COMP20050 Software Engineering Project 2

14. LibGDX details (Part 3)

Assoc. Prof. Pavel Gladyshev

UCD School of Computer Science. **Scoil na Ríomheolaíochta UCD.**

LibGDX support for tile maps

Tile-based video games

The playing area of a tile-based video game is made of small square or hexagonal graphic images called *tiles*, which are laid out in a grid.

The set of all distinct tiles available for use in the play area is called *tileset*.

Being reusable, tiles allow the game developers to quickly build large game levels without the need to draw everything from scratch while minimizing the use of system resources.

In addition to visual appearance of the game level, tile maps usually carry metadata about the location of obstacles, initial positions of monsters, and the player avatar, etc.

Tiled Map XML (TMX) is a popular data format for storing tile map data.

Tiled is an open-source editor for creation and modification of .tmx files.

Tiled Map XML

```
<?xml version="1.0" encoding="UTF-8"?>
<map version="1.2" tiledversion="1.3.2" orientation="orthogonal"
renderorder="right-down" compressionlevel="-1" width="5" height="5"
tilewidth="32" tileheight="32" infinite="0" nextlayerid="4" nextobjectid="2">
 <tileset firstgid="1" name="tileset" tilewidth="32" tileheight="32"
 tilecount="3" columns="3">
  <image source="tileset.png" width="96" height="32"/>
 </tileset>
 <imagelayer id="2" name="Image Layer">
  <image source="background.jpeg" width="160" height="160"/>
 </imagelayer>
 <layer id="1" name="Tile Layer" width="5" height="5">
  <data encoding="csv">
0,0,0,0,0,
0,1,1,1,0,
0,1,2,1,0,
0,1,1,3,0,
0,0,0,0,0
</data>
 </layer>
 <objectgroup id="3" name="Object Layer">
  <object id="1" name="Board" x="32" y="32" width="96" height="96"/>
 </objectgroup>
</map>
```

Image Layer

Tile Layer

Object Layer

Tiled map example code (1)

```
public class MyGame extends ApplicationAdapter {

    final static int HEIGHT = 400;
    final static int WIDTH = 400;

    BitmapFont helvetique;
    SpriteBatch batch;
    TiledMap tiledMap;
    TiledMapImageLayer imageLayer;
    TiledMapTileLayer tileLayer;
    MapLayer objectLayer;
    MapObjects objects;
    float mapWidth;
    float mapHeight;
    MapProperties boardProp;
    Rectangle boardRect;
    OrthogonalTiledMapRenderer renderer;
    OrthographicCamera camera;
```

Tiled map example code (2)

```
@Override
public void create() {
    helvetique = new BitmapFont(Gdx.files.internal("helvetique.fnt"));
    tiledMap = new TmxMapLoader().load("sample_map.tmx");
    imageLayer = (TiledMapImageLayer) tiledMap.getLayers().get("Image Layer"); // .get(0);
    tileLayer = (TiledMapTileLayer) tiledMap.getLayers().get("Tile Layer"); // .get(1);
    objectLayer = tiledMap.getLayers().get("Object Layer"); // .get(2);
    objects = objectLayer.getObjects();
    mapHeight = imageLayer.getTextureRegion().getRegionHeight();
    mapWidth = imageLayer.getTextureRegion().getRegionWidth();
    boardProp = objects.get("Board").getProperties();
    boardRect = new Rectangle(
            (float) boardProp.get("x"),
            (float) boardProp.get("y"),
            (float) boardProp.get("width"),
            (float) boardProp.get("height")
    );
    renderer = new OrthogonalTiledMapRenderer(tiledMap);
    batch = new SpriteBatch();
    camera = new OrthographicCamera();
}
```

Tiled map example code (3)

```
@Override
public void render() {
    ScreenUtils.clear(1.0f, 1.0f, 1.0f, 1.0f);

    camera.setToOrtho(false,Gdx.graphics.getWidth(),Gdx.graphics.getHeight());
    camera.translate(
            -(Gdx.graphics.getWidth()-mapWidth)*0.5f,
            -(Gdx.graphics.getHeight()-mapHeight)*0.5f
    );
    camera.update();
    renderer.setView(camera);
    renderer.render();
    batch.setProjectionMatrix(camera.combined);
    batch.begin();
    helvetique.draw(batch,"Hello :)",0,0);
    batch.end();
}

@Override
    public void dispose() {
        batch.dispose();
        helvetique.dispose();
        tiledMap.dispose();
    }
}
```

Tiled map example

Hello :)