

1

Meet Unity

Welcome to *Building Levels in Unity*! This book will teach you how to build a complete level from start to finish in a step-by-step fashion. Designed for artists, modelers, animators, game designers, and people with no prior knowledge of game engines, you will find a lot of interesting information about editor functionality, asset import, character tuning, terrain creation, audio, lighting, and much more. Each chapter will focus on a set of topics, to build a solid foundation, and prepare you for more advanced material. By following the examples and using complementary download content, you will have a completely playable and tuned 3D environment.

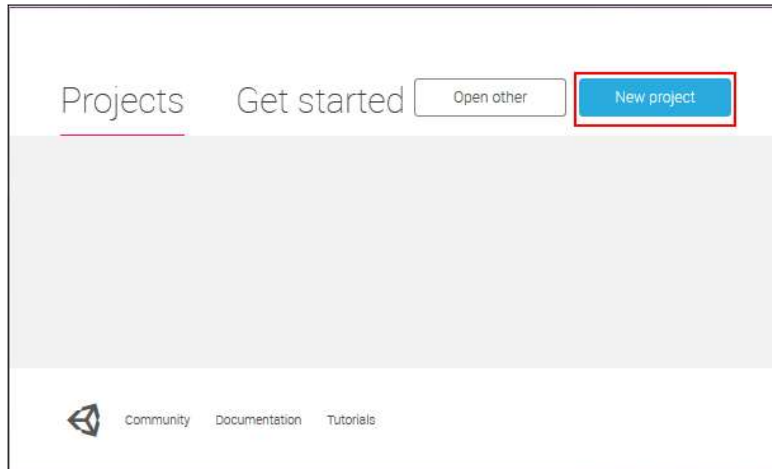
In this chapter, we will learn the following:

- Learn to freely navigate in the Unity editor
- Create our first project
- Learn about creation and use of built-in primitives
- Use and purpose of GameObjects and everything required to deal with them

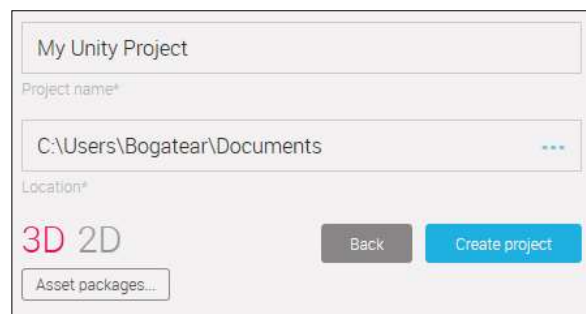
We will cover all the basics that are necessary for future learning and set up our project that will grow into a fully fleshed level. Let's begin!

Starting a project

I would assume that you've already downloaded Unity and finished the registration process. If you did, then by launching the program, you will be welcomed by the **Projects** screen where all your future projects will be located. Right now it is empty, so let's add one to it by clicking on the **New Project** button at the top-right corner.



This will open a new window that will ask you to enter the name of your project and its location on the hard drive.



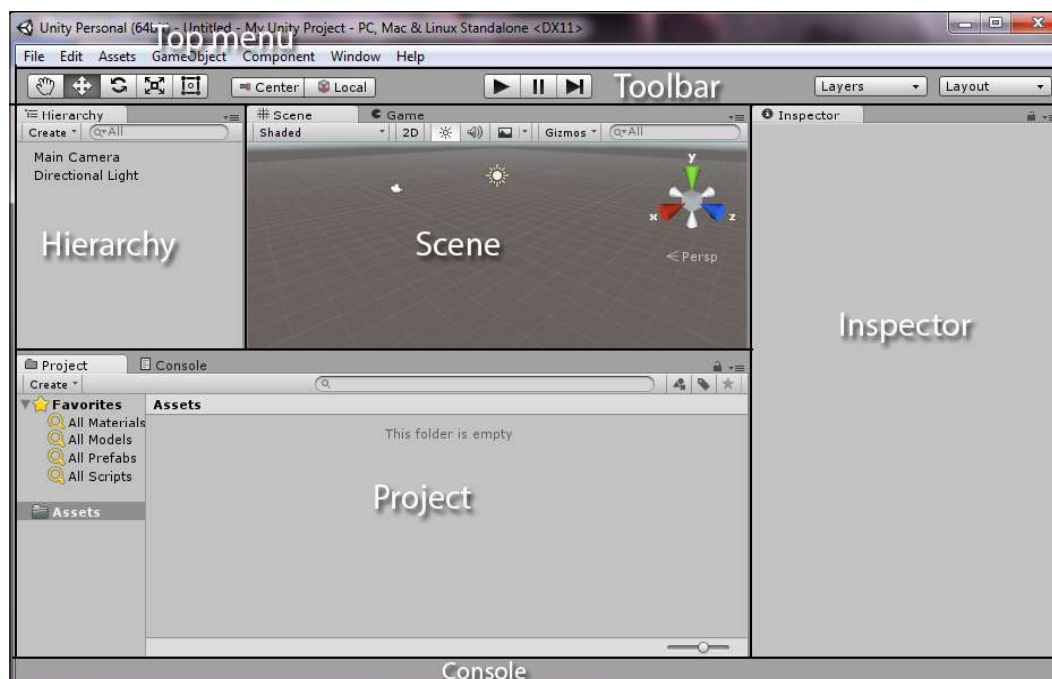
I'm going to skip the fancy naming and go with **My Unity Project** this time and leave the default **Location**. Make sure that you have the **3D** option selected, which will default our project to the 3D space and click on the **Create Project** button to proceed.

