Unity NPC and Skill Manager

User Manual

Table of Contents

User Manual..... 1

Getting Started..... 1

Creating NPCs..... 6

Creating Skills..... 16

Leveling Settings..... 22

Save System..... 24

Documentation..... 26

Classes, Structs and Enums..... 26

NPCBase..... 26

Relationship..... 26

Skill Manager..... 27

SkillInformation..... 27

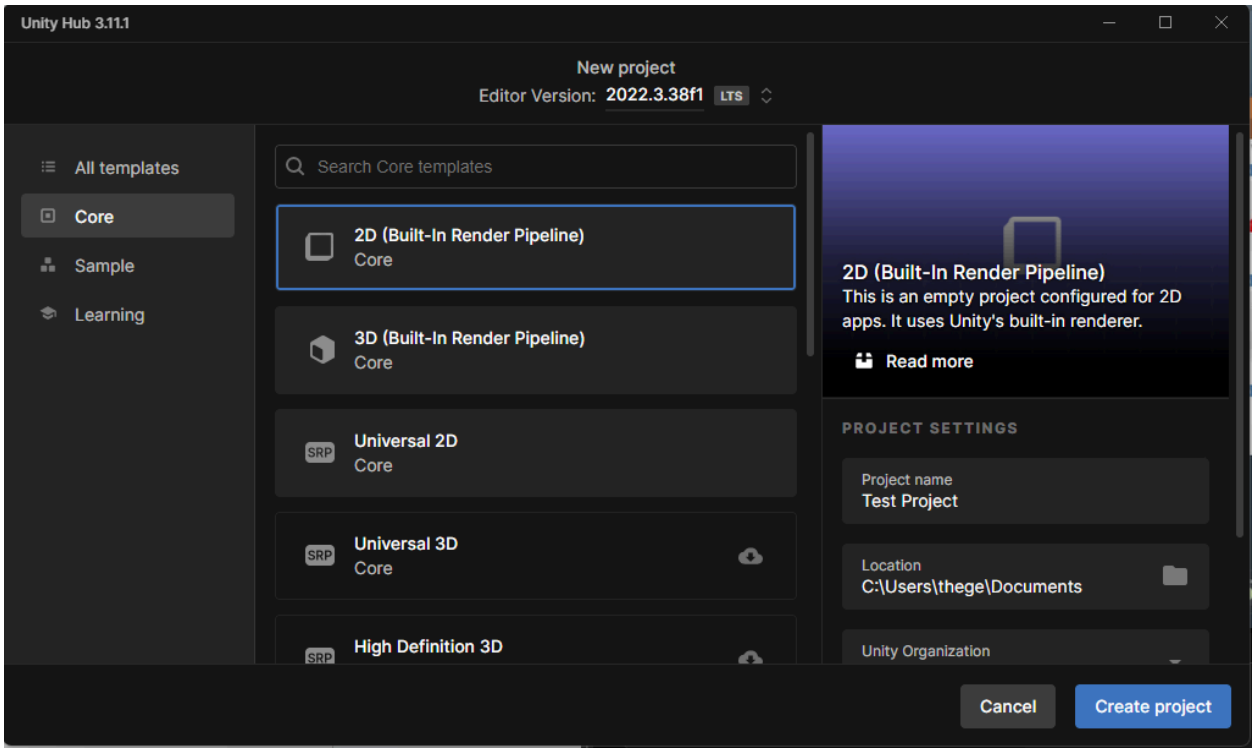
Sentiments..... 28

RelationshipInformation..... 28

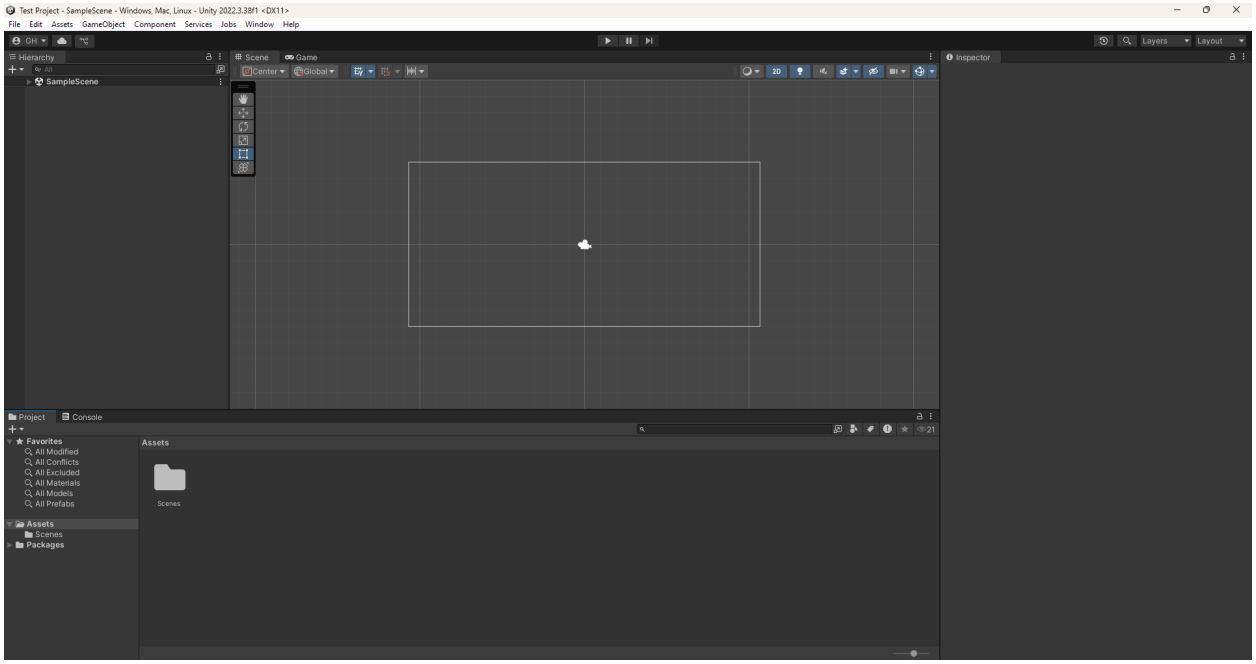
Getting Started

This section will show you how to **set up** the package in your Unity Project.

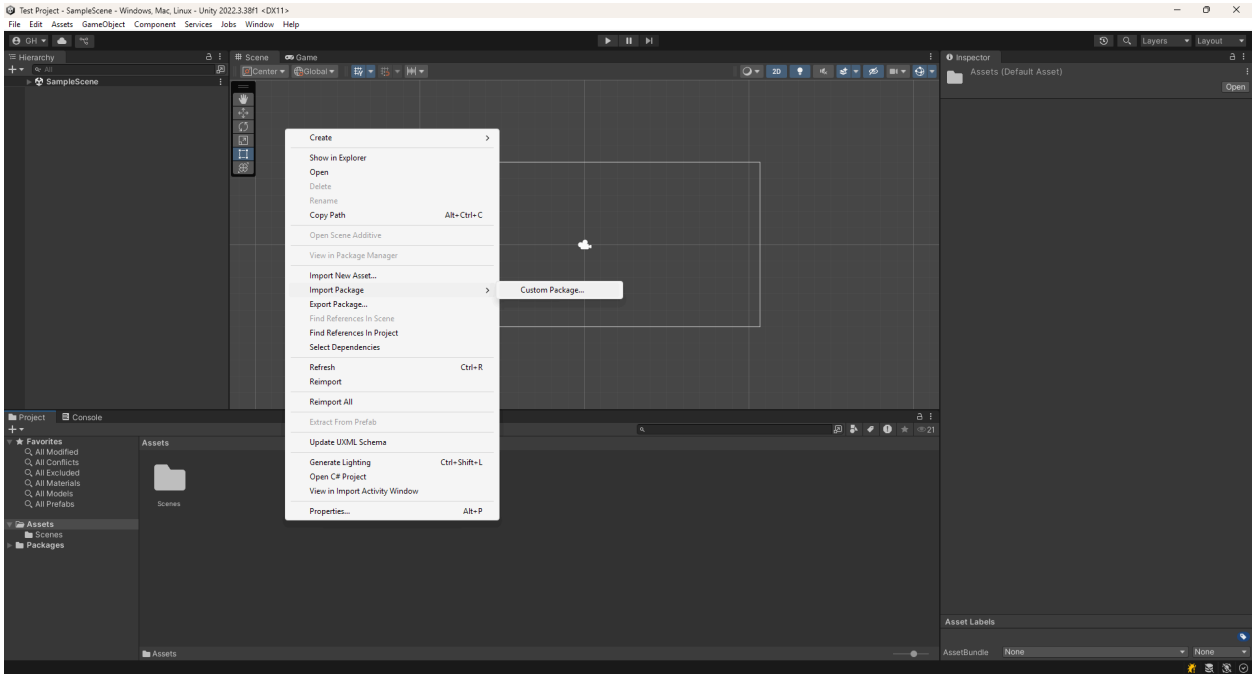
Create or open your Unity project.



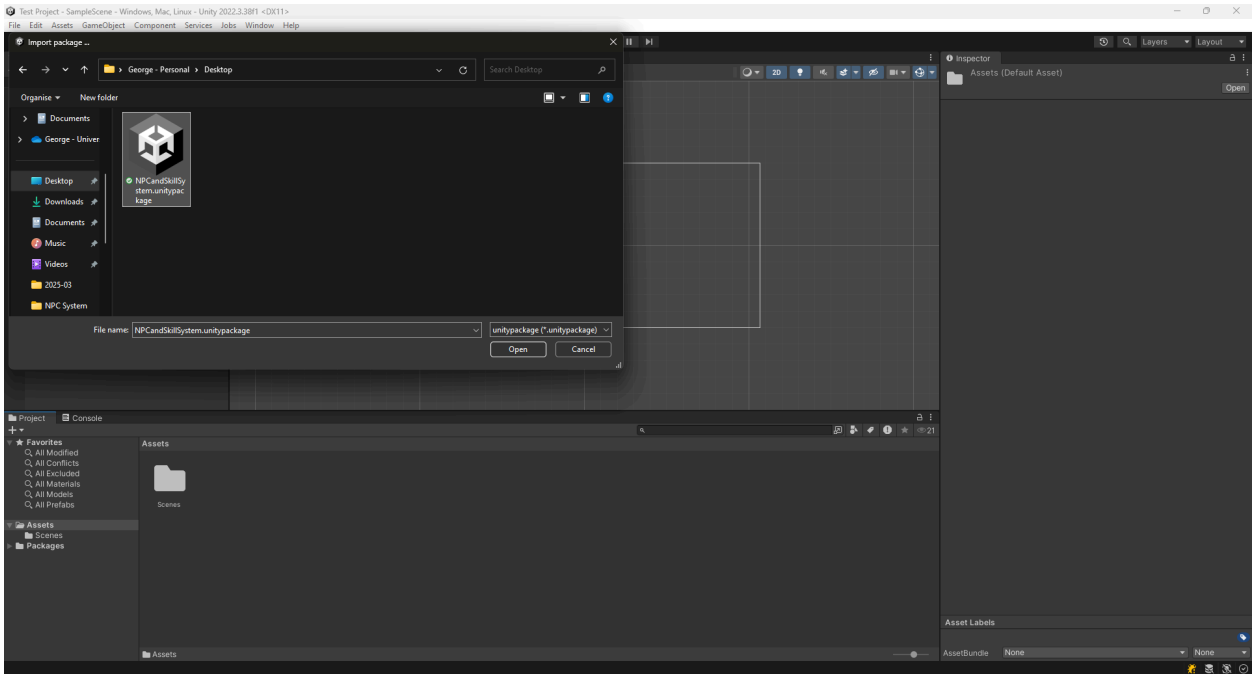
Navigate to the root Assets folder in the Project Window.



