**GAMEDEVREJECTS:**

**UNITY EDITOR INTERFACE**

**Orlando Unity3d Development Group**

**Date: 12-Jan-22**

**All Copyright GAMEDEVREJECTS©2021**

Contents

[**01.4.0** **Unity Editor Interface** 3](#_Toc92921580)

[**01.4.1** **Interface Overview** 3](#_Toc92921581)

[**01.4.2** **(Top) Menu Bar** 3](#_Toc92921582)

[**01.4.3** **(Middle) Toolbar** 4](#_Toc92921583)

[**01.4.4** **6 x Windows: Scene, Hierarchy, Project, Inspector, Game, and the Console** 4](#_Toc92921584)

[**01.4.5** **Hierarchy Window** 4](#_Toc92921585)

[**01.4.6** **Scene Window** 4](#_Toc92921586)

[**01.4.7** **Game Window** 4](#_Toc92921587)

[**01.4.8** **Inspector Panel** 4](#_Toc92921588)

[**01.4.9** **Project Panel** 4](#_Toc92921589)

[**01.4.10** **Console Window** 4](#_Toc92921590)

[**01.4.11** **Unity XYZ axes** 5](#_Toc92921591)

[**01.4.12** **Create 1 x GameObject** 5](#_Toc92921592)

[**01.4.13** **Main Scene Navigation Maneuvers** 5](#_Toc92921593)

[**01.4.14** **Manipulating the Game Object** 5](#_Toc92921594)

[**01.4.15** **Toolbar Tools Panel** 5](#_Toc92921595)

[**01.4.16** **Menu Bar - File** 9](#_Toc92921596)

[**01.4.17** **Menu Bar - Edit** 9](#_Toc92921597)

[**01.4.18** **Menu Bar - Assets** 11](#_Toc92921598)

[**01.4.19** **Menu Bar - GameObject** 12](#_Toc92921599)

[**01.4.20** **Menu Bar - Component** 13](#_Toc92921600)

[**01.4.21** **Menu Bar - Window** 14](#_Toc92921601)

[**01.4.22** **Menu Bar - Help** 15](#_Toc92921602)

[**Glossary** 17](#_Toc92921603)

[**Resources** 18](#_Toc92921604)

# **01.4.0 Unity Editor Interface**

The objectives are:

* **Learn to navigate the Unity Editor Interface**
* **Change the development scene’s point of view with the Scenes Manipulation tools**
* **Know how to transform a Game Object’s position, rotation, and scale using the Game Object’s manipulation tools**

Our goals are to introduce you to the Unity’s editor, the interface, where all the key tools are. How to navigate your workspace. We will talk about different concepts like translating, rotating, in 3D space, about the different axes.

We will show you where the different manipulation tools are, how to use them to move your Game Objects to different locations, and how to get the different points of view or perspectives.

# **01.4.1 Interface Overview**

So let’s start with a high-level tour of the editor interface. The Unity Editor interface is made up of the following:

* 1 x **(Top)** **Menu bar**.
* 1 x **(Middle)** **Toolbar**; and
* Graphical user interface, application

  Description automatically generated6 x Tabs or Windows the: **Scene, Hierarchy, Project, Inspector, Game, and the Console**

# **01.4.2 (Top) Menu Bar**

Starting from the Top of the Screen, below the Title. We have the Top Left horizontal menu bar, which gives you access to all of Unity’s components and tools to add different functionality to the gameobjects in your scene. The following options:

* **File, Edit, Assets, GameObject, Component, Window, and Help**

# **01.4.3 (Middle) Toolbar**

Now immediately beneath the Menu Bar is the **Toolbar**.

This contains all the “manipulation tools” to move your Game Objects, around in your scene.

The toolbar is not a window and it is the only part of the Unity interface that you cannot rearrange.

# **01.4.4 6 x Windows: Scene, Hierarchy, Project, Inspector, Game, and the Console**

Beneath the menu bar and toolbars, you have a collection of 6 x (tabs) Windows and Panels the **Scene, Hierarchy, Project, Inspector, Game, and the Console** in default layout (you can arrange them to suit your personal preferences).

