Make your own 2D Platformer

# Introduction

This tutorial will take you through the steps required to develop a simple 2D platformer game using Unity and C#.

You will be able to design your own levels using Unity, which you can then give to friends and family to play.

# Required Software

Unity is the only required piece of software, make sure MonoDevelop is installed when installing Unity.

<http://www.unity3d.com/get-unity>

# Let’s Begin

## Starting

* Open Unity.
* Choose ‘Open Other’.
* Navigate to the project directory and choose ‘Select Folder’.

## Unity Interface

* The Play button is at the top.
* The ‘Game’ window is the final result of everything put together when you press the play button.
* The ‘Scene’ window is how you view your game while you are making it.
* The ‘Hierarchy’ is a list of everything that is in the Scene.
* The ‘Inspector’ is a way to view the details of a particular GameObject.
* The ‘Project’ window is where you keep all of your files.

## Variables

* If you press play, the green alien just falls off the screen.
* To fix that we are going to modify the Players’ Position X variable.
* A variable is a word that represents something, like a number.
* At the moment the Players’ Position X variable is -5, let’s change it to 0.
* Press play again, now the green alien (or the ‘Player’) lands in the middle of the platform.

## Movement

* At the moment the Player can’t do anything.
* So we’ll make the Player move.
* In the ‘Project’ window navigate to the Scripts folder and double click the Player script.
* Add the following code between the ‘Player Update’ and ‘End Player Update’ sections.

|  |  |
| --- | --- |
| //--------------------------------  // Player Update   |  | | --- | | if (Input.GetAxis ("Horizontal") != 0.0f) {  rigidBody.AddForce(new Vector2(Input.GetAxis ("Horizontal") \* movementSpeed, 0.0f));  }  animator.SetFloat("Speed", Mathf.Abs(Input.GetAxis("Horizontal"))); |   // End Player Update  //-------------------------------- |

* Save the file and then go to back to Unity and press the Play button.

## Jumping

* Next we’ll work on making the Player jump.
* Add the following code just below the code that we added for movement, but above the ‘End Player Update’ section.

|  |  |
| --- | --- |
| |  | | --- | | if (Input.GetAxis ("Horizontal") != 0.0f) {  rigidBody.AddForce(new Vector2(Input.GetAxis ("Horizontal") \* movementSpeed, 0.0f));  }  animator.SetFloat("Speed", Mathf.Abs(Input.GetAxis("Horizontal"))); |   // End Player Update  //-------------------------------- |