Don't worry about importing packages at this point; we can always do that later. If you already have a project, creating a new one is done by navigating to **File | New Project** in the top menu.

Windows

We are now going to take a quick look at the Unity editor and all its basic tools and features. You don't have to learn everything by heart at this point, especially if this is your first introduction to this engine; just taking mental notes about what's possible and what tools are available will suffice. We will look into, and study these tools in greater detail later in the book, so don't overwhelm yourself trying to remember everything.

Whenever you first open Unity, you will see a default layout like this:



There are a lot of windows on display. We will be using naming conventions assigned to the windows in the preceding screenshot in all the examples throughout the book, so make sure to reference it whenever you feel lost.

The Scene window

The Scene window allows you to place, select, and edit objects in the level. It's like a sandbox designed for quick and comfortable manipulations of level building. This is where you, as an artist/designer, will spend most of your time working.

Scene navigation

There are a number of ways to navigate in the scene window, some of them you may find familiar from using other apps, and some are unique to the Unity editor.

Arrow navigation

Keyboard navigation arrows can be used to simulate walking through the environment. Move forward and backward with the up and down arrow keys, or pan left and right with the left and right arrow keys. Holding the *Shift* key will significantly increase movement speed. You probably won't use it too often, but it's a nice feature to have nonetheless. In order for hotkeys to have an effect, you need to have the **Scene** window selected as active, to do so simply left-click inside of it.

Mouse navigation

The most common method for navigation revolves around the mouse. Here are a few control options:

- Hold the middle mouse button and drag to pan.
- Hold *Alt* + the left mouse button and drag to orbit around the center of the screen.
- Hold *Alt* + right mouse button and drag or scroll the mouse wheel to zoom in and out.
- Holding the **Shift** key will allow you to navigate faster.

You will notice the **Hand Tool** icon in the **Toolbar** changing whenever you switch between panning, orbiting, and zooming.



Flythrough

The Flythrough mode allows you to fly around your level from a first person perspective.

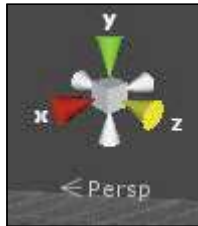
To enter the Flythrough mode, hold the right mouse button, and then use:

- The *WASD* keys to move forward, backward, and sideways
- The mouse to look around
- The *E* and *Q* buttons to go up and down

Holding the *Shift* key will allow you to fly faster. This navigation method is much more flexible than and similar to using arrow keys; however, it's still a lot more exotic than common.

Gizmo

Located in the upper-right corner of the Scene window, this allows you to quickly change camera orientation by clicking on one of the six cones, or switch between **Perspective mode** and **Isometric mode** by clicking on the text or cube in the middle.



Keep in mind that flythrough navigation is specifically designed for the Perspective mode. Gizmo is only available if you are working in the 3D mode.

The scene control bar allows you to choose various options for level viewing.

The drop-down menu (on the left) picks the draw mode for the scene.



The most notable three options of the Shading Mode are:

- **Shaded:** This is the default mode, which draws the object's surface with textures.
- **Wireframe:** This draws the object's wireframe.
- **Shaded Wireframe:** This combines both the previous modes with surfaces, textures and wireframes drawn.

Next are three switches that enable/disable 2D mode, lighting, and audio.

The effects button allows you to switch the following effects: skybox, fog, flares, animated materials. Clicking on a button will toggle them simultaneously; clicking on a drop-down menu will allow switching them individually.