# **01.4.5 Hierarchy Window**

Navigating from left to right you have the Hierarchy Window. The Hierarchy window lists all the Game Objects in your application or Scene. The Hierarchy is organized in an outline “list” view; so they are easier to find. By default when you create a new project Unity will install two Game Objects: **1 x Main Camera, 1 x Directional Light** in a sample scene. The Main Camera will dictate the visual Point of View of your User or Player. So every application you create whether its 2D or 3D in Unity will require at least 1 camera, simply because you need to display the rendered scene to the user engaging in your application, as simple as that. The Directional Light will dictate the shadows cast on the GameObjects in the Scene; and where the light is coming from; and you can have multiple light sources in your scene, to create different effects obviously.

# **01.4.6 Scene Window**

The main window, where you see the game you are creating. The visual space inside the Unity IDE where you can add, edit, change & modify your gameobjects/assets. In a nutshell Games or applications are made simply by linking 1 or more scenes (aka, levels) linked together. The Scene view is the point of view as a developer.

# **01.4.7 Game Window**

The Game window is the rendered view of your Scene Window from the Camera(s) in your game. It is representation of your final, published game, on a particular device. The Game View is the point of view as a user.

# **01.4.8 Inspector Panel**

On the far right, we have the Inspector Panel. The inspector details the position, rotation, scale, dimensions, (i.e. length, height, width etc.) of the gameobject you have selected or highlighted in your Scene View. In addition the inspector details all the components and layers you have added to the Game Object E.g. like Scripts, Textures, Materials, Physics, Tags etc. In other words, if you navigate to the Hierarchy, and select a game object, in your scene view. The inspector is going to detail and display, all the components that constitute that Game Object.

# **01.4.9 Project Panel**

At the very bottom of your screen, is the Project Window. The Project Window is a repository where you access all the asset files (e.g. like image files, 3d models, audio files, meshes, scripts, animations etc.), data files etc. from a “physical location” on your hard drive to use in your application. Also in your Project Panel, you have a Favorites drop-down list i.e. **All Materials, All Models, All Prefabs**, this is a quick search option. Unity will search all the assets in your folder based on the specific asset filter type you have selected.

# **01.4.10 Console Window**

The Console Window shows errors, warnings and other messages generated by Unity.

You can also show your own messages in the Console using the Debug class.

Everything that is written to the Console Window (by Unity, or your own code) is also written to a Log File.

Open the Console from Unity’s main menu, as follows:

[ Navigate to MENU BAR ] – (Select).Window -- General -- < Console >.

# **01.4.11 Unity XYZ axes**

Before we dive into the Navigation and Manipulation. Let’s take a moment to explain some essential Geometry. In Unity in 2D/3D world. The axes are always represented by the same color; which is why you have the gizmo in the top right corner to remind you. The Red X axis denotes Left and Right (Horizontal plane), the Green Y-axis is Vertical (Up and Down), and the blue z-axis (is your forward and back axis i.e. in and out of your screen). All relative to the Camera’s frustrum located in your Scene.

# **01.4.12 Create 1 x GameObject**

Okay to demonstrate more of the functionality inside of the Unity Editor. Let’s create a GameObject as follows:

**(NavigateTo).SCENE -- (IN).HIERARCHY -- (Right Click).3D Object -- / (Select).Cube /**

**(NavigateTo).INSPECTOR – (IN).TRANSFORM => / (Select).”3DOTS” – (Select).Reset /**

**(IN).TRANSFORM – POSITION -- / (CheckType).< X= 0, Y= 0, Z= 0 > /**

**(IN).TRANSFORM – ROTATION -- / (CheckType).< X= 0, Y= 0, Z= 0 > /**

**(IN).TRANSFORM – SCALE -- / (CheckType).< X = 1, Y= 1, Z= 1 > /**

Which is placed in the Centre of the World, which is also locked in the Centre of your Point of View.

