



Learn and create

Unity 5

From Zero to Proficiency (Foundations)

A step-by-step guide to creating your first game

Patrick Felicia

<http://freepdf-books.com>

A step-by-step guide to creating your first game.

Patrick Felicia

Unity 5 From Zero to Proficiency (Foundations)

Copyright © 2015 Patrick Felicia

All rights reserved. No part of this book may be reproduced, stored in retrieval systems, or transmitted in any form or by any means, without the prior written permission of the publisher (Patrick Felicia), except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either expressed or implied. Neither the author and its dealers and distributors will be held liable for any damages caused or alleged to be caused directly or indirectly by this book.

First published: October 2015

Published by Patrick Felicia

Credits

Author: Patrick Felicia

About the Author

Patrick Felicia is a lecturer and researcher at Waterford Institute of Technology, where he teaches and supervises undergraduate and postgraduate students. He obtained his MSc in Multimedia Technology in 2003 and PhD in Computer Science in 2009 from University College Cork, Ireland. He has published several books and articles on the use of video games for educational purposes, including the Handbook of Research on Improving Learning and Motivation through Educational Games: Multidisciplinary Approaches (published by IGI), and Digital Games in Schools: a Handbook for Teachers, published by European Schoolnet. Patrick is also the Editor-in-chief of the International Journal of Game-Based Learning (IJGBL), and the Conference Director of the Irish Symposium on Game-Based Learning, a popular conference on games and learning organized throughout Ireland.

Support and Resources for this Book

So that you can complete the project presented in this book seamlessly, a website has been setup, and it includes all the material you need to complete the project presented in the next chapters (e.g., textures or solutions for each chapter), as well as bonus material.

To avail of this content, you can open the link:

<http://www.learntocreategames.com/learn-unity-ebook/>.

<http://www.learntocreategames.com> also provides you with the opportunity to subscribe to a newsletter, obtain exclusive discounts and offers on new books, and gain access to video tutorials on creating games.

Why should you subscribe?

- Be the first to be notified of new resources available.
- Receive regular updates and tutorials on creating games.
- Receive a newsletter with tips and hints on game development.

This book is dedicated to Helena

Preface

After teaching Unity for over 4 years, I always thought it could be great to find a book that could get my students started on Unity in a few hours and that showed them how to master the core functionalities offered by this fantastic software.

Many of the books that I found were too short and did not provide enough details on the why behind the actions recommended and taken; other books were highly theoretical, and I found they lacked practicality and would not get my students' full attention. In addition, I often found that game development may be preferred by those with a programming background but that those with an Arts background, even if they wanted to get to know how to create games, often had to face the issue of learning to code for the first time.

As a result, I started to consider a format that would cover both: be approachable (even to the students with no programming background), keep students highly motivated and involved using an interesting project, cover the core functionalities available in Unity to get started on game programming, provide answers to common questions, and also provide, if need be, a considerable amount of details for some topics.

This book series entitled From Zero to Proficiency does just this. In this book series, you have the opportunity to play around with Unity's core features, and essentially those that will make it possible to create an interesting 3D game rapidly. After reading this book series, you should find it easier to use Unity and its core functionalities.

This book series assumes no prior knowledge on the part of the reader, and it will get you started on Unity so that you quickly master all the wonderful features that this software provides by going through an easy learning curve. By completing each chapter, and following step-by-step instructions, you will progressively improve your skills, become more proficient in Unity, and create a survival game using Unity's core features in terms of programming (C# and JavaScript), game design, and drag and drop features.

In addition to understanding and being able to master Unity's core features, you will also create a game that includes many of the common techniques found in video games, including: level design, object creation, textures, collision detection, lights, weapon creation, character animations, particles, artificial intelligence, and menus.

Throughout this book series, you will create a game that includes both indoor and outdoor environments where the player needs to find its way out of the former through tunnels, escalators, traps, and other challenges, avoid or eliminate enemies using weapons (i.e., gun or grenades), drive a car or pilot an aircraft.