The Gizmos menu allows you to control overlay graphics. You can toggle their visibility, control scale, and enable/disable the grid. This menu supports both built-in and custom gizmos created by programmers.



On the right is a search box allowing you to filter items by name or type. The searched objects will remain as is; however, all other objects will go gray scale and become semi-transparent, allowing filtered objects to stand out.



That's the basics of the Scene window. Most of these options are rarely used during development, but they do come in handy the closer you get to the polishing phase.

The Game window

The Game window shows the final representation of your level and how it will look in the game. Unlike the Scene window, you can set up custom cameras and show your game through them.

Play testing

In the middle of the toolbar, you will find three control buttons that will allow you to test your level in the game.



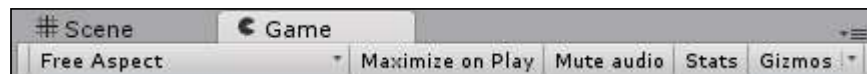
The buttons are **Play**, **Pause**, and **Step**. The first two are self-explanatory, the third one will play the game for a fraction of time before pausing and waiting for the next click.

Clicking on the **Play** button will activate the Game window, which will replace the Scene window.

You can switch between Scene and Game windows while testing, or placing them side by side to maximize efficiency, we will discover how to do that in future topics. That allows for a great amount of debugging flexibility; you can move objects, activate scripts, place enemies, all while the game is in progress. However, most changes made during the Play mode will be reset as soon as you stop testing, so be cautious when making those changes. There are useful plugins that you can download from the Unity Asset Store that will allow you to save changes made during the testing, whether to use them or not is completely up to personal preference.

Game window controls

The first drop-down menu controls the aspect ratio. You can use the available presets or create your own. This is especially useful if you are making a game for mobile devices.



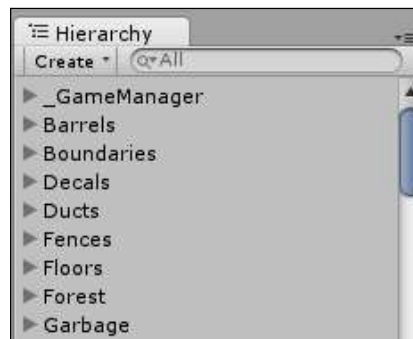
The following menus from the screenshot are explained as follows:

- **Maximize on Play:** This will toggle a fullscreen mode whenever you are play testing.
- **Mute audio:** This does exactly what you'd expect it to do.

- **Stats:** This will display rendering statistics, useful for reference when working on optimization.
- **Gizmos:** This menu is the same as the one in the Scene window. These options are very useful during play testing, they allow you to, for example, see triggers and raycasts.

The Hierarchy window

The **Hierarchy** window shows you the content of the level. All the objects that exist in the level will appear in the Hierarchy window.



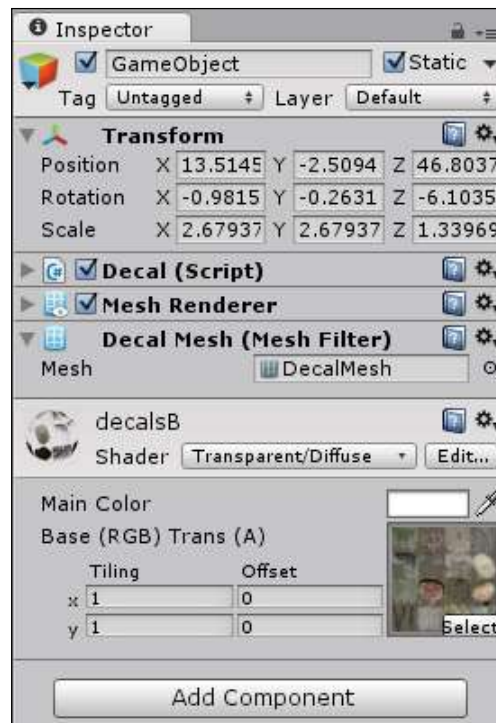
You can create new objects with a **Create** drop-down menu and quickly focus on them by:

1. Selecting objects in the **Hierarchy** window.
2. Moving the mouse cursor to the **Scene** window.
3. Pressing the *F* key.

This will allocate the camera and focus it on a selected object.

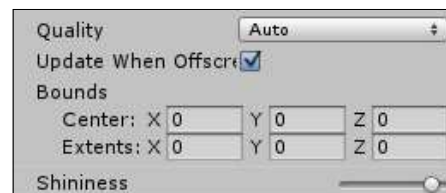
The Inspector window

Selecting objects in the Scene or Hierarchy window will allow you to add new Components, edit them and set up materials in the Inspector window. The Public script variables can also be edited without modifying the source code. All the operations regarding importing and some project-related settings will appear in the Inspector window as well.