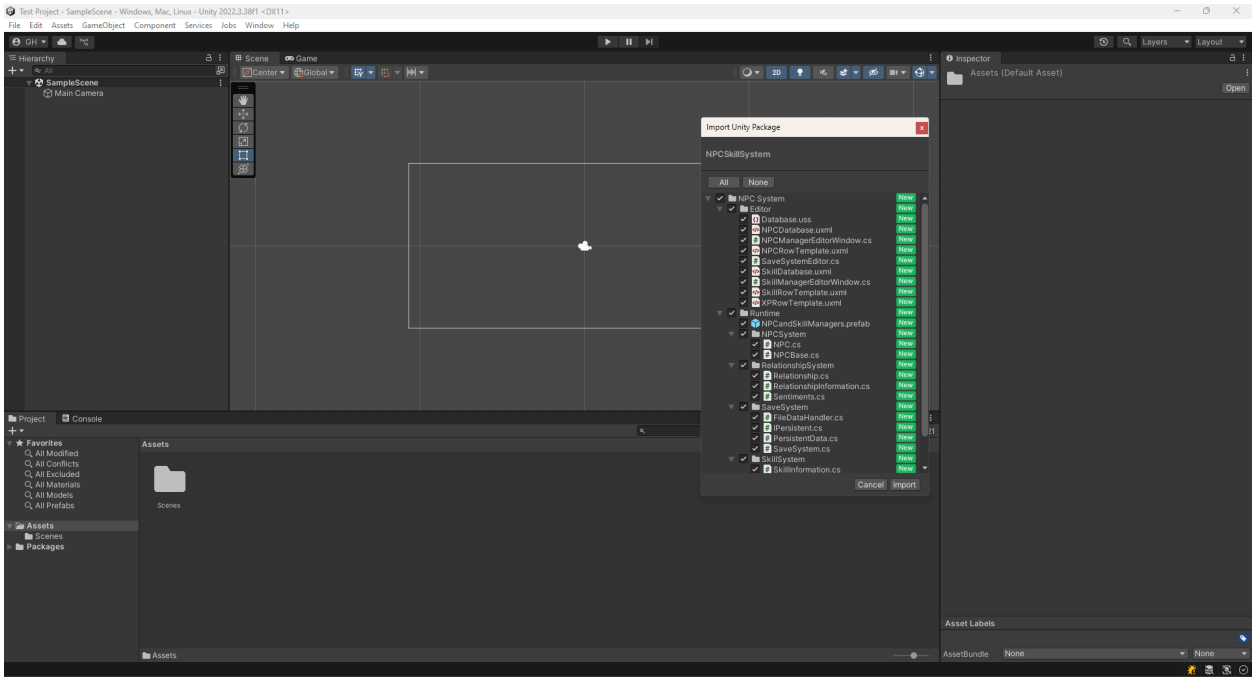
Right-click in the space and select Import Package, Custom Package.



Select the NPCandSkillSystem.package file and click Open.

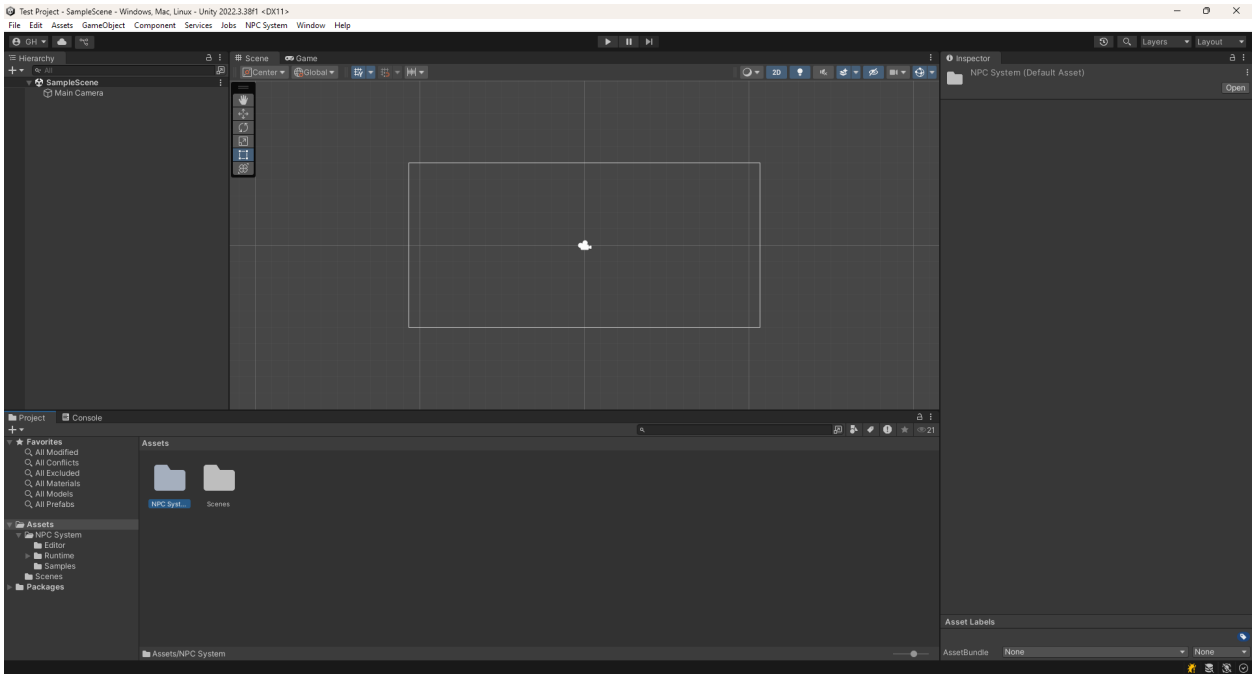


On the pop-up menu, leave All selected and click Import.

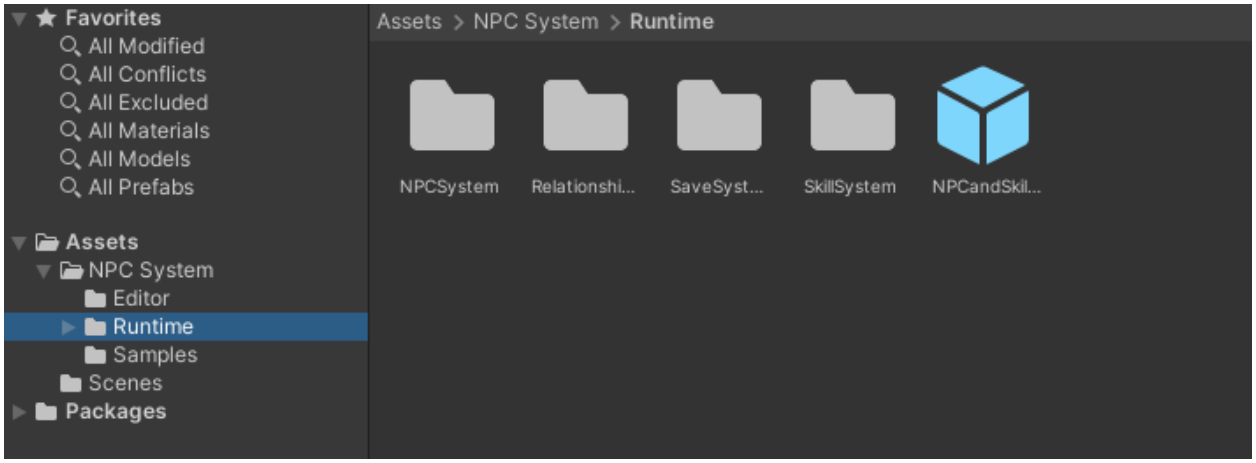


Note: Do not move or rename assets from their existing locations.

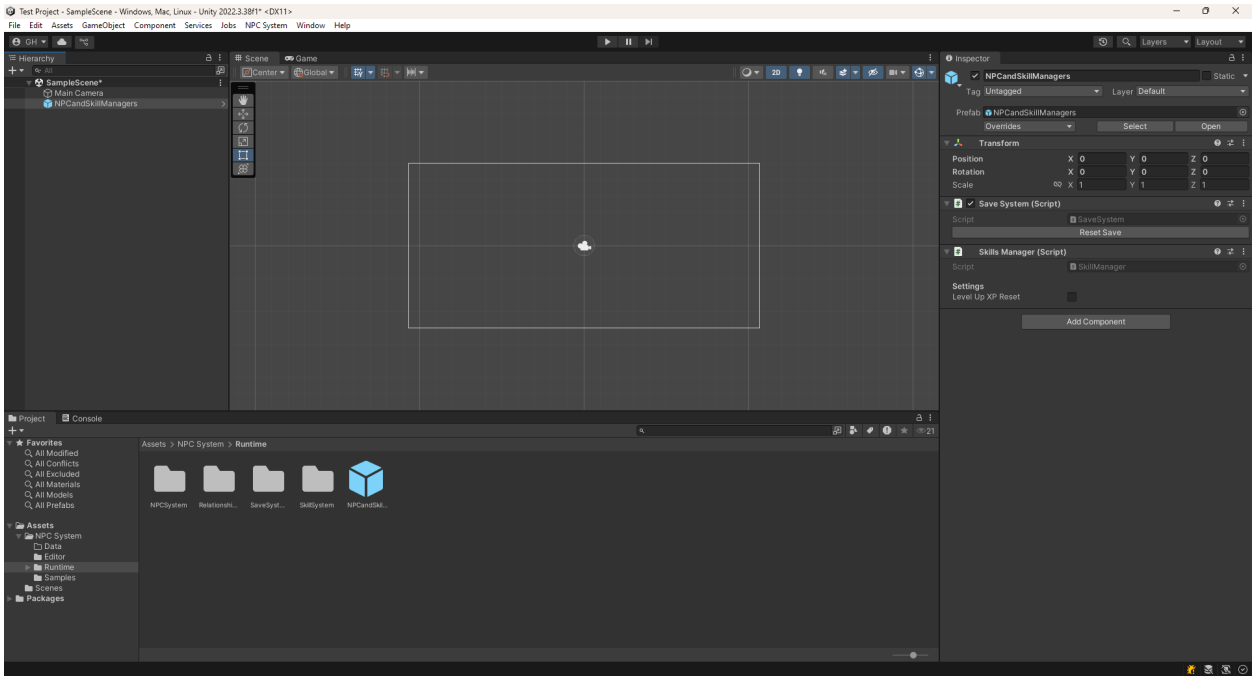
Finally, you'll need to place the NPCandSkillManager Prefab into the scene in order to handle saving and data persistence.



Go to Assets, NPC System, Runtime.



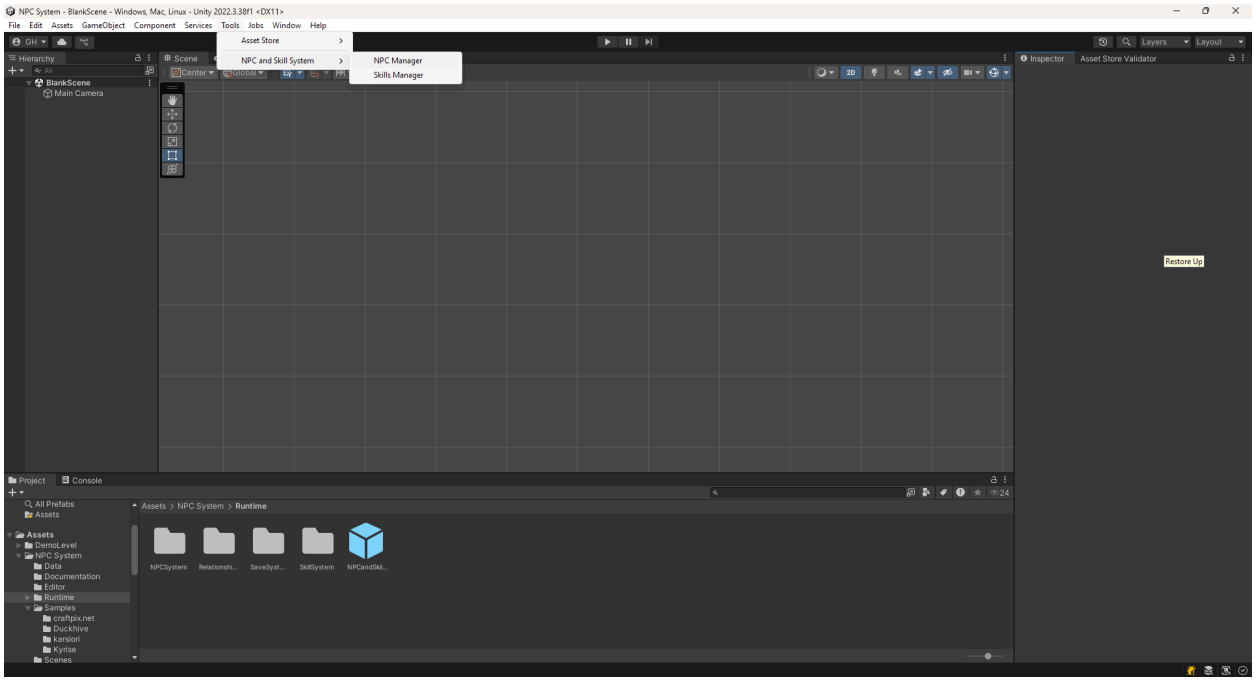
Drag in the NPCandSkillManager Prefab into the Scene Hierarchy.