You will learn how to create customized menus and simple user interfaces using Unity's new UI system, and animate and give (artificial) intelligence to Non-Player Characters (NPCs) who will be able to follow your character using Mecanim and NavMesh navigation.

Finally you will also get to export your game for the web at the different stages of the books, so that you can share it with friends and get some feedback.

Content Covered by this Book

Chapter 1, The Benefits of Using Unity, provides general information on game engines and explains why you should use such software, and how, by using Unity more specifically, you can create games seamlessly.

Chapter 2, Installing Unity and Becoming Familiar with the Interface, takes you through the very first steps of installing Unity and becoming familiar with the interface. It will also show you the different shortcuts necessary to navigate through scenes and projects in Unity.

Chapter 3, Creating and Exporting your First Scene, gets you to create and export your first scene by combining built-in objects. You will learn how to manage objects, apply textures and colors, and transform objects to create a simple scene.

Chapter 4, Transforming Built-in Objects to Create an Indoor Scene, explains how you can create an indoor scene (i.e., a maze) with built-in shapes. You will also work with and manage lights in your scene to set the atmosphere and navigate through the scene with a First-Person Controller.

Chapter 5, Creating an outdoor Scene with Unity's Built-in Terrain Generator, explains how you can create an outdoor scene with water, hills, sandy beaches and palm trees using Unity's built-in assets. You will also be able to add and control a car and a plane.

Chapter 6 provides answers to frequently asked questions based on specific themes and topics (e.g., assets creation or transformations).

Chapter 7 summarizes the topics covered in this book and also provides useful information if you would like to progress further with this book series.

What you Need to Use this Book

To complete the project presented in this book, you only need Unity 5.0 (or a more recent version) and to also ensure that your computer and its operating system comply with Unity's requirements. Unity can be downloaded from the official website (<http://www.unity3d.com/download>), and before downloading, you can check that your computer is up to scratch on the following page: <http://www.unity3d.com/unity/system-requirements>. At the time of writing this book, the following operating systems are supported by Unity for development: Windows XP (i.e., SP2+, 7 SP1+), Windows 8, and Mac OS X 10.6+. In terms of graphics card, most cards produced after 2004 should be suitable.

In terms of computer skills, all knowledge introduced in this book will assume no prior programming experience from the reader. This book does not include any scripting (this will be introduced in the second book in the series). So for now, you only need to be able to perform common computer tasks such as downloading items, opening and saving files, and be comfortable with dragging and dropping items and typing.

Who this Book is for

If you can answer yes to all these questions, then this book is for you:

1. Are you a total beginner in Unity or programming?
2. Would you like to become proficient in the core functionalities offered by Unity?
3. Would you like to teach students or help your child to understand how to create games?
4. Would you like to start creating great games?
5. Although you may have had some prior exposure to Unity, would you like to delve more into Unity and understand its core functionalities in more detail?

Who this Book is not for

If you can answer yes to all these questions, then this book is **not** for you:

1. Can you already easily create a 3D game with Unity with built-in objects, controllers, cameras, lights, and terrains?
2. Are you looking for a reference book on Unity programming?
3. Are you an experienced (or at least advanced) Unity user?

If you can answer yes to all three questions, you may instead look for the next books in the series. To see the content and topics covered by these books, you can check the official website (www.learntocreategames.com/learn-unity-ebook).

How you will Learn from this Book

Because all students learn differently and have different expectations of a course, this book is designed to ensure that all readers find a learning mode that suits them. Therefore, it includes the following:

- A list of the learning objectives at the start of each chapter so that readers have a snapshot of the skills that will be covered.
- Each section includes an overview of the activities covered.
- Many of the activities are step-by-step, and learners are also given the opportunity to engage in deeper learning and problem-solving skills through the challenges offered at the end of each chapter.
- Each chapter ends-up with a quiz and challenges through which you can put your skills (and knowledge acquired) into practice, and see how much you know.
- The book focuses on the core skills that you need; some sections also go into more detail; however, once concepts have been explained, links are provided to additional resources, where necessary.