The **Inspector** window contains different types of properties such as values, references, arrays, colors, gradients, and curves. Let's take a good look at the first four of them.

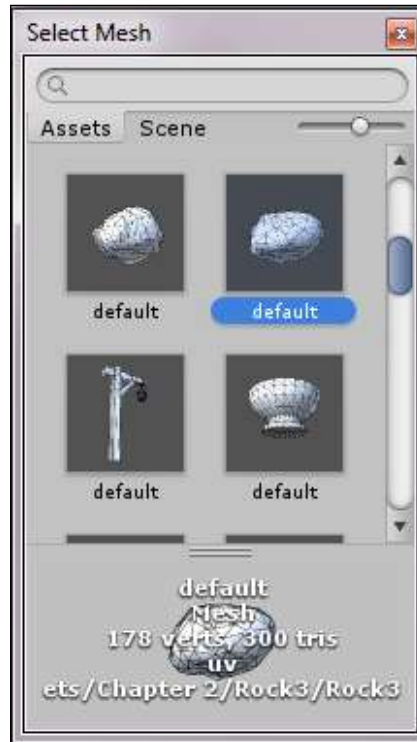
Values are the most commonly used property. It can be anything: drop-down menu, checkbox, text field or even a slider. They are no different here in Unity than they are in any other application, just your standard numbers, Booleans, multi choices and strings.



References, on the other hand, are slightly trickier. They require a reference of an object or a Component from the Project, Hierarchy, or Scene windows.

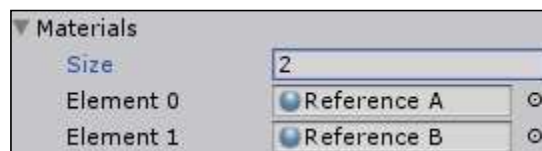


You can assign a reference by dragging and dropping the object into the reference field or using a circle on the right side to open an object picker.



It's important to know that if the required reference type is a Component, you can sometimes reference an object that has that Component attached to it. The reference will be assigned to a first Component of that type. We will talk about the Components later in this chapter.

One thing you need to be aware of when dealing with arrays is how to control their **Size**. By incrementing the value, you will create more elements that will inherit properties from the last element of the array. However, decrementing the value will delete elements for good, so when you increment the size back, new elements will gain the value of the last element, not the value they used to have before decrementing.

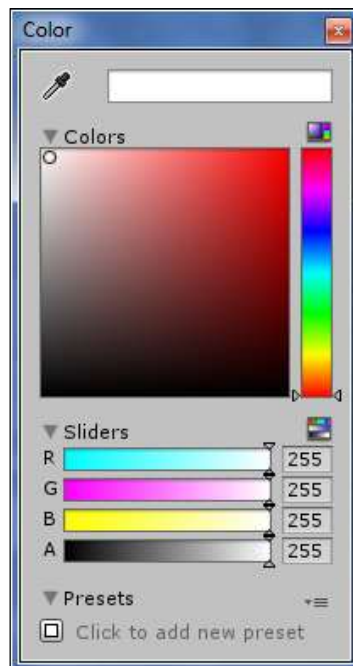


Based on the preceding screenshot, we can safely state that incrementing the **Size** parameter to 3 will create **Element 2** with assigned **Reference B** reference just like its predecessor **Element 1**. On the other hand, if we first decrement the **Size** to 1 and then increment it back to 2, **Element 1** reference will be changed from **Reference B** to **Reference A**.

Color is your usual RGB value. Clicking on the eyedropper tool will allow you to pick a color from the screen by hovering and left clicking.



Alternatively, you may select the color by opening the **Color Picker** tool and clicking on the color field. Here you can also create your own libraries of presets for future use.



The latter two property types are not as common and we will look into them when they become relevant.

The Project window

This is similar to the Hierarchy window, however, it contains all the assets that exist within the project and can be used in the project.



The default Project window layout is a new feature added with the 4th version of Unity. It's separated into two sections. The left side will show you a folder hierarchy and a list of **Favorites** (search inquiries). The right side will display previews of objects in the folder as well as the folder's location from the root, at the top.

From here you can select, organize, drag and drop assets into the Scene, Hierarchy, Inspector windows, or create new entities by clicking on the **Create** drop-down menu in the top-left corner.