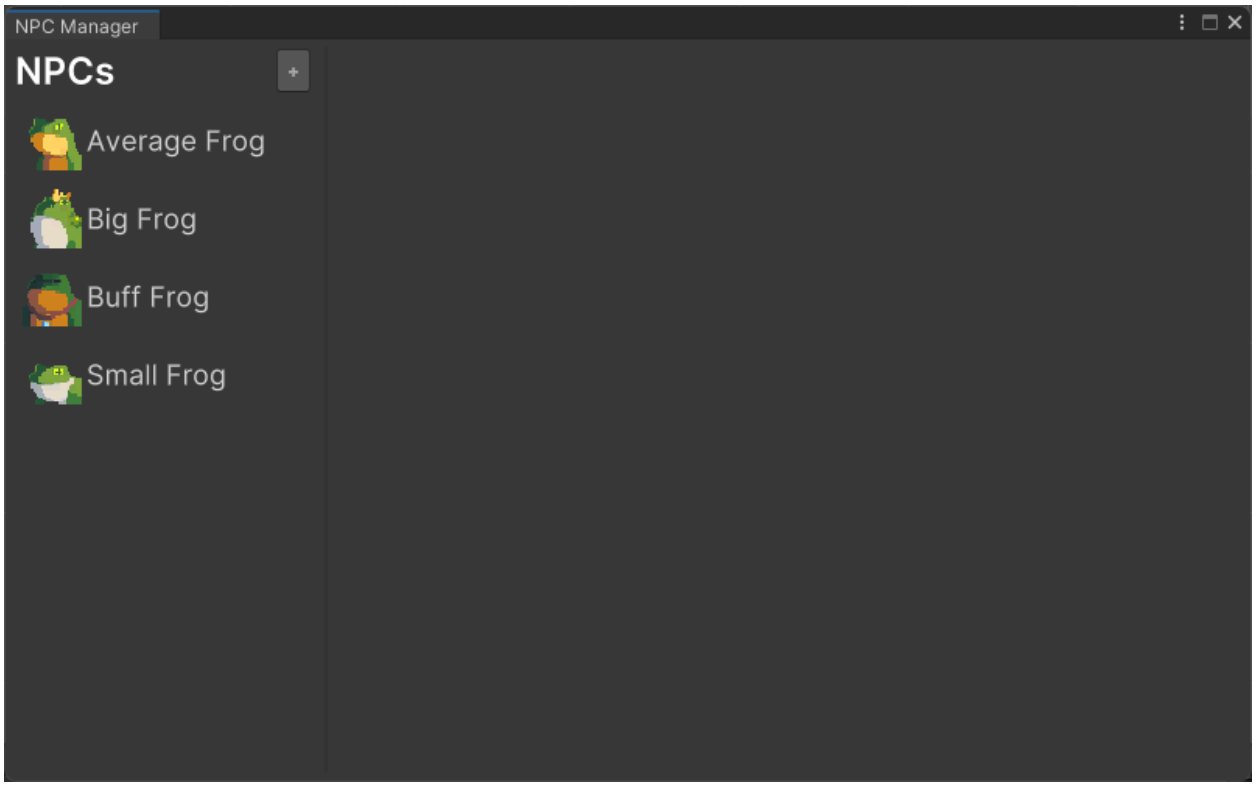
Creating NPCs

This section will show you how to create and manage NPCs.
To start **creating** NPCs, you'll have to open the NPC Manager Window.

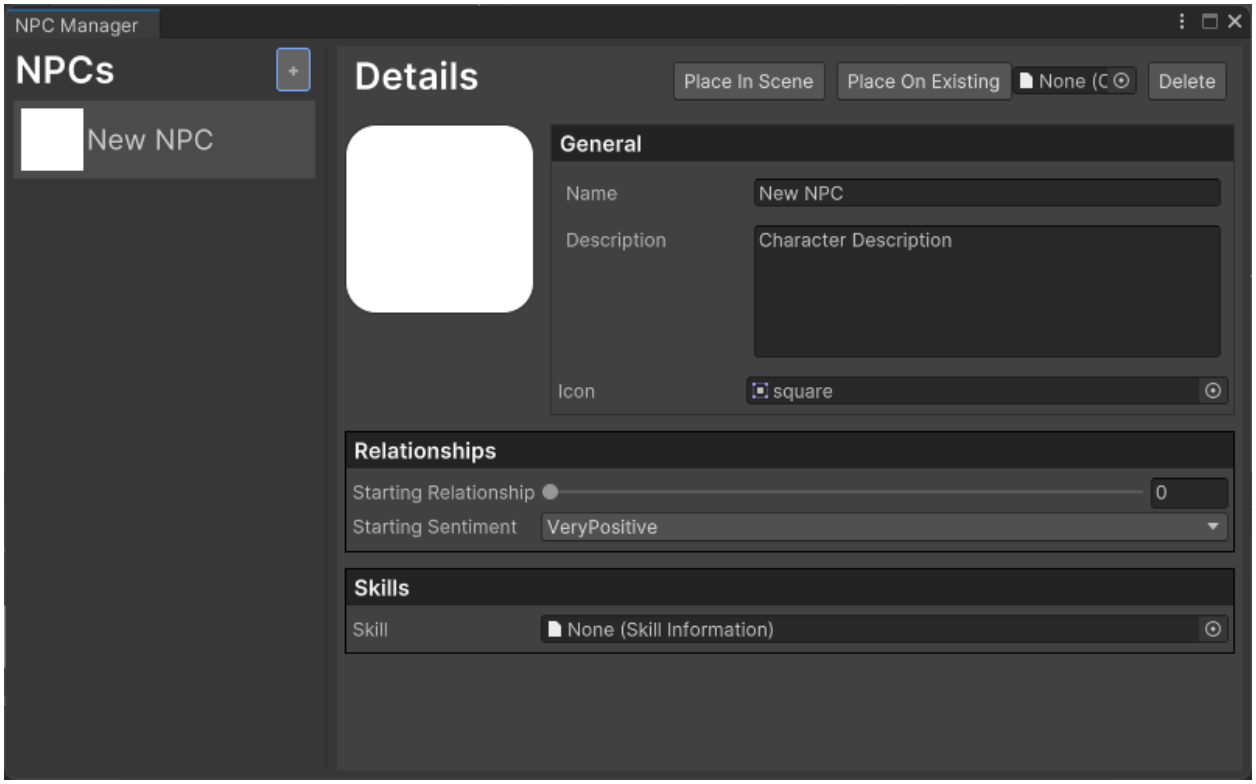
On the toolbar at the top of the Editor, go to Tools, NPC and Tool System, NPC Manager.



This window will appear. There's some examples to get you started, but you can delete them anytime. (You'll learn how to do that below)

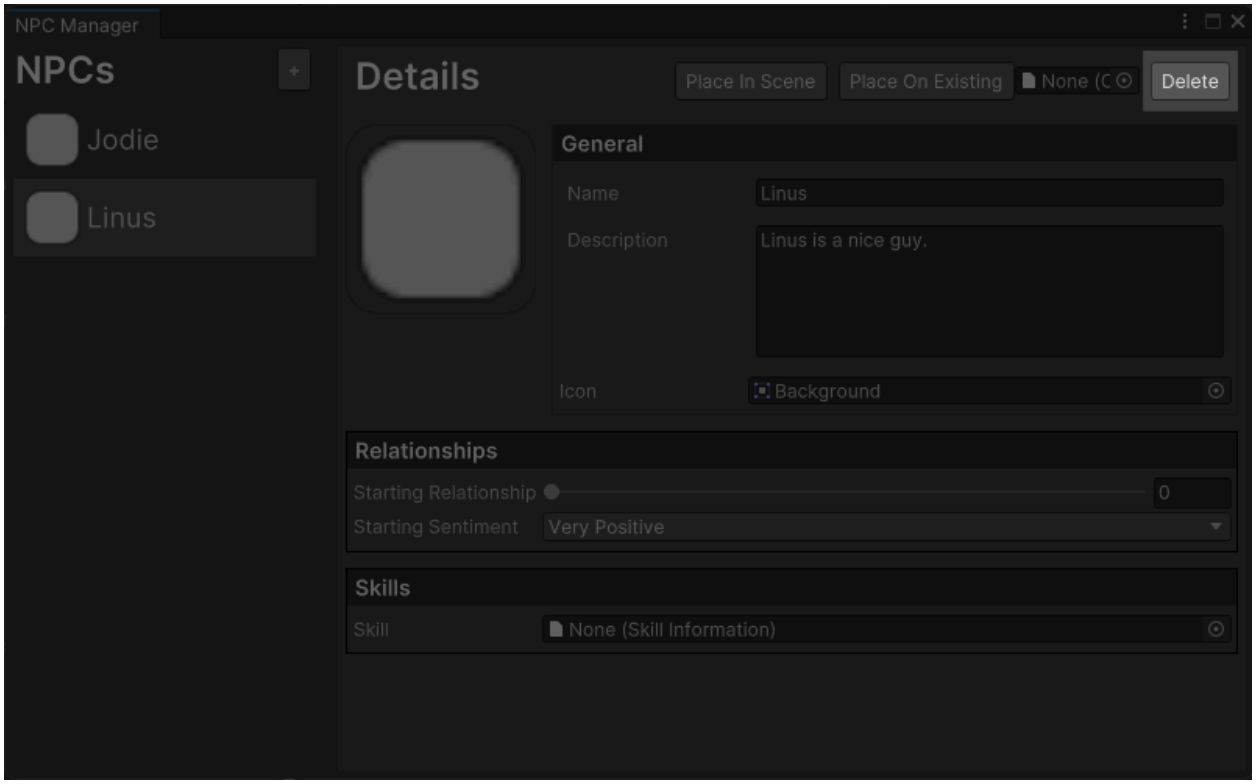


Click the + button to add a new NPC. The new NPC will appear in the NPC's List on the left, and its details appear in the Details section, on the right.

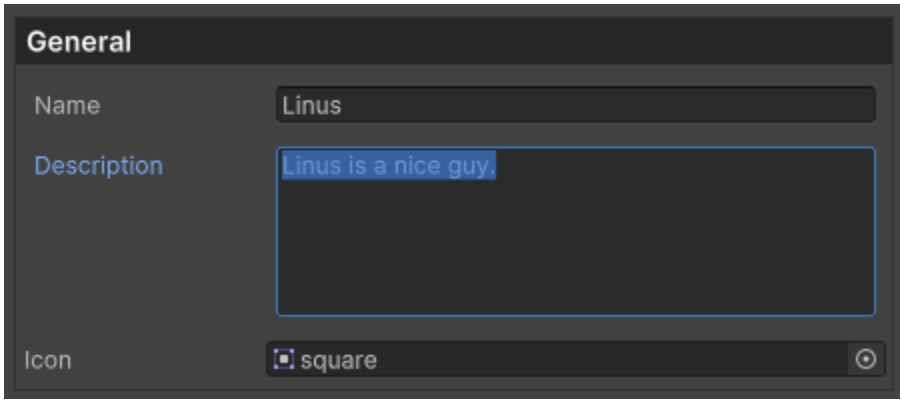


Note: The Details section will always display the information for the currently NPC form the NPC's List.

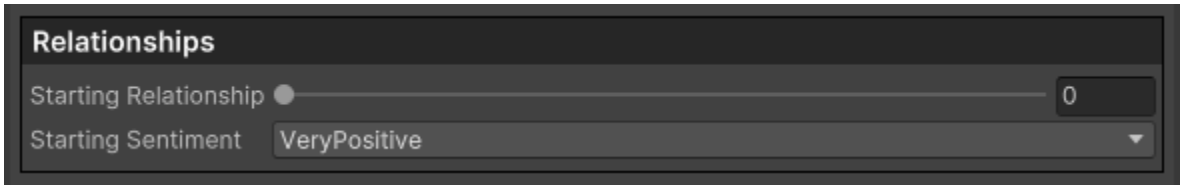
At the top right is the Delete button. Press this to delete the currently selected NPCs. Check before deleting as you cannot recover deleted NPCs.