Format of each Chapter and Writing Conventions

Throughout this book, and to make reading and learning easier, text formatting and icons will be used to highlight parts of the information provided and make it more readable.

The full solution for the project presented in this book is available for download on the official website (<http://www.learntocreategames.com/learn-unity-ebook>). So if you need to skip a section, you can do so; you can also download the solution for the previous chapter that you have skipped.

Special Notes

Each chapter includes resource sections so that you can further your understanding and mastery of Unity; these include:

- A quiz for each chapter: these quizzes usually include 10 questions that test your knowledge of the topics covered throughout the chapter. The solutions are provided on the companion website.
- A checklist: it consists of between 5 and 10 key concepts and skills that you need to be comfortable with before progressing to the next chapter.
- Challenges: each chapter includes a challenge section where you are asked to combine your skills to solve a particular problem.

Author's notes appear as described below:

Author's suggestions appear in this box.

Checklists that include the important points covered in the chapter appear as described below:



- Item1 for check list
- Item2 for check list
- Item3 for check list

How Can You Learn Best from this Book

- **Talk to your friends about what you are doing.**

We often think that we understand a topic until we have to explain it to friends and answer their questions. By explaining your different projects, what you just learned will become clearer to you.

- **Do the exercises.**

All chapters include exercises that will help you to learn by doing. In other words, by completing these exercises, you will be able to better understand the topic and gain practical skills (i.e., rather than just reading).

- **Don't be afraid of making mistakes.**

I usually tell my students that making mistakes is part of the learning process; the more mistakes you make and the more opportunities you have for learning. At the start, you may find the errors disconcerting, or that the engine does not work as expected until you understand what went wrong.

- **Export your games early.**

It is always great to build and export your first game. Even if it is rather simple, it is always good to see it in a browser and to be able to share it with you friends.

- **Learn in chunks.**

It may be disconcerting to go through five or six chapters straight, as it may lower your motivation. Instead, give yourself enough time to learn, go at your own pace, and learn in small units (e.g., between 15 and 20 minutes per day). This will do at least two things for you: it will give your brain the time to “digest” the information that you have just learned, so that you can start fresh the following day. It will also make sure that you don't “burn-out” and that you keep your motivation levels high.

Feedback

While I have done everything possible to produce a book of high quality and value, I always appreciate feedback from readers so that the book can be improved accordingly. If you would like to give feedback, you can email me at learntocreategames@gmail.com.

Downloading the Solutions for the Book

You can download the solutions for this book after creating a free online account at www.learntocreategames.com/learn-unity-ebook. Once you have registered, a link to the files will be sent to you automatically.

Improving the Book

Although great care was taken in checking the content of this book, I am human, and some errors could remain in the book. As a result, it would be great if you could let me know of any issue or error you may have come across in this book, so that it can be solved and the book updated accordingly. To report an error, you can email me (learntocreategames@gmail.com) with the following information:

- Name of the book.
- The page where the error was detected.
- Describe the error and also what you think the correction should be.

Once your email is received, the error will be checked, and, in the case of a valid error, it will be corrected and the book page will be updated to reflect the changes accordingly.

Supporting the Author

A lot of work has gone into this book and it is the fruit of long hours of preparation, brainstorming, and finally writing. As a result, I would ask that you do not distribute any illegal copies of this book.

This means that if a friend wants a copy of this book, s/he will have to buy it through the official channels (i.e., through Amazon, lulu.com, or the book's official website:

www.learntocreategames.com/learn-unity-ebook).

If some of your friends are interested in the book, you can refer them to the book's official website (<http://www.learntocreategames.com/learn-unity-ebook>) where they can either buy the book, enter a monthly draw to be in for a chance of receiving a free copy of the book, or to be notified of future promotional offers.