Importing custom packages into the project

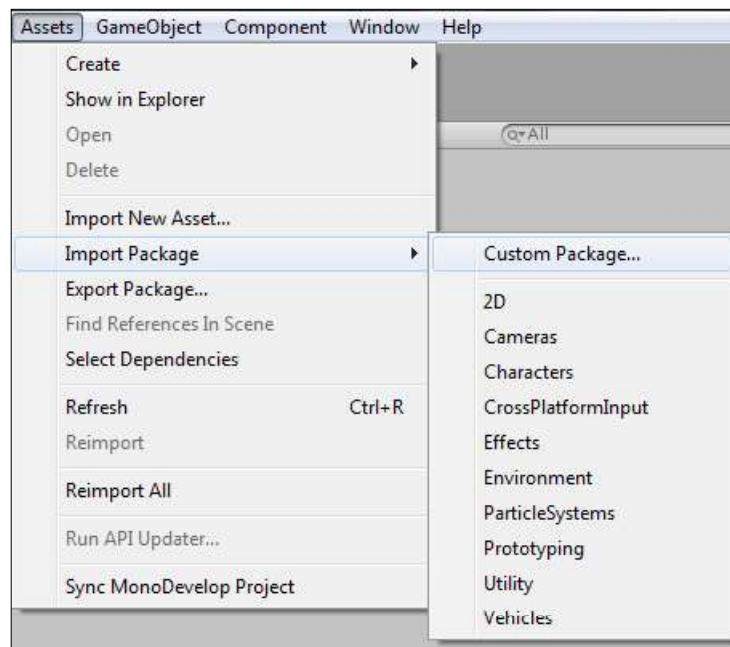
In order to continue working with this book's content, you need to learn how to import custom packages in Unity.

Packages are essentially a set of assets compressed into a single file. Packages help transfer data quickly and efficiently between different Unity projects or computers.

To import the package, do the following:

1. Right-click in the blank space of the **Project** window, alternatively, select **Assets** from the top menu.
2. Navigate to **Import Package | Custom Package**.
3. Navigate to the book files saved on your computer.
4. Select the **BuildingLevelsWithUnity.unitypackage** file and click on **Open**.

This will open the window in which you will be able to select files you want, or don't want, to import and then click on import to finalize the process.



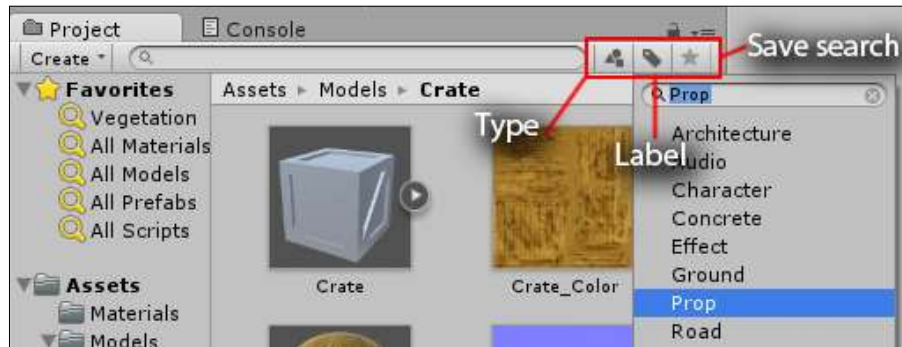
Now you know how to import packages to keep up with the book's material.

Search

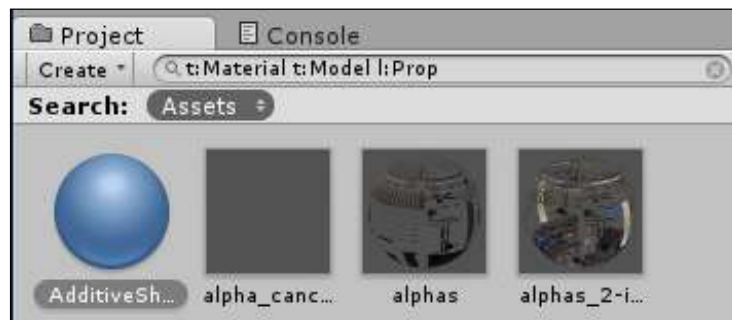
You can sort the assets in the folders or use search to locate them by name, type, or labels:

1. Typing in the keywords in a search field will allow you to filter objects that have these keywords in them.
2. Click on a **Type** icon located on the right side of the search field to search by type.

3. Click on a **Label** icon to search by label.
4. Click on the **Save** button to save the search query to the **Favorites** list on the left.



