Programação Orientada a Objetos para Jogos Digitais I 28/08/2022

Malcoln Lucas Minson Rafael RA: 1680962123035 Prof. Labrada

Atividade Framework 2

Objetivo

Criar um simples jogo de um balde que anda de um lado para outro, afim de coletar gotas que caem do céus, com som e sprites dados pelo professor.

Código

```java
package com.astradev.gota;

import java.util.Iterator;

import com.badlogic.gdx.ApplicationAdapter;
import com.badlogic.gdx.Gdx;
import com.badlogic.gdx.Input.Keys;
import com.badlogic.gdx.audio.Music;
import com.badlogic.gdx.audio.Sound;
import com.badlogic.gdx.graphics.GL20;
import com.badlogic.gdx.graphics.OrthographicCamera;
import com.badlogic.gdx.graphics.Texture;
import com.badlogic.gdx.graphics.g2d.SpriteBatch;
import com.badlogic.gdx.math.MathUtils;
import com.badlogic.gdx.math.Rectangle;
import com.badlogic.gdx.math.Vector3;
import com.badlogic.gdx.utils.Array;
import com.badlogic.gdx.utils.TimeUtils;

public class Gota extends ApplicationAdapter {
    private SpriteBatch batch;
    private OrthographicCamera camera;
    private Texture gotaImage;
    private Texture baldeImage;
    private Sound gotaSound;
    private Music chuvaMusic;

    private Rectangle balde;
    private Array<Rectangle> droplets;
    private long instanceLastDroplet;
```

```
33      @Override
34      public void create () {
35          batch = new SpriteBatch();
36          gotaImage = new Texture(Gdx.files.internal("gota.png"));
37          baldeImage = new Texture(Gdx.files.internal("balde.png"));
38
39          gotaSound = Gdx.audio.newSound(Gdx.files.internal("gota.wav"));
40          chuvaMusic = Gdx.audio.newMusic(Gdx.files.internal("chuva.mp3"));
41
42          chuvaMusic.setLooping(true);
43          chuvaMusic.play();
44
45          camera = new OrthographicCamera();
46          camera.setToOrtho(false, 800, 480);
47
48          //Criar Balde
49          balde = new Rectangle();
50          balde.x = 800 / 2 - 64 / 2;
51          balde.y = 20;
52
53          balde.width = 64;
54          balde.height = 64;
55
56          droplets = new Array<Rectangle>();
57          newDroplet();
58      }
59
```

```
60      @Override
61      public void render () {
62
63          tick();
64
65          Gdx.gl.glClearColor(0, 0, 0.2f, 1);
66          Gdx.gl.glClear(GL20.GL_COLOR_BUFFER_BIT);
67
68          camera.update();
69
70          batch.setProjectionMatrix(camera.combined);
71
72          batch.begin();
73          batch.draw(baldeImage, balde.x, balde.y);
74          for(Rectangle droplet: droplets) {
75              batch.draw(gotaImage, droplet.x, droplet.y);
76          }
77          batch.end();
78      }
```

```
80      @Override
81      public void dispose () {
82          baldeImage.dispose();
83          gotaSound.dispose();
84          chuvaMusic.dispose();
85          gotaSound.dispose();
86          batch.dispose();
87
88      }
```

```
90      private void tick() {
91          if(Gdx.input.isTouched()) {
92              Vector3 location = new Vector3();
93              location.set(Gdx.input.getX(), Gdx.input.getY(), 0);
94              camera.unproject(location);
95              balde.x = location.x - 64 / 2;
96          }
97
98          if(Gdx.input.isKeyPressed(Keys.LEFT))
99              balde.x -= 200 * Gdx.graphics.getDeltaTime();
100         if(Gdx.input.isKeyPressed(Keys.RIGHT))
101             balde.x += 200 * Gdx.graphics.getDeltaTime();
102         if(balde.x < 0) balde.x = 0;
103         if(balde.x > 800 - 64) balde.x = 800 - 64;
104
105         if(TimeUtils.nanoTime() - instanceLastDroplet > 1000000000)
106             newDroplet();
107
108         for (Iterator<Rectangle> iter = droplets.iterator(); iter.hasNext();) {
109             Rectangle droplet = iter.next();
110             droplet.y -= 200 * Gdx.graphics.getDeltaTime();
111             if (droplet.y + 64 < 0) iter.remove();
112             if (droplet.overlaps(balde)) {
113                 gotaSound.play();
114                 iter.remove();
115             }
116         }
117
118     }
```

```
120     private void newDroplet() {
121         Rectangle droplet = new Rectangle();
122         droplet.x = MathUtils.random(0, 800-64);
123         droplet.y = 480;
124         droplet.width = 64;
125         droplet.height = 64;
126         droplet.add(droplet);
127         instanceLastDroplet = TimeUtils.nanoTime();
128     }
129 }
```

Conclusão
Aprendizado de itens básicos