In this guide we will be experimenting with game feel using a 3rd person character with modifiable properties for speed, jump, etc.

# Step 1 - Importing the Template Project

A screenshot of a computer

Description automatically generated

You can ignore a pop-up window asking to open the Asset Library as you have no projects.

Click ***+*** *Import* in the top left. Navigate to the *GameFeelTemplate* folder and select the *project.godot* file.

A screenshot of a computer program

Description automatically generatedA screenshot of a computer program

Description automatically generated

Name the project in *Project Name* and navigate to the folder you want to save the project in.

Click *Create Folder* to add an empty folder in the *Project Path* folder with the name of your project. The project will be placed in this created folder.

# Step 2 - Importing Assets

Start by dragging the *Environment* folder from the Windows File Explorer into the *FileSystem* window in Godot. In the *Base* folder there are many models which can be used to build out a level.

# Step 2 - Setting up Input Map

A screenshot of a computer

Description automatically generated

Under *Project* select *Project Settings* and open the *Input Map* tab.

To add new input actions, enter the name in the *Add New Action* text box and click *Add*.

A screenshot of a computer

Description automatically generated

# Step 3 - Creating the Level Scene

A screenshot of a computer

Description automatically generated

In the *Scene* panel, select *3D Scene*. This will be the root node of the level scene.

A screenshot of a computer

Description automatically generated

Double-click on the created node in the *Scene* panel to rename it, e.g. to *Level* and save it with the same name.

A screenshot of a computer

Description automatically generated

Click the **+** in the *Scene* tab to add a *StaticBody3D*, with a *CollisionShape3D* and *MeshInstance3D* as child nodes. Double-click on the root *Node3D* node to rename it, e.g. to *Level*.

A screenshot of a computer program

Description automatically generated

While selecting the *CollisionShape3D* in the *Scene* hierarchy, in the *Inspector* add a *BoxShape3D* to the *Shape* property of the *CollisionShape3D*. Set its *Size* to be wider, e.g. 10, 0.5, 10. This will be the shape of the level floor which the player will collide with.

A screenshot of a computer

Description automatically generated

Add a *BoxMesh* to the *MeshInstance3D* with the same *Size* as the collision *BoxShape3D*. This will be the visible part of the floor which the player can see.

A screenshot of a computer

Description automatically generated

Click the three vertical dots along the panel above the viewport to add a default *DirectionalLight3D*, and a default *WorldEnvironment*.

We now have a basic level to place a player in. Drag and drop the *Player.tscn* from the *FileSystem* into the *Scene* tab or the *Viewport* of the Level scene.

# Step 3 - Creating the Player Scene

To create the player scene, first open a new tab.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

In the *Scene* panel, select *Other Node* and search for *CharacterBody3D*.

A screenshot of a computer

Description automatically generated

Click the **+** in the *Scene* tab to add *CollisionShape3D* and *MeshInstance3D* nodes just like with the level scene. Double-click on a node to rename it, e.g. *CharacterBody3D* to *Player*.

A screenshot of a computer

Description automatically generated

Selecting the *CollisionShape3D*, in the Inspector add a *CapsuleShape* to the *Shape* property.

Drag the *sophia\_skin.tscn* we added earlier from the FileSystem into the Scene.

A computer screen shot of a white cylinder

Description automatically generated

You should now have a *Player* scene like this.

A screenshot of a computer

Description automatically generated

Next select the Player root node (*CharacterBody3D*) and in the *Inspector* click where it says *<empty>* in the *Script* property.

Select *Quick Load* and select the *Player.gd* that was included in the assets.

Add a *Node3D* and name it CameraPivotHorizontal, and add a child Node3D named CameraPivotVertical. To this node, add a child of type *SpringArm3D* (by searching in the list of node types). These nodes handle pivoting around the camera and the distance from the camera. As a child of the *SpringArm3D*, finally add a *Camera3D* node.

A screenshot of a computer

Description automatically generated