However, the search in Unity is a lot more powerful than that. You can combine multiple search queries of type and/or label by holding the *Ctrl* button or simply type them in a search field (t:typename for type, l:labelname for labels, searching by keyword works by inputting the keyword).



We will look more into Types and Labels in *Chapter 2, Importing and Configuring Props*.

Creating a custom search

Give this search functionality a try, and find objects that meet the following conditions:

1. Type: Texture and Model.
2. Label: Vegetation.
3. Keyword: Bush.

Give it a try and save the result as `Bushes`.

Rename

To rename an object in the Hierarchy or Project window, do the following:

1. Left-click on the entity.
2. Left-click again in the name area or press the *F2* key.

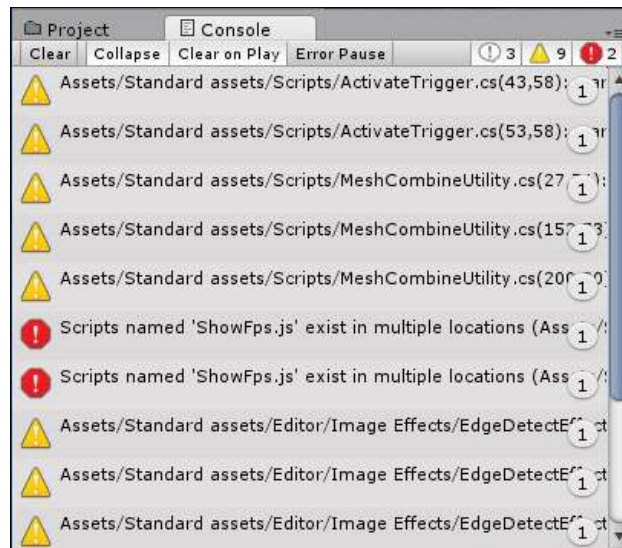


Do not double-click!

This is something you will have to get used to – double clicking will not enable renaming as you would expect, but may prompt to open the asset in the external application (for example, in the **Project** window, if you try to double-click on something like a texture file).

The Console window

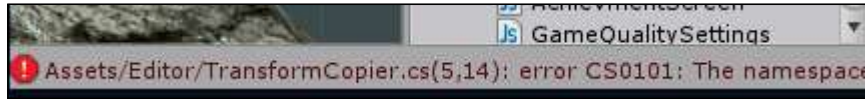
The console will display a list of warnings, errors, and messages as they appear (mostly during play testing), that are useful to pay attention to because Unity can warn you about all sorts of things, not only code related.



As you may notice, the Console window has multiple options available:

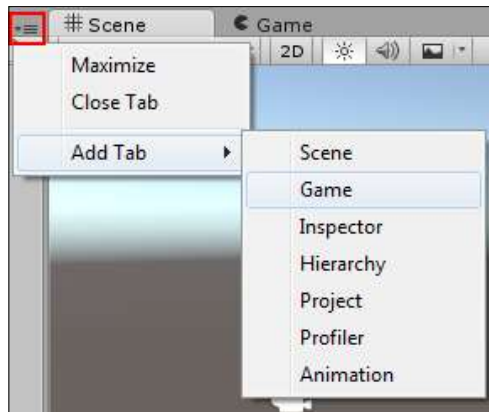
- **Clear:** This will clear all current logs
- **Collapse:** This will hide repeating logs
- **Clear On Play:** This will apply a **Clear** command every time you enter **Play** mode. It is recommended to always have it on.
- **Error Pause:** This will initiate Pause mode whenever the Error log appears

There is no need to have the Console window open all the time. The bottom of the screen has an area that will display the latest log in it whenever it appears. Double clicking on it will open the Console window.