# **01.4.13 Main Scene Navigation Maneuvers**

Now to change your point of view relative to a Game Object in your Scene this is done by 3 main navigation maneuvers: **Zoom, Orbit, and Grab (aka Move)**

The 3 movements involve clicking and dragging while holding down a combination of Alt (or Option on the Mac) and Control. (The exact controls vary for one-, two-, and three-button mice),…Here are the scene controls for Navigating the Scene using a 2-button mouse.

* **Zoom = Scroll Wheel (Roll Forward/Back) or Alt (Hold Down) + Right Click (Mouse) + Drag (Forwards/Back)**
* **Orbit = Alt (Hold down) + Left Click (Mouse) + Drag**
* **Grab = Alt (Hold down) + Scroll Wheel (Press Down) + Move Mouse**

Please note When Orbiting Unity locks your “point of view” and tries to keep the Game Object in the Centre of your Scene Space

Now, you do need a 3-button mouse to do this, and you do have the option to use the keyboard to execute the same maneuvers, but we don’t recommend it the easiest way is to use a mouse.

# **01.4.14 Manipulating the Game Object**

To manipulate your point of view relative to a Game Object, firstly you need to ensure your cursor lies within the boundaries of the Scene Window. We use the Hierarchy panel to select the Object

* **Focus = Shift + F** (places the object in your center point of view)

# **01.4.15 Toolbar Tools Panel**

The Toolbar consists of several “clusters” of controls. Each relate to different parts of the Editor.

The first tool in the toolbar, **the Hand Tool**, allows you to orbit pan (left, right, up and down around the Game Object, which always remains in the local centre of your Scene.

The **Move Too**l lets you move Game Objects along each of the individual XYZ axes by clicking and dragging the individual arrows on the gizmo.

The **Rotate tool** allows you to change the Game Object's position relative to the centre of the XYZ axes in degrees of rotation by clicking and dragging the axes of the wireframe sphere Gizmo that appears around it.

The **Scale tool** (looks like the move tool but has cubes at the end of the gizmo axes) lets you rescale the GameObject in the direction your dragging the gizmo. Now if you want to scale the cube uniformly in all 3 axes directions and not distort the cube then you will need to select, grab the centre of the gizmo.

The **Rect tool** is used to edit 2d imported images, or 2D game objects like planes in the 3d world. Once you select the Rect tool you to resize or rescale the image by dragging the blue spheres at the corners or the circle uniformly in the centre. Now the Rec tool can be used on 3d objects, but the direction of the axes will change dependent on your point of view.

The **All-in-one**-Scene Transform tool which allows you to access the Move, Rotate, Scale tools; realised together in the same Scene View.

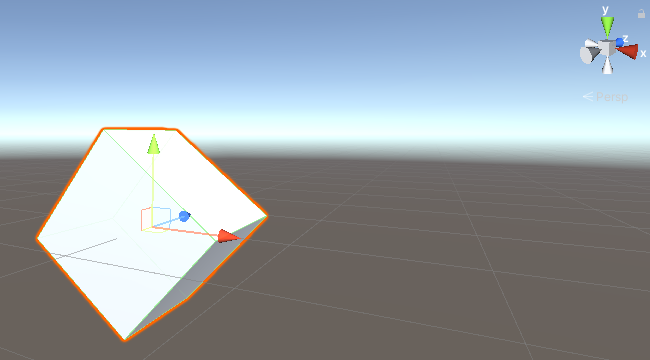
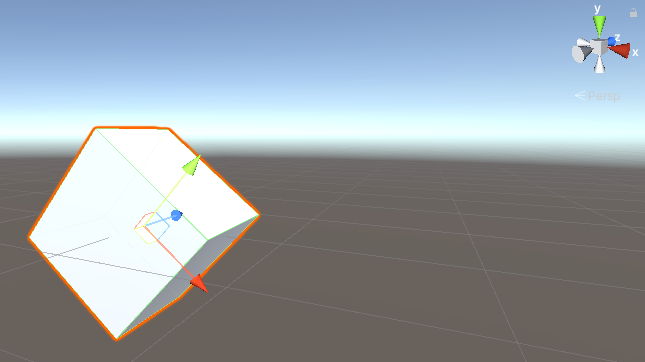
Alternatively, you can select each of the four Transform modes vis **the Shortcut keys**: **Q for Hand, W for Move, E for Rotate, R for Scale, T for Rect Tool and Y for Transform.**

The last tool is **the Edit Box Collider, which** allows you to adjust The Box Collider is the faint Green Outline surrounding the perimeter of a GameObject. It’s an essential component you need when you add physics to your Game Objects.