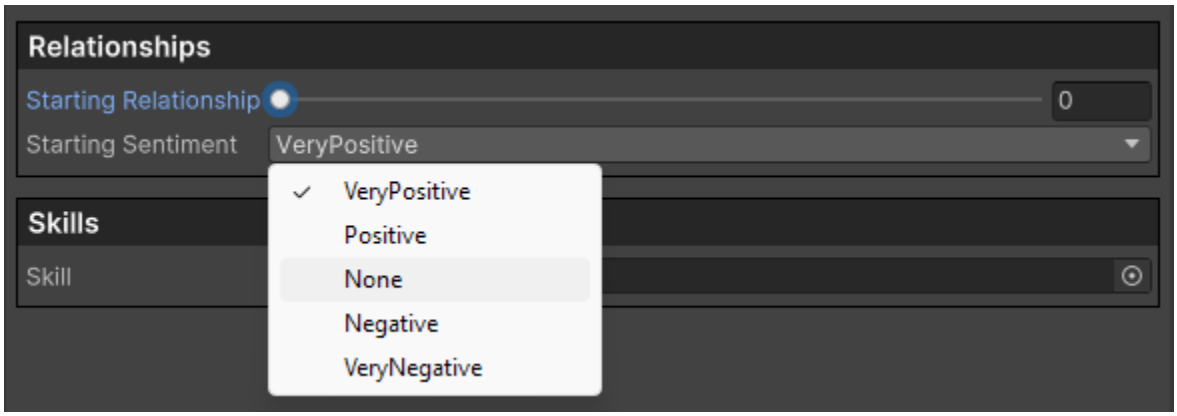
Under General, set the NPC's name, description and icon.



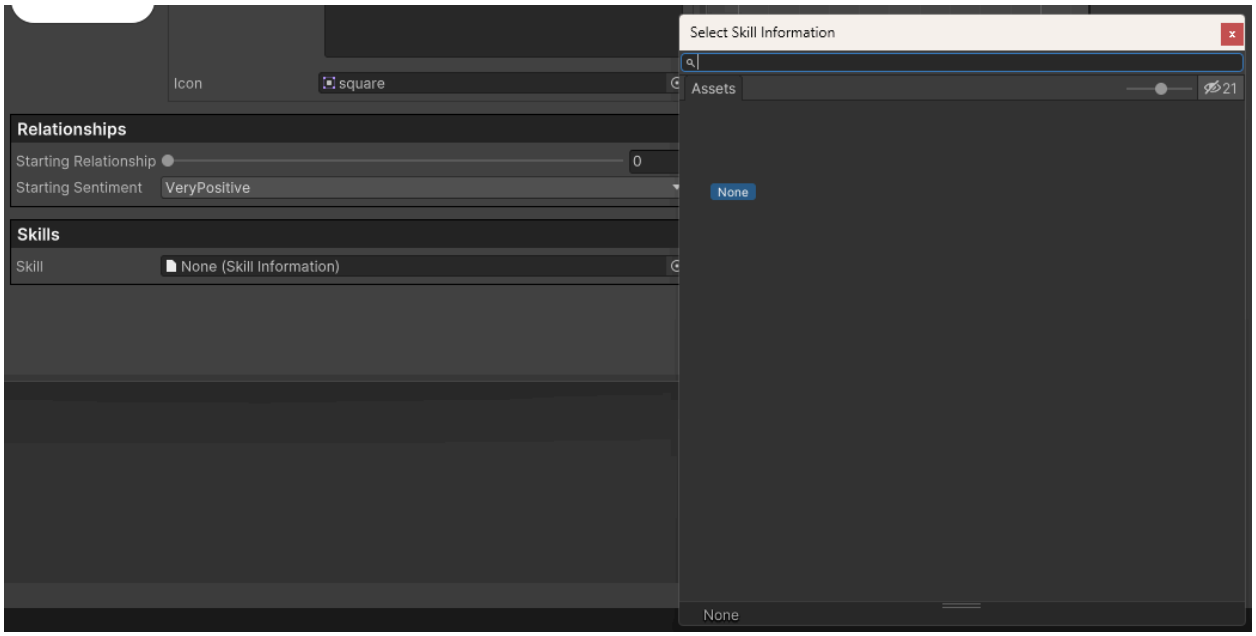
Under Relationships, set the Starting Relationship - the value of the player's relationship with this NPC when they first start the game.



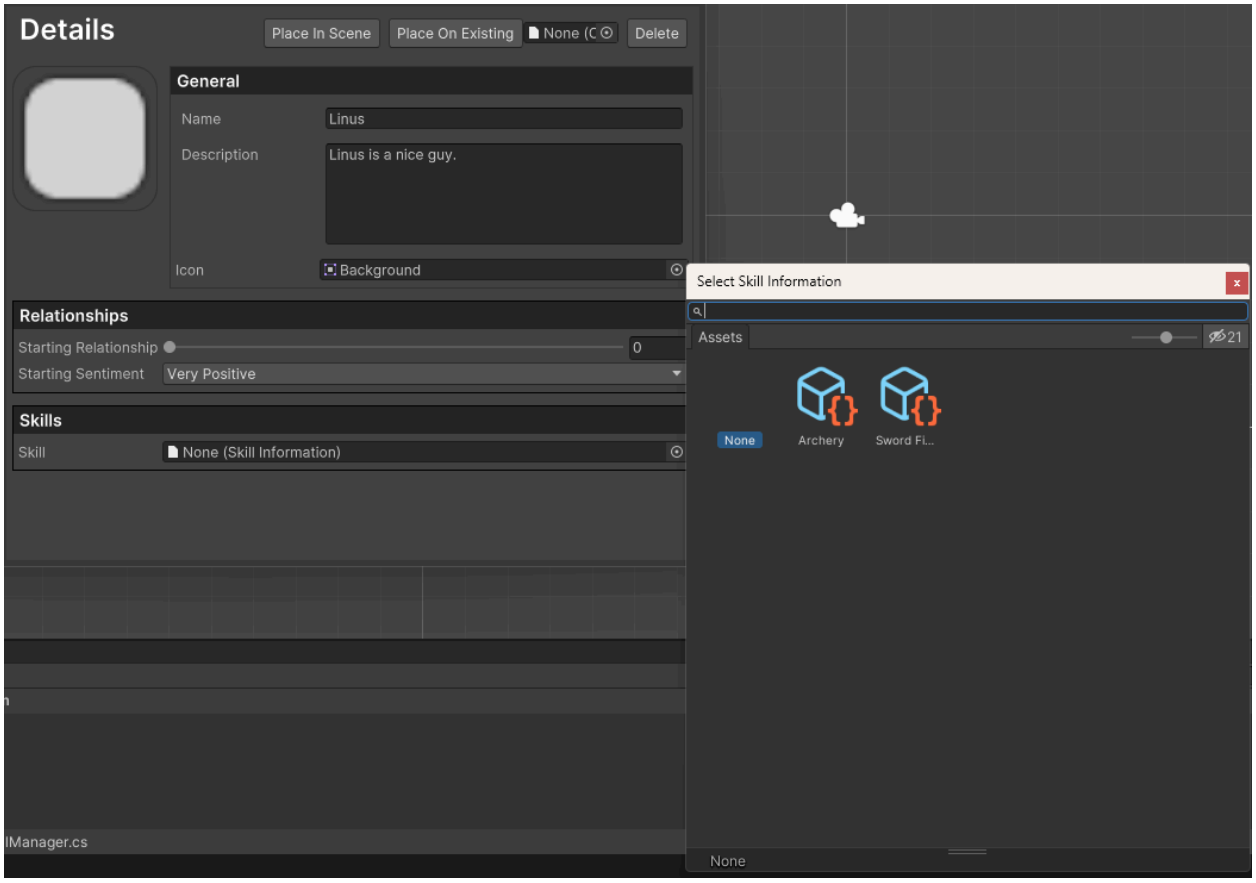
Set Starting Sentiment - the way this NPC feels towards the player when they first start the game.



Under Skills you can select a skill, but since we haven't created any yet, there are no options to select.



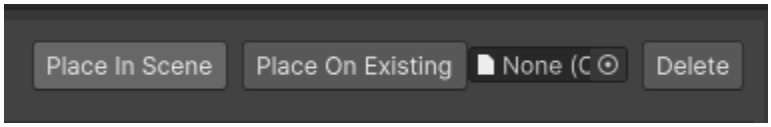
Once you have some skills, the menu will appear like this.



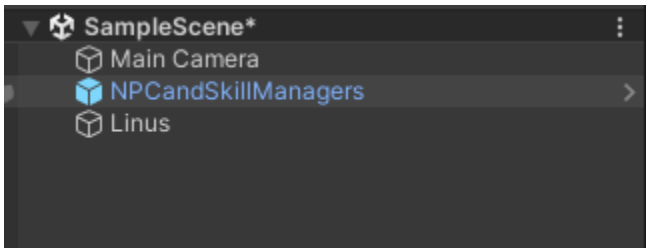
Now we have NPCs, we can **place** them into our **Scene**.

Select the desired NPC in the NPCs List.

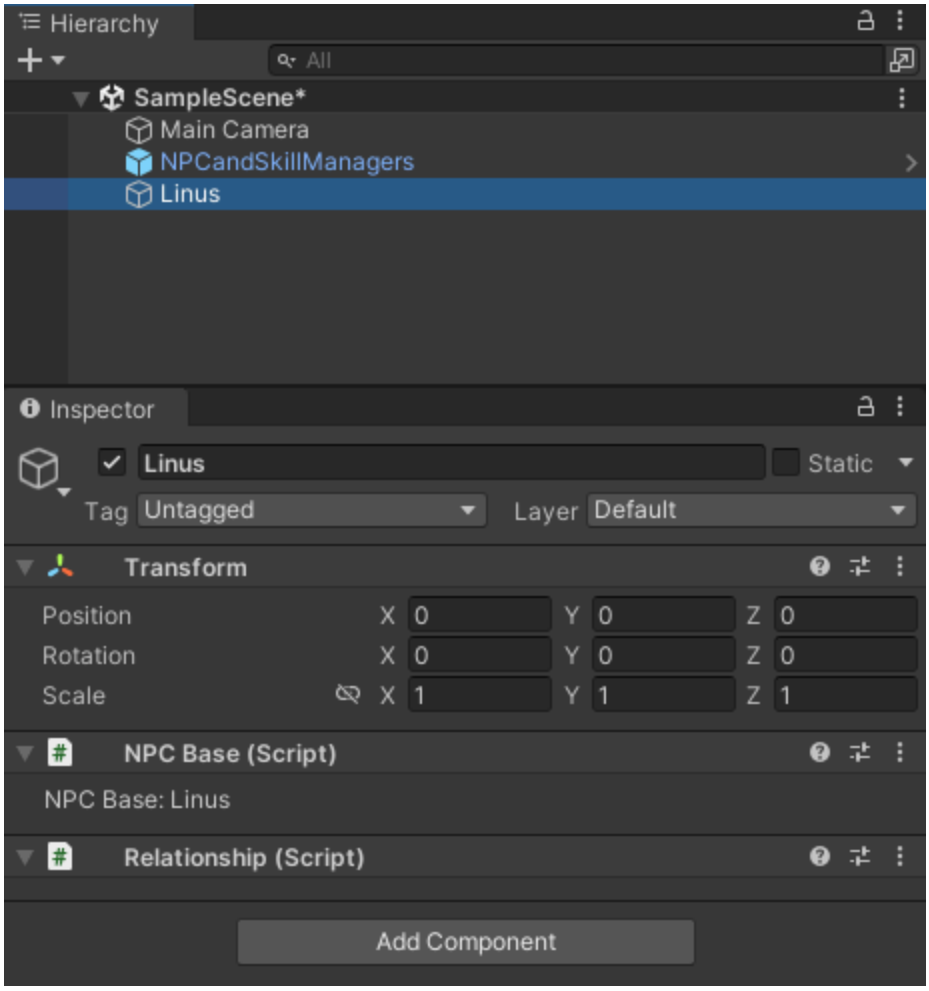
At the top of the Details Section there are 2 options.



Place In Scene will create a new GameObject containing the required components.