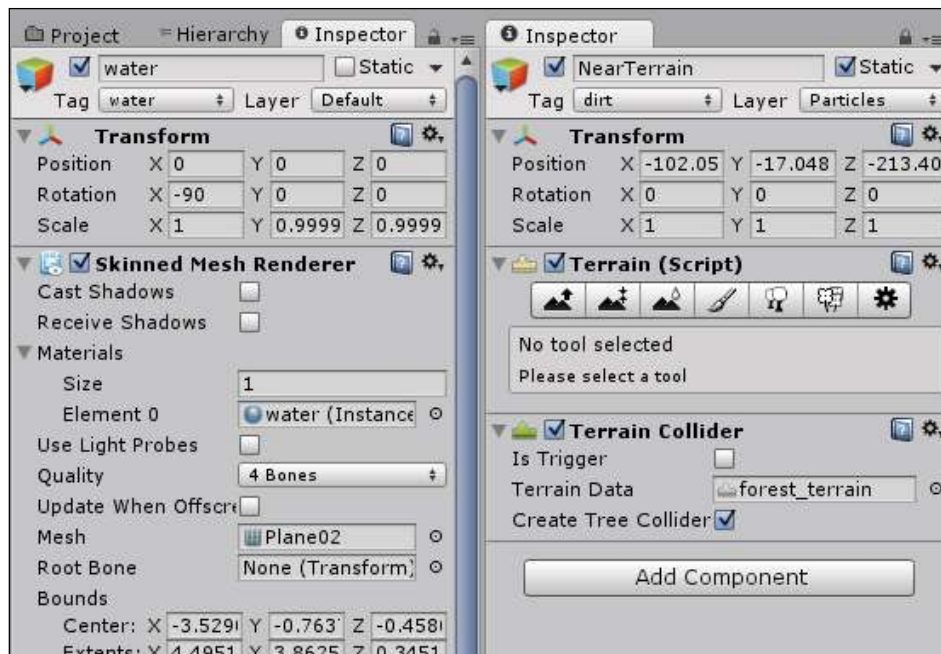


Window controls

Windows aren't limited to just one instance. You can create as many copies as you like by clicking on a drop-down menu at the top-right corner of each window and selecting **Add Tab**.



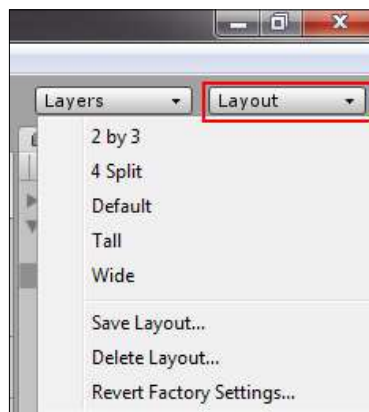
This is very useful, since windows like **Project** and **Inspector** have a **Lock** function (Lock sign next to Add Tab) allowing it to keep the current selection and enabling you to work with two objects at the same time.



Changing the window layout

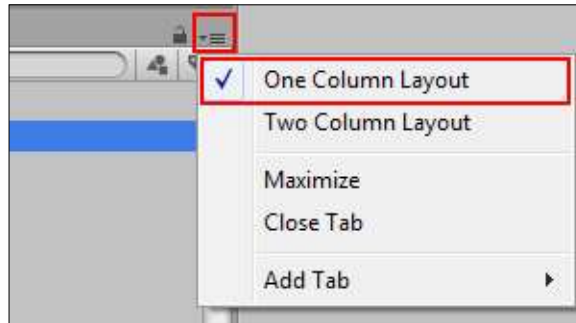
If you don't like the default window layout, you can change it anytime by dragging windows to different Docking Zones within the editor or drag them outside completely, to float.

There are a few layout presets that you may find useful. They are available under the **Layout** drop-down menu in the top-right corner of the editor or under **Window | Layouts** of the top menu.

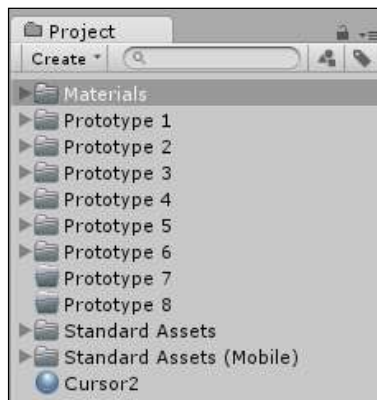


If you wish to switch to a simpler and more compact version:

1. Click on the drop-down menu in the rightmost top corner of the **Project** window.
2. Select **One Column Layout**.



If you've been using Unity prior to the 4th version, you should now find yourself at home.



The only real downside of using **One Column Layout** is a lack of saved search inquiries.