A box collider and edge colliders are used to detect collisions between gameobjects. They are an invisible shape around a gameobject, and trigger events to the console once the perimeter or volume of the collider has been breached by another gameobject.

Now you can also edit the Box Collider in the Inspector, and in your scene by grabbing the small little green squares on each of the colliders faces.

**Pivot and Local** Toggle Buttons. To demonstrate **the Local tool**, Lets rotate a game object at an angle to one of the axes. Now do you notice the Move Tool gizmo rotates with the Game Object. That’s because the Move Tool is locked to the Game Object. So that’s what we call local axes. Ie. The axes in the Game Object itself. So if I want to change the orientation of the Move Tool, and make it referenced the World, then we click, toggle the Local Axes button, and snaps to the World or Global.



**the Pivot Tool**, now demonstrate the Pivot Tool for example, if we have to game objects in the scene. If we select both cubes, and leave it on Pivot, then rotate or move one of the objects, they both rotate, or translate about their own pivot points. How if we select centre then they but if we select centre then rotate the Game Objects, then they rotate about an arbitrary point between the two Game Objects.

A screenshot of a video game

Description automatically generated

A screenshot of a video game

Description automatically generated

Ok, once you ‘ve completed your editing and you want to test your application from the end users perspective. The way we do that is in the Game View, using the Play Controls

The Unity **Play Controls**, in the centre allow you to play, pause or step through frame by frame to playtest your game or application.

To the right of the Unity Play Controls we have the **Unity Collab, Cloud Service, and Account Service Controls** => these allow you to upload and download updates to shared projects and manage our Cloud Service Settings.

The Account drop down menu provides a shortcut to accessing our account settings on the Unity ID web page and allows you to sign in or out of our account within the editor.

The **Layers** drop down menu allows you to quickly show and hide the gameobject layers of content in our application/game.

The **Layout** drop down menu allows you to switch between different saved layouts for the Unity Interface.

Now, the Windows in the Unity Editor can be re-arranged to suit your personal preferences, and ease of use.

When your happy with the layout you can save the layout as a custom layout in the Layout drop down menu, so you can return the saved layout in the future.

# **Menu Bar - File**

The File option is concerned with creating/saving an existing scenes, projects, assets, and files.

Graphical user interface, text, application, chat or text message

Description automatically generated

**New Scene** creates a new level, NOT a new project. (A new project creates a whole new game or application) which you can add in your build settings etc.

**Open Scene** just opens the scene.

**Save Scene/Save Scene** as just says over your Scene. It does not automatically create a new scene.

**New Project** creates a new game, file, or application in Unity.

**Open Project** opens the project; and **Save Project** saves the project that’s self-explanatory.

**Build Settings** is the window where you render (or publish) your games. The Platforms are the devices you would be rendering the games for PC, Mac or Linux, Android, WebGL, IOS, Xbox etc.

**Scenes in Build** is where you add scenes to be pushed to the build the file (.exe or .apk). Under Scenes To Load is where you would access/load the Scenes from your script. Once you pushed to build, you cannot edit your game afterwards, they will work as standalone pieces of software.

# **Menu Bar - Edit**

Next to File is the edit option.

All the “hotkeys” relating to performing standard editing operations in your scene/game are located here like Cut, Copy, Paste, Duplicate, Delete etc. Play around with them; and you will get superfast, and you will be using Unity like a pro Lol !

**Load/Save Selection** command allows you save and load collections of game objects. So you can select game objects and then go up to Select and then Save Selection 1 and then we can Load this Selection by clicking on Load Selection 1 which automatically loads your previously saved Selection. This is useful, because it will save you time reselect a lot of gameObjects in your scene if you must select many gameObjects in your scene.

