program. If you want to see the circle, simply uncomment this one line of code:

```

//only for testing
root.getChildren().add(earthData);

```

- 3) Next we are going to go into our AnimationTimer handle and move our circle behind the earth. Some trial and error went into figuring out how far to offset the circle from the earth.

a. Section 3 — Add the green highlighted code to your project.

```

97         // background image clears canvas
98         gc.drawImage( space, 0, 0 ); //draw the background
99         gc.drawImage( earth, x, y ); // draw the earth in a new location each tick
100        gc.drawImage( sun, 196, 196 );
101
102        // move the earthData behind earth
103        //(x and y are the top corner of the image, so we need to move them over
and down
104        // 24 was determined using trial and error
105        earthData.setCenterX(x+24);
106        earthData.setCenterY(y+24);

```

- 4) And now add the actual ufo image and place it directly on top of our ufoData rectangle that we placed in the top left corner earlier.

o Section 4 — Add the purple highlighted code to your project.

```

106        earthData.setCenterY(y+24);
107
108
109        //add the ufo
110        gc.drawImage(ufo, ufoData.getX(), ufoData.getY());
111

```

- Run your program, the ufo should appear in the top left.

- 5) Next we need to check if the ufo and earth are colliding with each other — this is the best part!

o Section 5 — Add the orange highlighted code to your project.

```

110        gc.drawImage(ufo, ufoData.getX(), ufoData.getY());
111
112        //check if the ufo is colliding with the earth
113        if (ufoData.intersects(earthData.getBoundsInLocal())) {
114            System.out.println("COLLISION");
115
116            this.stop();
117
118            Alert alert = new Alert (AlertType.INFORMATION);
119            alert.setTitle("Information Dialog");
120            alert.setHeaderText("Collision!!");
121            alert.setContentText("Aliens have invaded earth");

```

alert.show();

ICS3U JavaFX 07: UFO Game

- Running your program at this time will not display anything, as the ufo and earth are not going to ever collide
- 6) Now we will give the user a chance to move the ufo around the screen using the arrow keys
 - a. Section 2 – Add the pink highlighted code to your project (notice that this code is placed above the AnimationTimer)

```
63     theScene.addEventFilter(KeyEvent.ANY, keyEvent -> {
64
65         //move the ufo right
66         if (keyEvent.getCode()== KeyCode.RIGHT) {
67             ufoData.setX(ufoData.getX()+10);
68         }
69
70         //move the ufo up
71         if (keyEvent.getCode()== KeyCode.UP) {
72             ufoData.setY(ufoData.getY()-10);
73         }
74
75         //move the ufo left
76         if (keyEvent.getCode()== KeyCode.LEFT) {
77             ufoData.setX(ufoData.getX()-10);
78         }
79
80         //move the ufo down
81         if (keyEvent.getCode()== KeyCode.DOWN) {
82             ufoData.setY(ufoData.getY()+10);
83         }
84     });
85
86
87
88     new AnimationTimer()
```

- Play your game!

Programming Exercise:

- Part A: Edit the game to include a different message if the ufo runs into the sun
- Part B: Edit the game so that there are two ufos that start on opposite sides
 - o If either ufo hits the earth, open an alert
 - o If the two ufos collide with each other, open an alert
 - o Bonus: improve this game as much as you want!

Hand this medium project into the dropbox please!