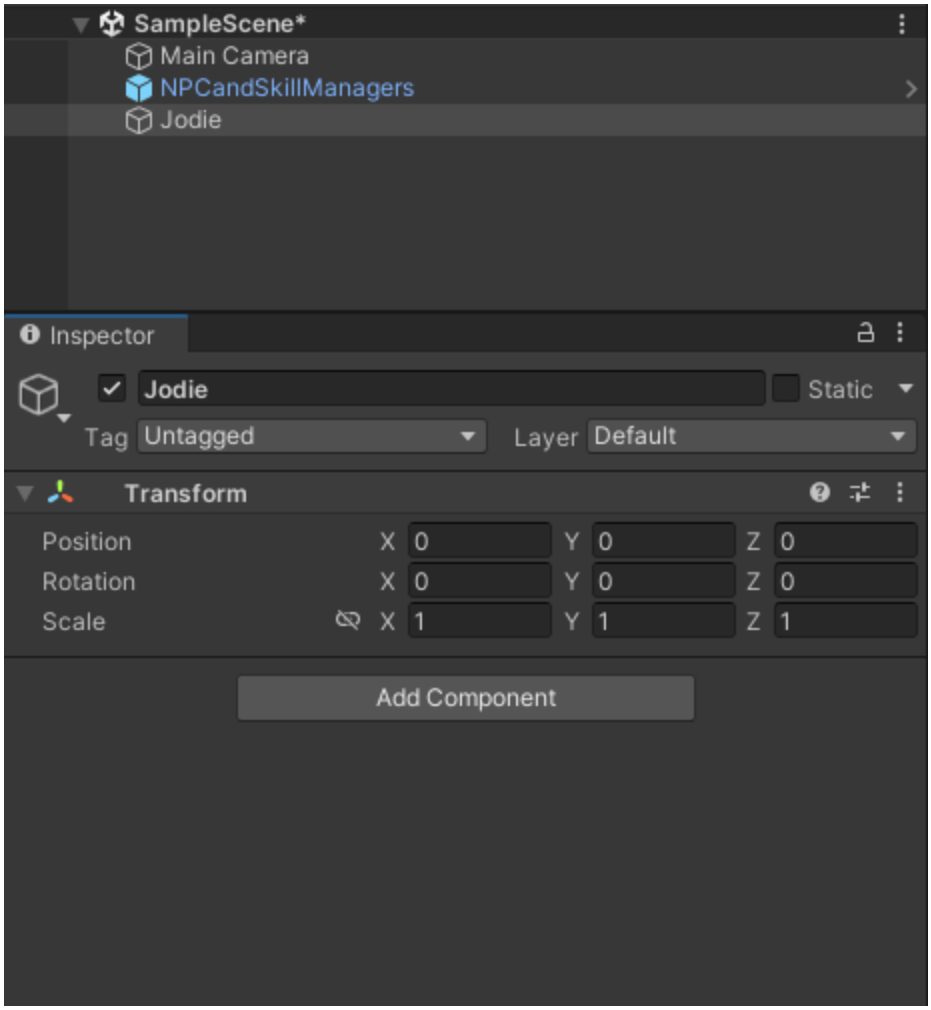


In the Inspector you can see the NPC Base script and the Relationship script.

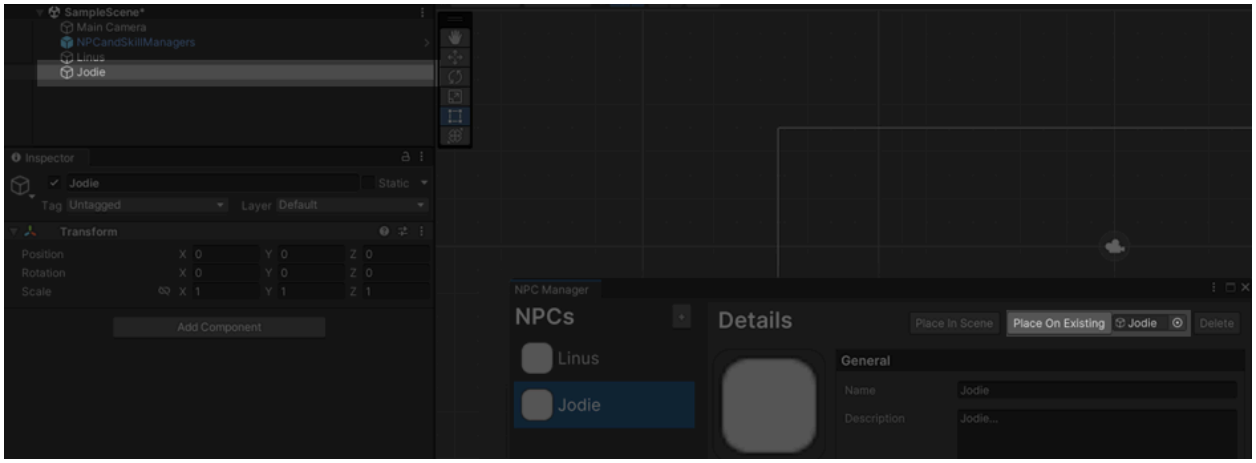


Note: You can check which NPC's data is connected to a GameObject under the NPC Base script.

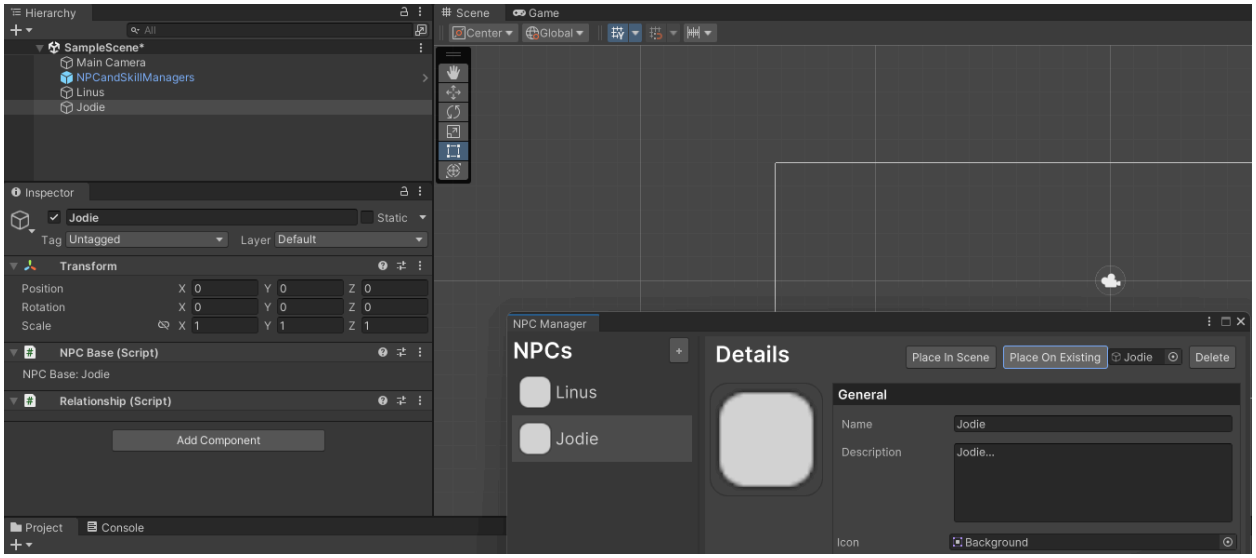
You can also connect existing GameObjects to an NPC.



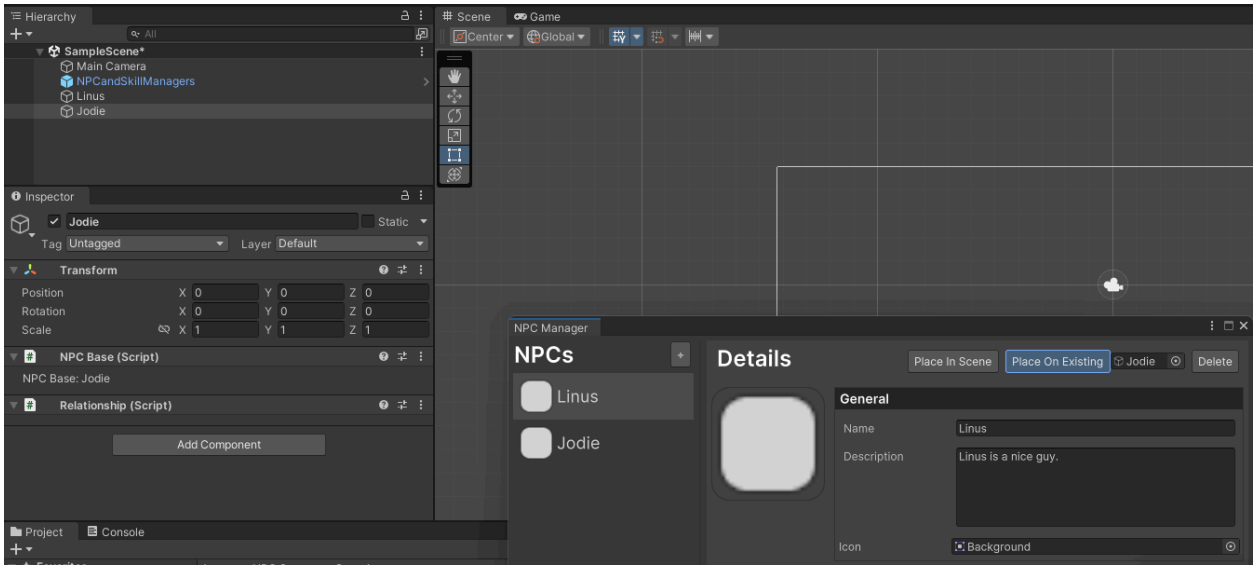
Click and drag the object from the Scene Hierarchy, into the slot next to the Place On Existing button.



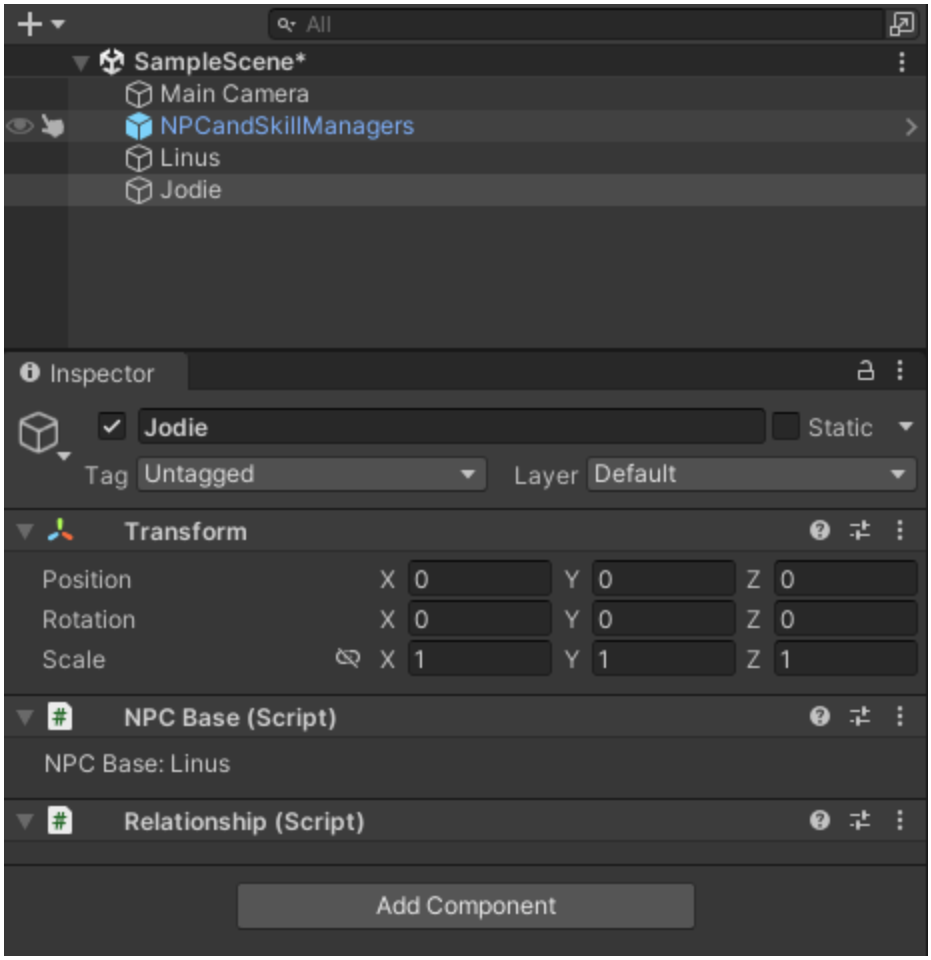
Press the Place On Existing button.



The needed components will be added to the GameObject.