**Project Settings** option which will allow you to customize the settings for your Unity Editor regarding the current project your working on. For example if we click on Input it displays the Input manager which allows you to choose the mapping of keys for jumping, moving forward etc. The Audio option will allow you to choose the maximum volume of game sounds and so on. The Time command will allow you to choose the fixed time step for physics.

**Frame Selected** command allows you to focus on a particular game object. So it will center the view port on the selected game object. So you can either select the game object in the scene, or select it in the Hierarchy window and then move the pointer in the scene view and press F and you see it centers the focus on the game object

The **Select** command will select all objects so if you're in the Hierarchy window and you click on Select all in the Edit menu it will select all game objects. Similarly if you click on the Project window and click on Select all in the Edit menu it will select everything in the Project window. So Select will select all the content in the chosen window or panel.

The **Find** option is used to search and locate a game object. You find for game objects by typing in the search bar in the chosen panel.

The **Preferences** options allows you to edit your preferences of the Unity Editor. For example we can choose to compress assets on import, auto refresh content. There are other options for example if we go to **External Tools** we can choose what code editor should open our scripts by default. We can also customize the colors of our UI elements in the editor, using the shortcut keys.

We also have the **Play, Pause and Step** commands in the Edit dropdown menu. The Play command will allow us to Play the game, the Pause command will pause the game play and the Step command will allow us to see the game frame by frame which is useful for testing the game and debug it.

We can also **Sign in and Sign out** by clicking on these options, at current I am Signed in already so you see the Sign in option greyed out.

The **Physics** option allows you to control the physic settings of the game such as the gravity etc So Physics if for 3D games and Physics2D is for 2D games.

Next you have the **Quality settings** for the render graphic settings, shadow settings.

The **Graphics settings** allow you to customize the shaders, resolution quality and so on.

**The Graphics Tier** allows you to choose options to check out how your scene or game is rendered on different graphics cards. Different graphics cards have different shader models, so we can check how the game is rendered on these graphics’ cards.

The **Grid and Snap** Settings allow you to snap a GameObject to a grid projected along the X, Y, or Z axes.

Graphical user interface, text, application

Description automatically generated

# **Menu Bar - Assets**

The first option "Create" allows us to create a new script, scene, folder, material etc.

We can also perform the same actions when we right click in the project folder and navigate to the Create option.

If we click on Show in Explorer it will show us the assets folder in the windows file explorer.

We can also open and delete scenes, and individual folders present here in the project window.

Clicking on Copy path will copy the path to the current folder that we're in.

The most common options that you will use are the Import **New Asset, Export package and import package**.

If we click on **Import new assets** we can import audio files, 3D models, fonts, and other assets for us to use in our game. Let's try it out.

We can **import a package** into Unity and we can also export our game as a package with these 2 options. According to the official Unity documentation: Packages are collections of files and data from Unity projects, or elements of projects, which are compressed and stored in one file, like Zip files. Reimport will import all the packages to your project folder again. The open C# project option will allow you to open a new C# project or file.

Graphical user interface, text, application, email

Description automatically generated

# **Menu Bar - GameObject**

**Create Empty** and **Create Empty Child** is an invisible object, it has transform values, it is a single point without dimensions ! We mainly use it to create a parent group or folder to group similar gameObjects together in the Hierarchy View. Or we can use an empty GameObject to define a point in space for use as a waypoint.

**Make as first/last sibling**: this changes the order of the cube siblings as seen in the Hierarchy window so now Cube B is the elder sibling when we click on set as first sibling. Similarly if we click on set as second sibling while cube B is still selected, it will make it the younger sibling of cube A as seen by its placement in the Hierarchy window

**Move to View**: This is an alignment command. It will move the selected object to the center of the viewport. By selecting an object such as a cube from the Hierarchy window and then choosing Move to View from the dropdown menu we can move an object to the current view or center of the viewport where are able to see it. So, it physically moves the game object to the center of the scene window. It moves it to the focal point of your scene view.

This is very useful for moving GameObjects that might be extremely far in your scene to the center of your scene/viewport.

