RPG, part 1

Starting Point

The RPG repository.

Make a fork of the repository you will push your work in. This is NECESSARY for completing the project (homework in a future class).

Steps

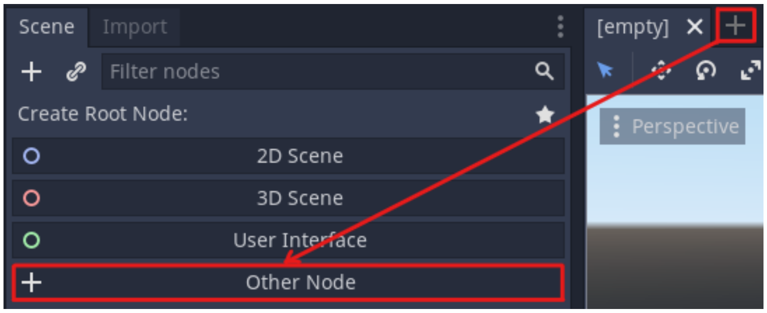
Examine the solution and read through the scripts so you have an understanding how the movement works. Especially pay attention to the collision detection code.

Camera following the player

The camera should follow the player (i.e. have the same position). It could be done either by making the camera node a child of the player node, or by attaching a script which periodically sets the position (which is more flexible, makes further modifications to the camera behavior easier).

Task 1 - Camera, with a script such that it follows player.
Task 1.1 - Give an export property to the script, such that you can select in Inspector which object to follow.

Relevant resources:
NodePath — Godot Engine (stable) documentation in English
Node — Godot Engine (stable) documentation in English
Camera2D — Godot Engine (stable) documentation in English



To create a new scene, click the + icon, then select “Other Node” and search for the desired root node. Don’t forget to save the scene.

TileMap Collisions

Some of the objects should be colliding in the world, so that the player cannot walk through e.g. trees and the house.

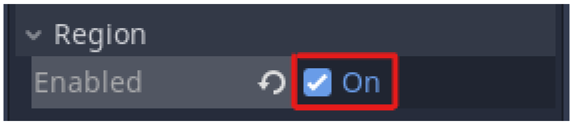
The tilemaps in Godot 4 became quite powerful, and you can set various attributes to each tile, such as a collision shape. A simple full-square collider should work for us.

Task 2 - Open the WorldTileset resource and set colliders on everything you don't want the player to be able to walk through.

Relevant resources:
[Using TileMaps — Godot Engine (stable) documentation in English]
[Using TileSets — Godot Engine (stable) documentation in English]

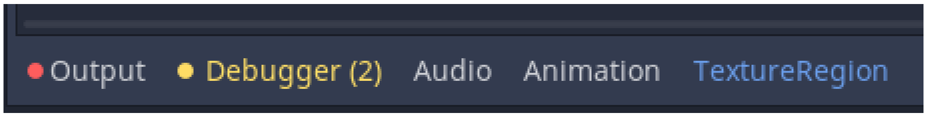
Collectible coins

Coins are placed in the world, but aren't collectible. Whenever the player overlaps with the coin, it should be collected (the player's state is updated and the coin disappears).

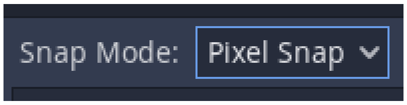


Region
Enabled ☑ On

When selecting a sprite from a spritesheet, set the region of it's texture.



Enable pixel snapping for easy selection.



Snap Mode: Pixel Snap

Task 3 - Create a collectible coin. Store the amount of coins in the PlayerState class.

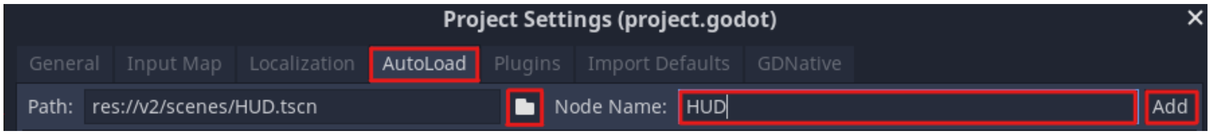
Relevant resources:
[Area2D — Godot Engine (stable) documentation in English]
[Using Area2D — Godot Engine (stable) documentation in English]
[Node — Godot Engine (stable) documentation in English] (`is_in_group()` method)

GUI for coins count

The number of collected coins should be displayed in the HUD.

scenes/HUD.tscn - A scene with the GUI is already prepared. It just needs to be added to AutoLoad.

Task 4 - Create a script for the HUD. Add methods for updating health and coins total. It shall be accessible via autoload (add just the scene, not the script). Have it update whenever a coin is collected.



Path: res://v2/scenes/HUD.tscn

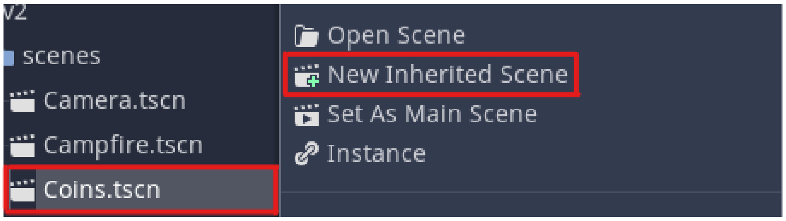
Node Name: HUD

Relevant resources:
[Singletons (AutoLoad) — Godot Engine (stable) documentation in English](#)
[Autoloads/ Singletons :: Godot Recipes (kidscancode.org)](#)
[Label — Godot Engine (stable) documentation in English](#)

Coins of different value

There should be different variants of coins with different sprites and different values.

Task 5 - Create a new scene CoinsLarge with a different sprite, make its class inherit from Coins, and add a different amount of coins.



To create an inherited scene, right click the base scene in the FileSystem panel and select “New Inherited Scene”.

Relevant resources:
[Inheritance :: Godot Recipes (kidscancode.org)](#)

Useful resources

[Step by step — Godot Engine (stable) documentation in English](#)
[GDScript — Godot Engine (stable) documentation in English](#)
[Applying object-oriented principles in Godot — Godot Engine (stable) documentation in English](#)

Assets sources

Kenney • Roguelike/RPG pack
Kenney • Roguelike Characters
Kenney • Tappy Plane