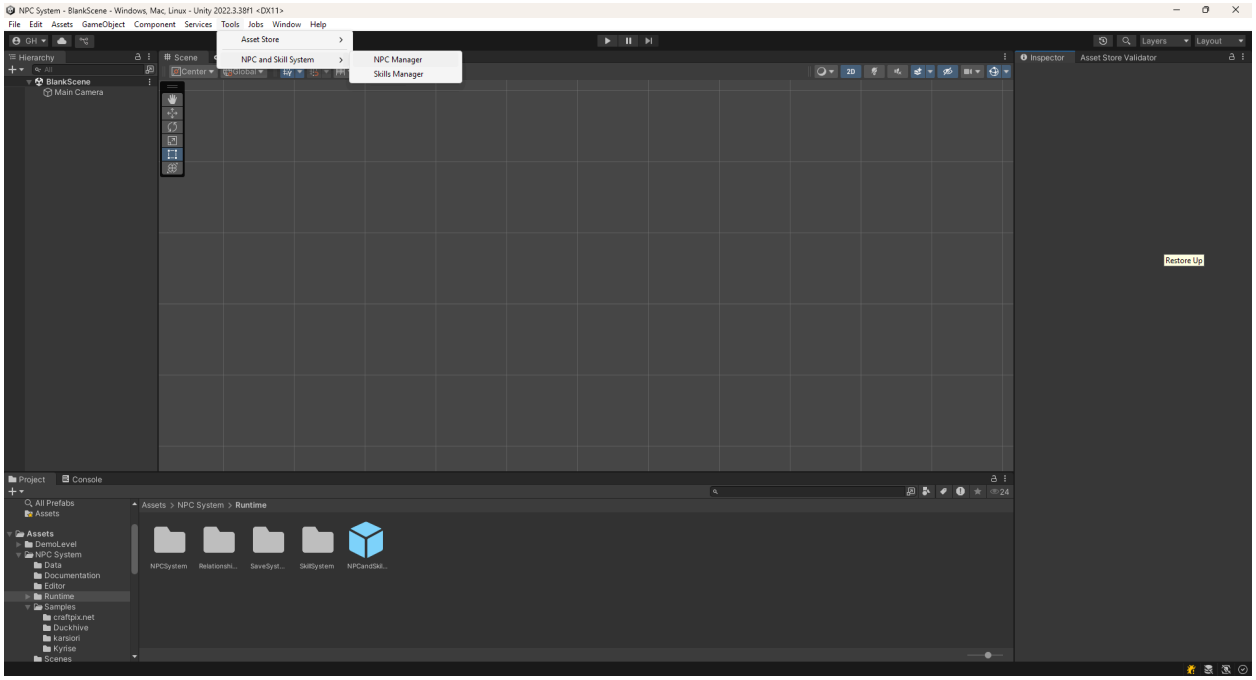
If you want to swap the NPC connected to the GameObject, simple repeat on the Details section of the new NPC.


All
SampleScene*
Main Camera
NPCandSkillManagers
Linus
Jodie
Inspector
Jodie
Static
Tag
Untagged
Layer
Default
Transform
Position
Rotation
Scale
X 0
Y 0
Z 0
X 0
Y 0
Z 0
X 1
Y 1
Z 1
NPC Base (Script)
NPC Base: Linus
Relationship (Script)
Add Component

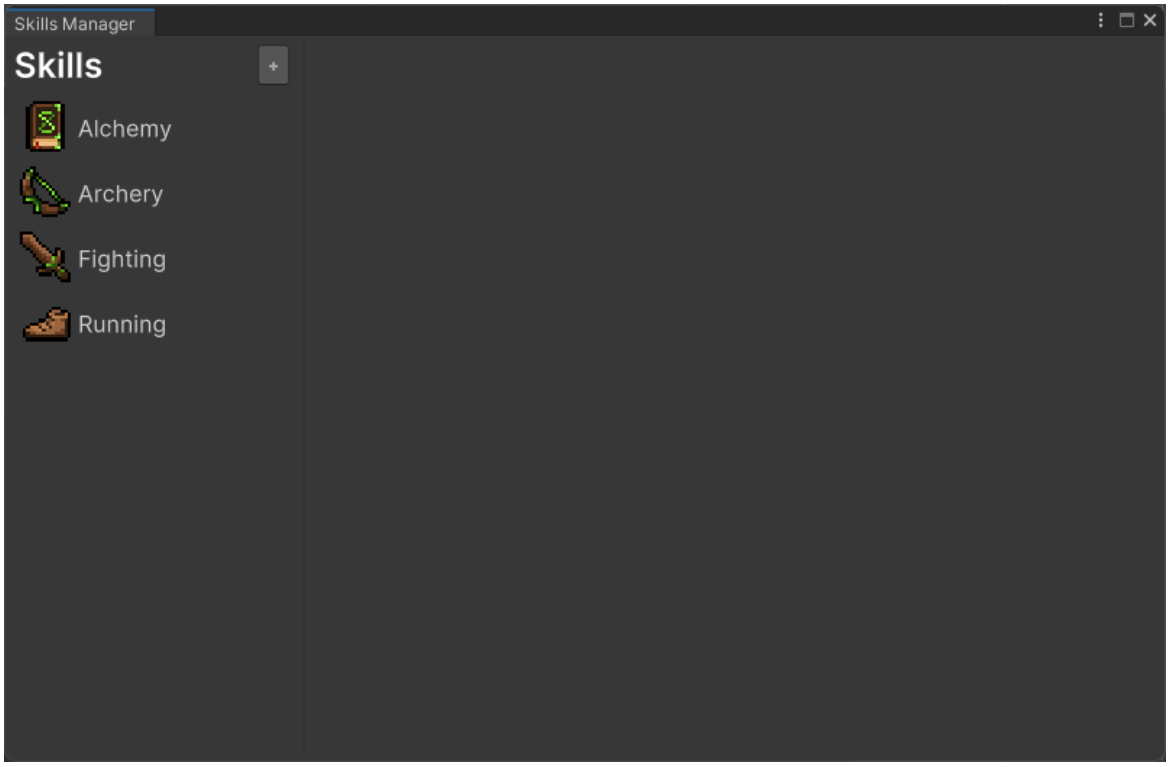
Creating Skills

This section will describe how to create and manage skills.
To create and manage skills you will need to open the Skills Manager window.

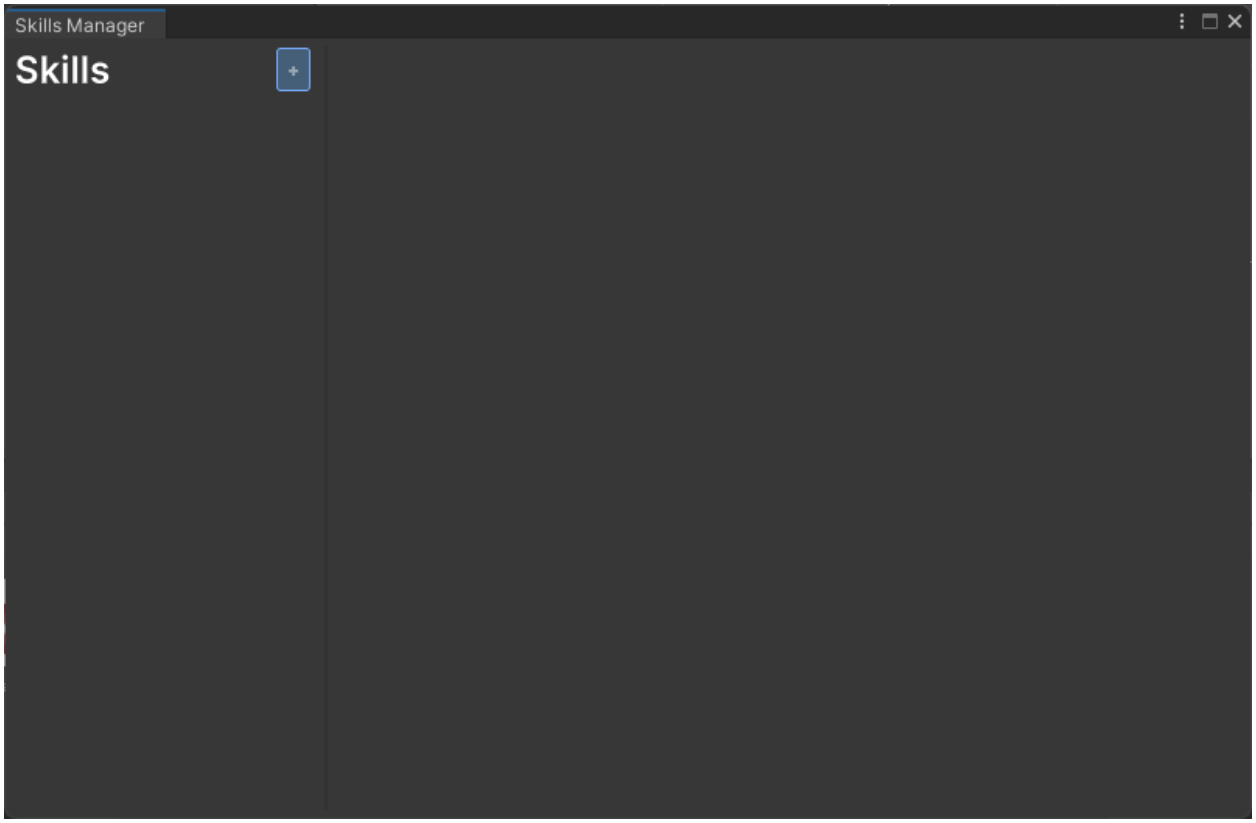
On the toolbar at the top of the Editor, go to Tools, NPC and Tool System, Skills Manager.



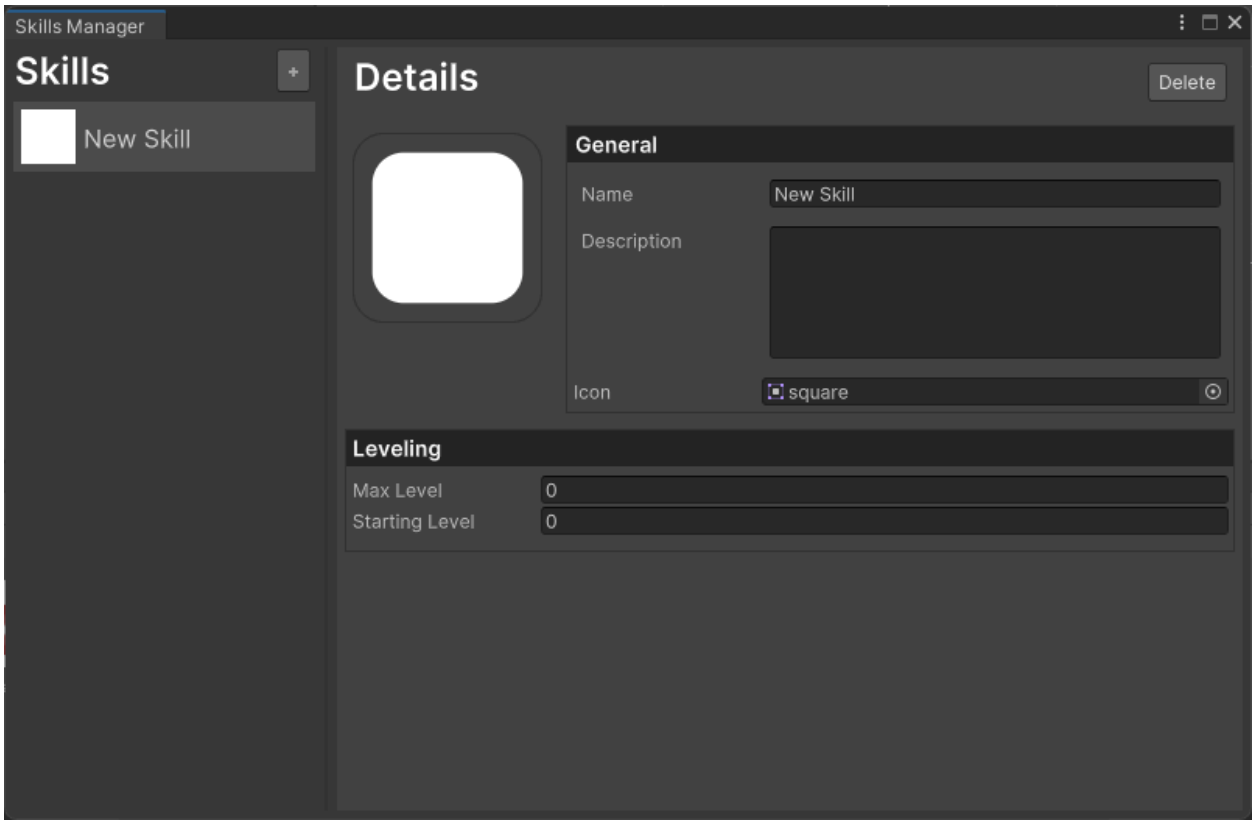
There’s some examples to get you started, but you can delete them anytime.