**Align with View**: This is another alignment command; it is like the Move to view command however in addition to moving the selected game object it will also rotate the object with respect to the current view. So it will move and rotate the selected object with respect to the position and rotation of the viewport. This is especially useful for moving a Camera into alignment.

**Toggle Active State**: turns GameObjects on and off. This can also be done with the first checkbox that appears in the Inspector window next to the game object’s name.

Graphical user interface

Description automatically generated with medium confidence

# **Menu Bar - Component**

The component menu is the “easy place” where we find various effects and systems you want to add additional functionality to the gameObjects in your scene.

**Physics** if you want to add a rigid body and add a source of gravity to your game Objects.

**Audio** if say you want to add an Audio clip or sound in your game.

**Rendering** is for adding new light, cameras to the scene

Table

Description automatically generated

# **Menu Bar - Window**

The **Panels** option will allow you to view these panels which are normally displayed in the Game editor for example Console, Game, Hierarchy, Inspector, Project, and Scene windows. You can also change the layout of the Unity editor to display as many panels as you want, so basically you can customize the way the Unity editor looks.

We have the **Collaborate** option here which opens here on the right side and will allow you to Collaborate using something called **Unity Collab.**

**Plastic SCM** is a full version control stack integrated with Unity.

The **Package manager** option will open the package manager window from where you can add and remove packages.

**Asset management** - version control allows you to manage the different assets in your project vi version control

**TextMesh Pro** is an easy-to-use system for high-quality text. It has many texts appearance and formatting options. So from here you can create fonts, import sprites for 2D games, import other tools from textmesh pro

The **General** option will allow you to see windows like Inspector, Scene, Console etc. You can see all these anyways in the Unity editor's default configuration

The **Rendering** option has all these different windows that will open for you to adjust the Lightning, Occlusion Culling and Light Explorer.

The **Occlusion Culling** is a feature that disables rendering of objects when they are not currently seen by the camera because they are obscured (occluded) by other objects. ... Occlusion culling removes additional objects from within the camera rendering work if they are entirely obscured by nearer objects.

The **Light Explorer** Window lets you see every Light in your Scene.

The **Animation** option will allow you to open different windows associated with creating and editing animations in your game. Similarly for the same for **Audio**, it will open an Audio mixed window

Then we have **Sequencing, Analysis, AI, and Toolkit** options which all let you open windows.

The **Sequencing** option will open a timeline window which lets you make cinematic sequences.

The **Navigation** window will allow you to open and manage any NavMeshes you have in your game for AI patrolling.

The **Analysis** option lets you analyse your game. When you make an application in Unity, it’s important that you analyze and debug it to make sure that it runs correctly on the target device. The Frame Debugger lets you freeze playback for a running game on a particular frame and view the individual draw calls that are used to render that frame. The physics debuggers allow you to inspect colliders and physics in your game. The Unity Profiler is a tool you can use to get performance information about your application.

The **UI Toolkit** option allows you to access collection of features, functionality, resources, and tools for developing user interfaces (UI).

Graphical user interface, text, application

Description automatically generated

# **Menu Bar - Help**

From the Help drop-down menu you can pull up the **Unity User Manual** or the **Unity Scripting Reference**.

There are other help options to check out the official **Unity forum**, **Unity answers** to get a specific question answered or look up older question-answers.

You can also check for Updates, look at the current Software licenses being used and manage the current software license for your version of Unity.

Additionally you can also give Unity feedback and **report a bug**.

**Premium Expert Help** is a new service by Unity and is in its Beta version so they're still testing and making it better. This allows you to reach out to a Unity expert and get help.

Graphical user interface, text, application

Description automatically generated

You use the PROJECT WINDOW to import/export assets like, materials, 3D models, textures, scripts, audio clips and so on to build our application to/from folders on the hard drive.

The Scenes we create in the Scene Window, are also saved and accessed via the Project Window as well.

# **Glossary**

# **Resources**

Scripting API: MonoBehaviour.StartCoroutine - Unity – Manual

<https://docs.unity3d.com/ScriptReference/MonoBehaviour.StartCoroutine.html>

/End