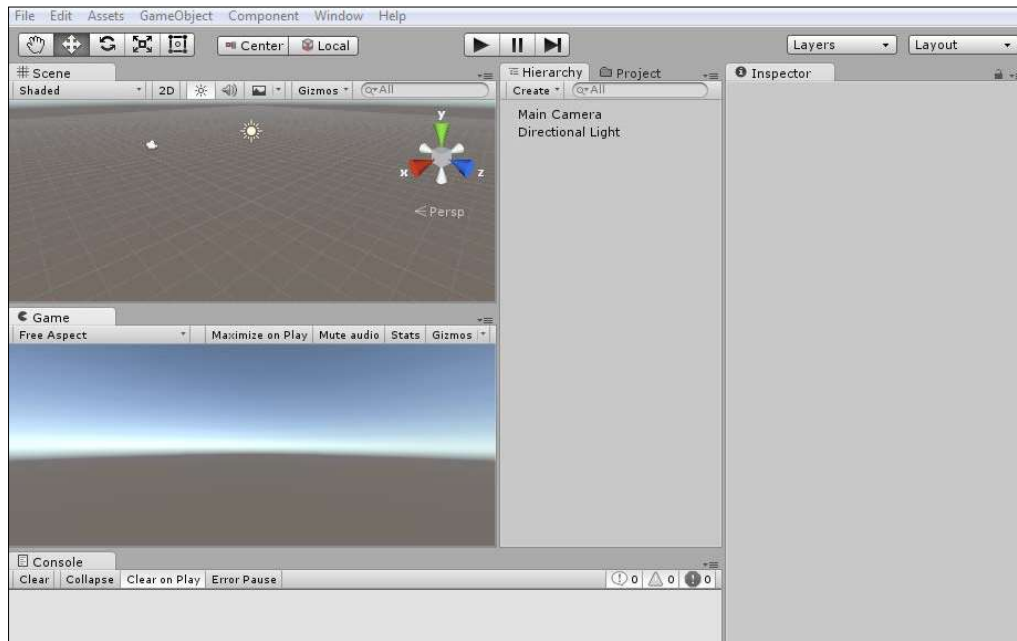


From here on, we will be using the **Tall** preset layout with Project window set to **One column layout** (the drop-down menu at the top-right corner of the Project window) to optimize our screen space, but you can choose whichever one you like, this is a preference call.

Creating a custom layout

Just like with the search functionality, it would be a nice idea to solidify this knowledge with a practical example:

1. Revert to default layout by navigating to **Layout | Revert Factory Settings...** in the toolbar, or **Window | Layouts | Revert Factory Settings...** in the top menu.
2. Recreate the layout presented on the following screenshot:



In order for this to work, you'll need to drag the windows around the editor in search of the invisible Docking Zones.

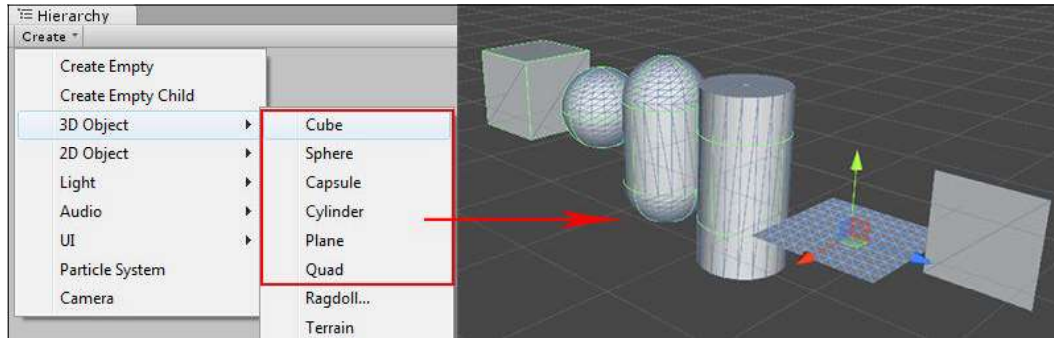
When you've customized the editor, save it by going to **Layout | Save Layout**.

The purpose of a Main Camera object

Cameras are unique GameObjects that serve as our eyes and ears in the game and the editor. By default there are two cameras—the Editor Camera and a Main Camera. The Editor Camera allows us to see objects in the Scene window and the Main Camera is the one we will be using during play testing. Sure, we can have as many cameras as we want and there are a lot of settings that we can play with, but at this point, it's enough to just be aware of it.

Creating primitives

Unity comes with its own set of basic primitives that can be created and edited for quick prototyping or greyblocking. There are **Cubes**, **Spheres**, **Capsules**, **Cylinders**, **Planes**, and **Quads**. They are available under **GameObject | 3D Object** in the top menu or under the **Create** menu of a Hierarchy window.



The created object will immediately appear, both in the Scene and inside of the Hierarchy window.

Summary

We've covered all the basics that newcomers need to know in order to be comfortable in Unity. Not all, but most of these features, tools, and functions will be used in the future chapters of this book. Whenever you find yourself lost feel free to return to this chapter and review that specific topic. If you need more information, use the official Unity documentation under **Help | Unity Manual** in the top menu.

In the next chapter, we will start importing assets and configuring them to be used in our level. We will discuss how to export your assets from other apps and how to configure them in Unity, and also set up materials and LODs.