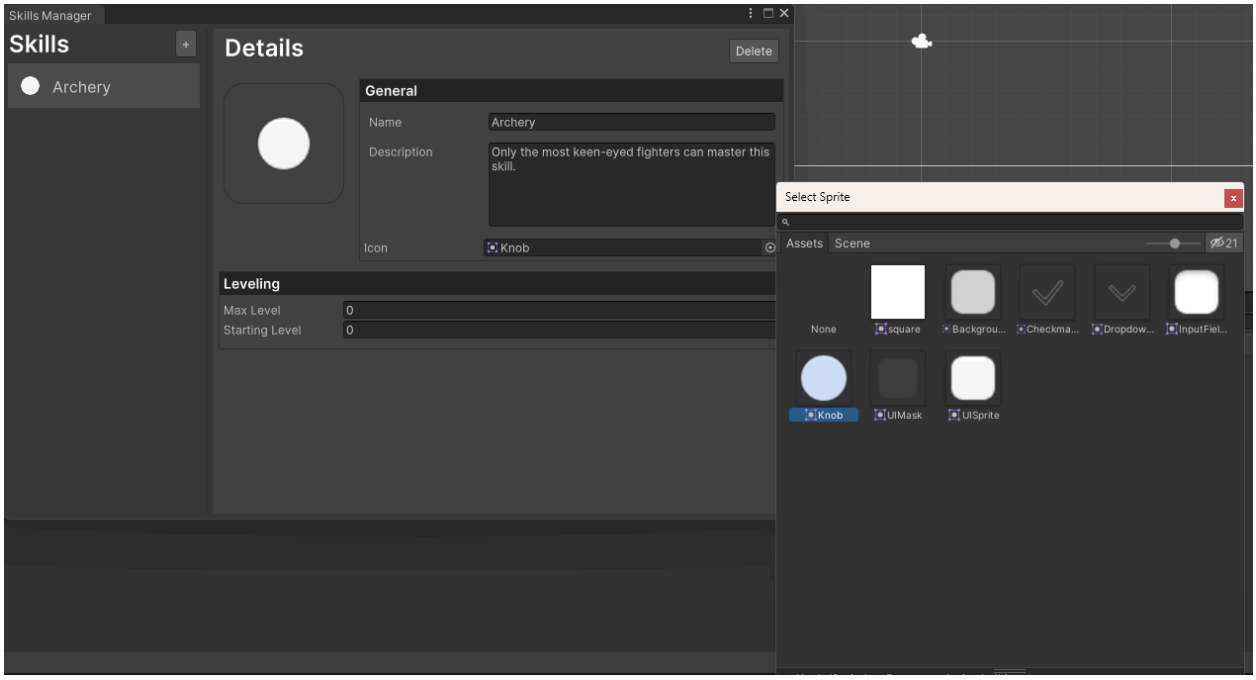
Press the + button.



The newly created skill is shown in the Skills List on the left, and the details are shown in the Details Section on the right.

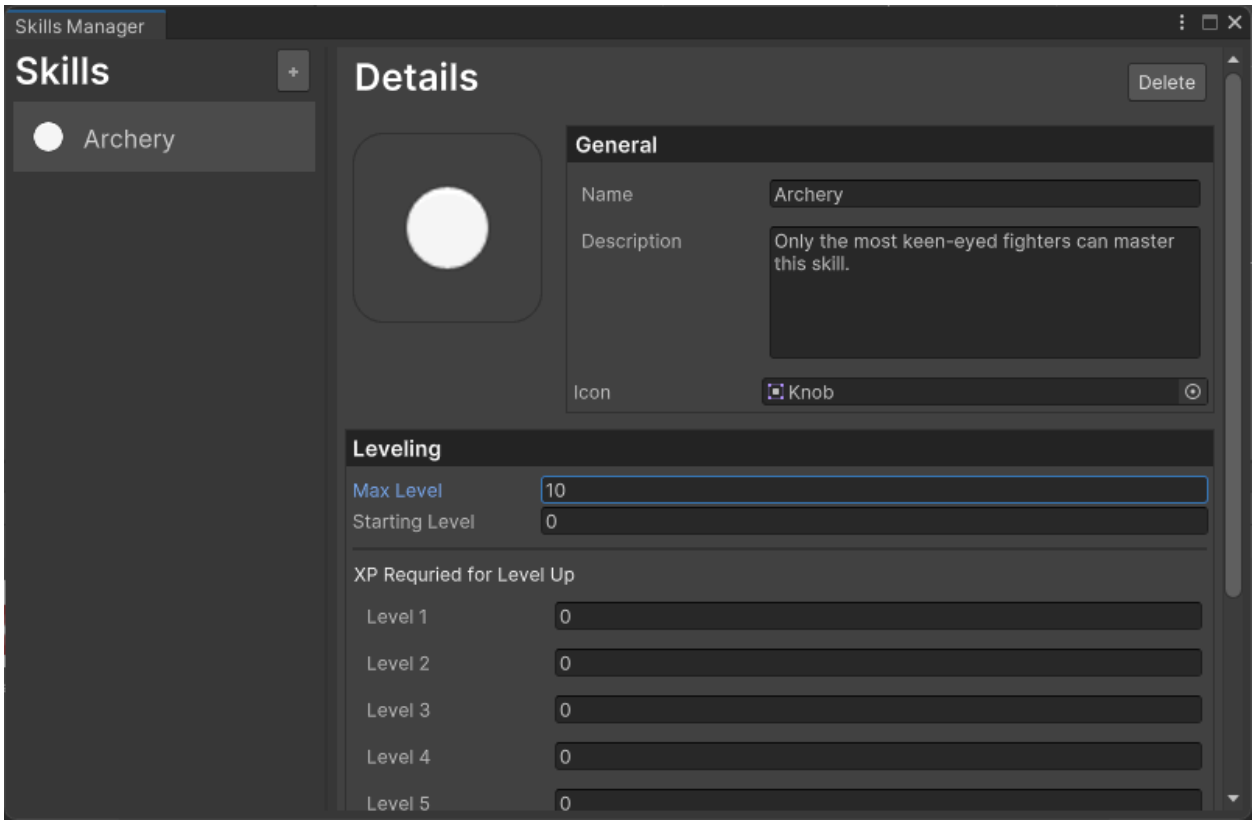


Define the name, description and set an icon.

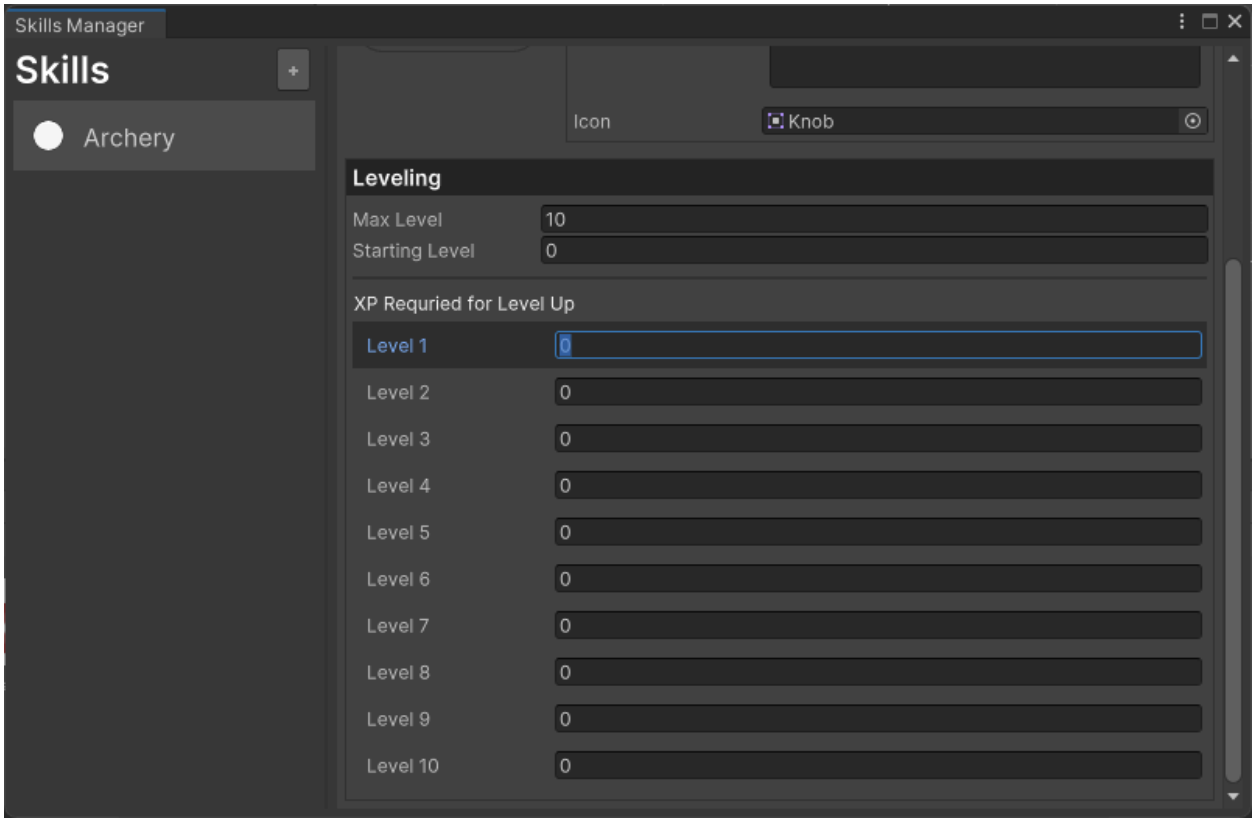


The Leveling section allows you to define how many levels the skill has and how the player will progress.

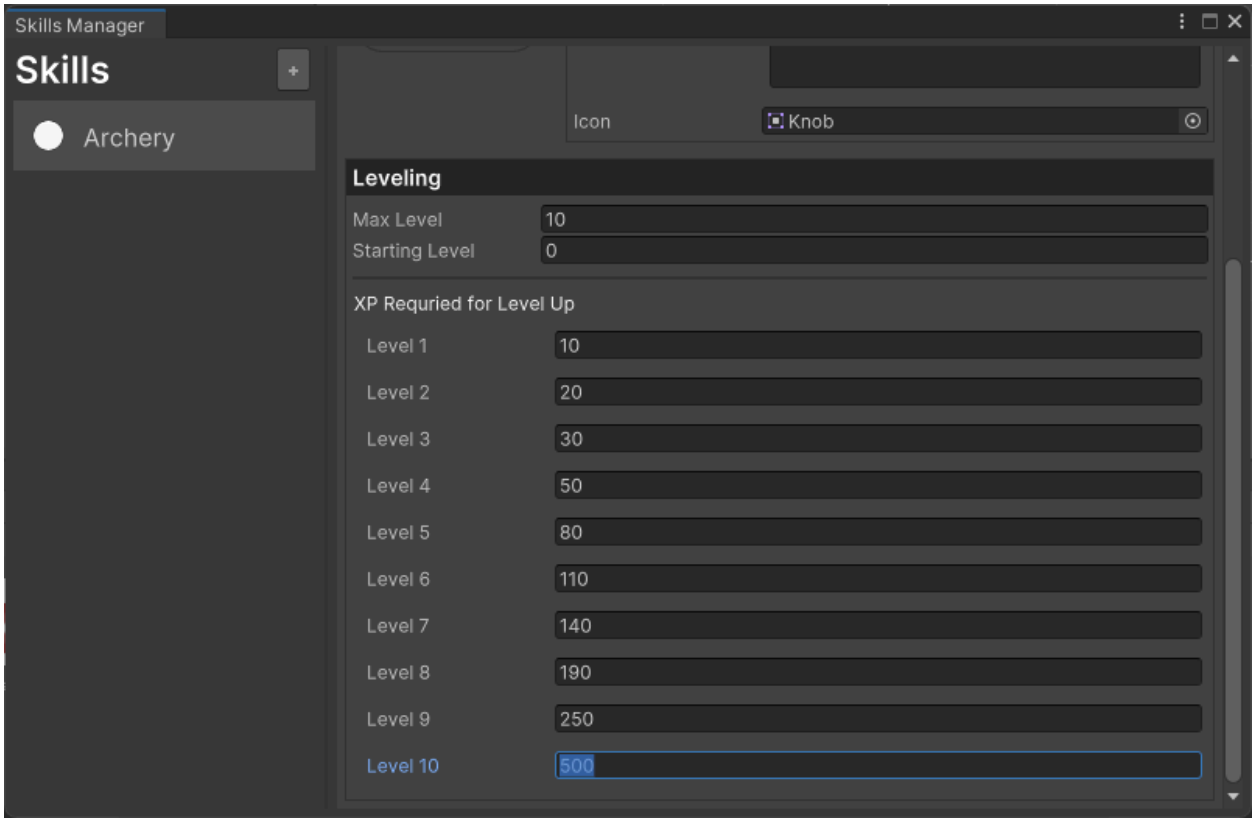
First, define the Max Level you want for the skill and the level the player will have when playing the game for the first time.



Once this value is entered, the number of levels will appear below int he XP Required for Level Up section.



Enter in the values for each level.

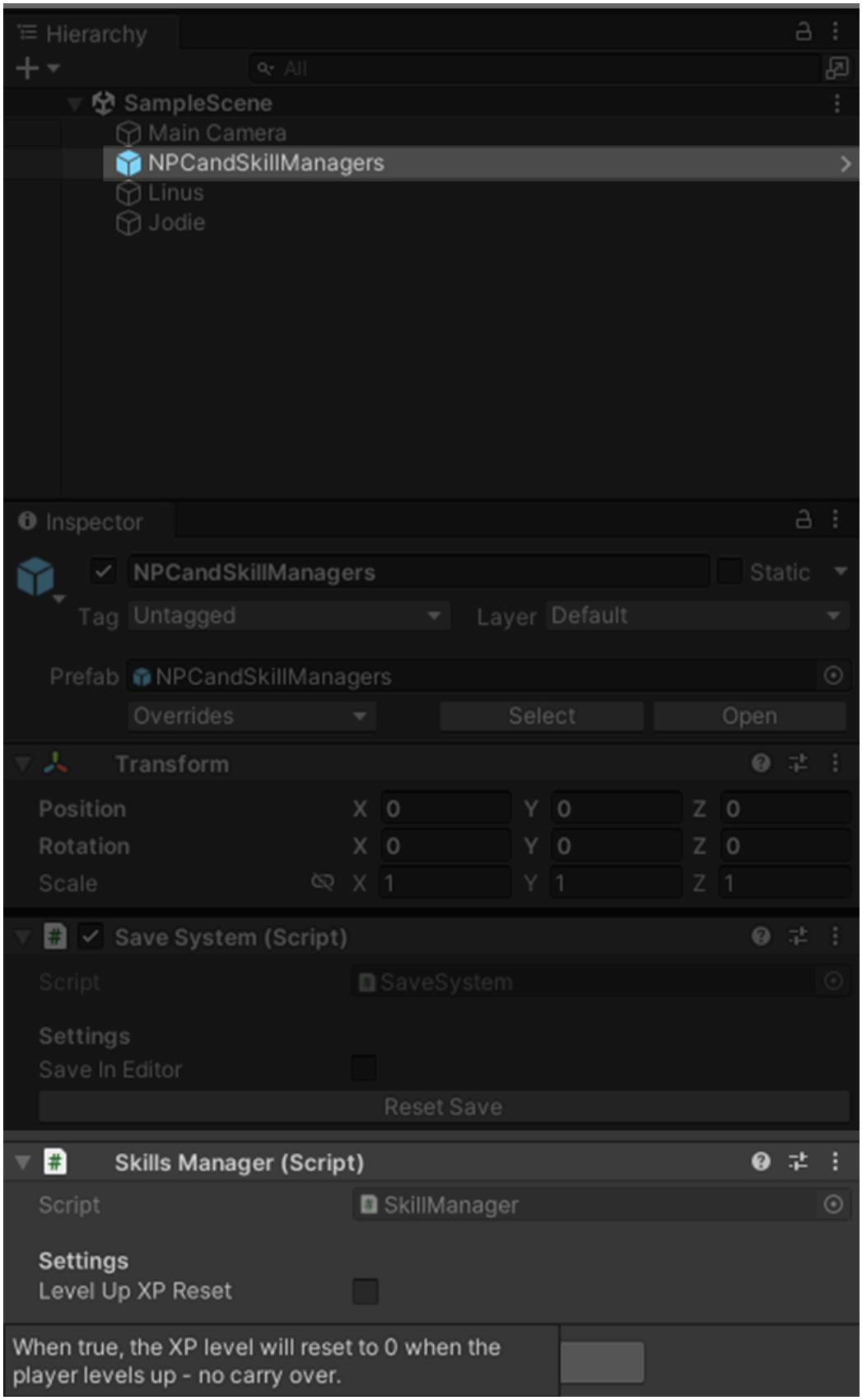


Note: These values are non-cumulative (the 10 XP gained in Level 1 don't count towards the 20 required for Level 2)
Note: All values must be non-zero.

Leveling Settings

You can decide how you want the leveling to be calculated.

Navigate in the Scene Hierarchy to the NPCandSkillManagers object. In the inspector - under Skill Manager, Settings - you'll see the Level Up XP Reset option.


Hierarchy
All
SampleScene
Main Camera
NPCandSkillManagers
Linus
Jodie
Inspector
NPCandSkillManagers
Static
Tag
Untagged
Layer
Default
Prefab
NPCandSkillManagers
Overrides
Select
Open
Transform
Position
Rotation
Scale
X 0
Y 0
Z 0
X 1
Y 1
Z 1
Save System (Script)
Script
SaveSystem
Settings
Save In Editor
Reset Save
Skills Manager (Script)
Script
SkillManager
Settings
Level Up XP Reset
When true, the XP level will reset to 0 when the player levels up - no carry over.

When true, the XP will be set to 0 when the player levels up and no XP can be carried over.

Example

Player is Level 1 and has 8 XP

XP required for Level 2 is 10.

The player gains 4 XP.

The player levels up to Level 2.

When Level Up XP Reset is true - the player will have 0XP.

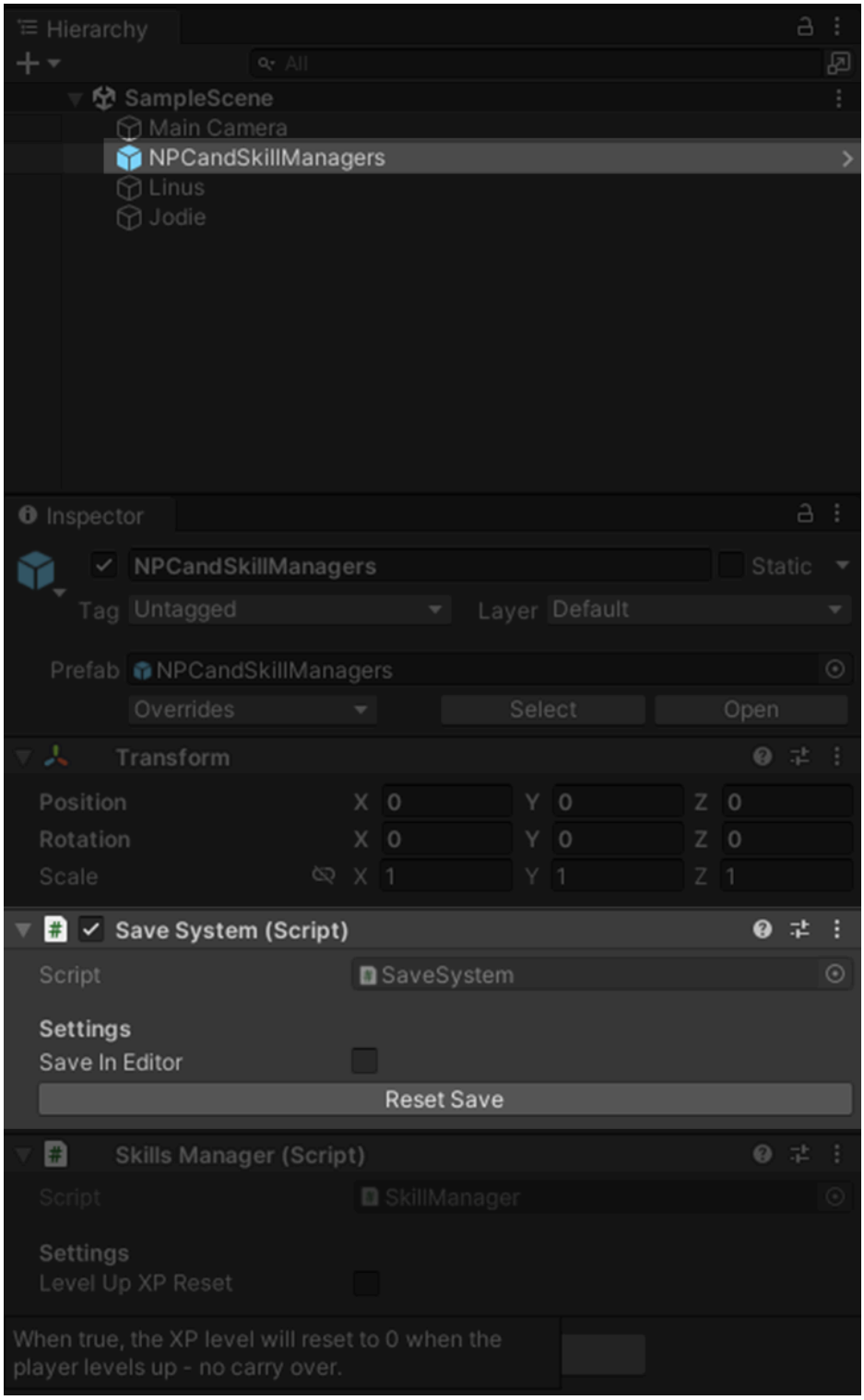
When Level Up XP Reset is false - the player will have 2XP.

Save System

The package comes with a save system which keeps track of relationship data and player skills during and between play sessions.

While viewing the Inspector for the NPCandSkillManagers object, you may have noticed the Save System settings.

The Save In Editor option allows you to decide if the game should save while testing in the editor.



While true, the game will save and load as if in play mode.

You've got the basics down, time to start creating!

Documentation

Classes, Structs and Enums

NPCBase
The NPC Base component is used to store and manage NPC data. It also handles the saving of relationship data.

Parameters

Name	Type	Description
NPCName	string	The name given to the NPC in the NPC Manager.
NPCDescription	string	The description given to the NPC in the NPC Manager.
Skill	SkillInformation	The skill associated with the NPC in the NPC Manager.

Relationship
The Relationship component handles the storage and update of relationship information for this NPC.

Methods

Name	Parameters	Returns	Description
AddSentiment	newSentiment (Sentiments)	None	Assigns the given sentiment to this NPC.
UpdateRelationship	Change (float)	None	Changes the current relationship value by the given change. Negative values will reduce value.
GetRelationshipInfo	None	RelationshipInformation	Return current relationship data in RelationshipInformation

			struct. Returns an empty struct if no data is found.

Skill Manager
This class handles changes in skill levels and XP values.
Access the class with its instance - SkillManager.Instance.

Methods

Name	Parameters	Returns	Description
GetSkillByName	skillName (string)	SkillInformation	Returns the SkillInformation struct from the name of the skill. Returns null if skill not found.
GetSkillLevel	skillName (string)	int	Return current level of skill. Returns 0 if skill not found.
GetXPLevel	skillName (string)	int	Return current XP level of skill. Returns 0 if skill not found.
GainXP	skillName (string), gainedXP (int)	None	Increase the XP level of skill with the name 'skill', by the amount of 'gainedXP'. Leveling up is also handled here.
LevelUp	skillName (string), resetXP (bool)	None	Increase skill level by 1. If resetXP is true, XP level is reset to 0.

SkillInformation
This class stores the definition of skill, as created in Skill Manager. Values can be accessed here but should only be edited in the Skill Manager.

Parameters

Name	Type	Description
SkillName	string	The name given to the Skill in the Skill Manager.

SkillDescription	string	The description given to the Skill in the Skill Manager.
MaxLevel	int	The maximum level that can be reached for this skill.
StartingLevel	int	The skill that the player starts with when first playing the game.
RequiredXP	List<int>	A list of values for the XP levels required for each skill level. This is ordered from the lowest to highest level.

Sentiments
This enum is used to show the current feeling of an NPC towards the player. This value impacts the player's ability to gain/lose relationship with this NPC.

Values	**Index**	**Impact**
VeryPositive	0	100% increase on relationship gain. 50% reduction in relationship loss.
Positive	1	50% increase on relationship gain. 25% reduction in relationship loss.
None	2	None
Negative	3	25% reduction on relationship gain. 50% increase in relationship loss.
VeryNegative	5	50% reduction on relationship gain. 100% increase in relationship loss.

RelationshipInformation
This struct stores the current relationship data.

Parameters

Name	**Type**	**Description**
relationshipValue	float	The value of the current relationship between this NPC and the player.
currentSentiment	Sentiments	The current sentiment held by this NPC.