1

The Benefits of Using Unity

This chapter is an introduction to game engines and Unity, and it explains the benefits brought by game engines, and more specifically how Unity can help you to create games seamlessly. The most recent features are explained, and examples of games created in Unity are also given so that you can evaluate the potential of this game engine.

If you already know of the benefits of Unity and game engines in general, you can skip to the next chapter.

After completing this section, you should be able to:

- Understand the concept of game engines.
- Know the features introduced by Unity 5.
- Understand the benefits of using Unity.

What is a game engine and should you use one?

Unity is a software that makes it possible to create video games without knowing some of the underlying technologies of game development, so that potential game developers only need to focus on the game mechanics and employ a high-level approach to creating games using programming and scripting languages such as C# or JavaScript. The term high-level here refers to the fact that, when you create a game with a game engine, you don't need to worry about how the software will render the game or how it will communicate with the graphics card to optimize the speed of your game. So using a game engine would generally offer the following features and benefits:

- **Accelerated development:** game engines make it possible to focus on the game mechanics. Because built-in libraries are available for common mechanics and features, these do not need to be rebuilt from scratch, and programmers can use them straightaway and save time (e.g., for the user interface or the artificial intelligence).
- **Integrated Development Environment (IDE):** an IDE helps to create, compile, and manage your code, and includes some useful tools that make development and debugging more efficient.
- **Graphical User Interface (GUI):** while some game engines are based on libraries, most common game engines make it possible for users to create objects seamlessly and to perform common tasks such as transforming, texturing, and animating, through drag and drop features. Another advantage of such software is that you can understand and preview how the game will look without having to compile the code (e.g., through scenes).
- **Multi-platform deployment:** with common game engines, it is possible to export the game you have created to several platforms with the click of a button (e.g., for the web, iOS, or Android) without having to recode the entire game.

Advantages of using Unity

There are several game engines available out there; however, Unity has proven to be one of the best engines. It has been used by game developers for several years and has been employed to produce successful 3D and 2D games. Several of these titles can be seen on Unity's website (<http://unity3d.com/gallery/made-with-unity/game-list>).

With Unity, you can create 2D or 3D games and produce several types of game genres including First-Person Shooters (FPS), Massive Multiplayer Online Role Playing Games (MMORPG), casual games, adventure games, and much more.

In addition to being able to create high-quality games with an easy-to-use interface, Unity makes it possible to export games to a wide range of platforms, including Android, iOS, Windows Phone 8, Mac, Linux, PS3, or XBOX360.

Unity includes all the necessary tools that you need to create great games and it also simplifies the application of useful techniques to improve the quality of your game. For example, it includes Mono Develop, an IDE that will help you to code faster, built-in Artificial Intelligence (AI) modules (e.g., navmesh navigation) that you can use with no prior knowledge of AI, lights, built-in objects, or a finite state machine that you can apply to your characters for customized behaviors.

Finally, in order to control the game, you can use high-level programming and scripting languages such as Boo, C# or JavaScript. This is useful for those who have already been exposed to one of these languages to transfer their skills to game programming in Unity. It also provides programmers with a choice depending on their level of proficiency. For example, beginners may prefer JavaScript, while more advanced coders may prefer to use C# (i.e., for those with prior experience of Object Oriented Programming).

Novelties introduced in Unity 4.6 and 5.0

Currently, Unity is in version 5. While the Unity team is consistently working hard to improve the features and functionalities included within, the software has been through a steady pace of changes and improvements since its first launch. Bugs are being fixed quickly and the Unity team is always looking into making this software easier to use and more efficient.

While subsequent versions will, without a doubt, introduce interesting new features, most of the skills and knowledge that you will acquire in this book should still be relevant.

Unity 4 introduced much-awaited exciting features such as Mecanim (i.e., for character animations). Unity 4.6 also marked an important step for users as it included Unity UI, a feature that makes it possible to create smoother, dynamic and more intuitive Graphical User Interfaces (GUI) for your game. Meanwhile, amongst other things, previously premium features found in version 4.x (e.g., navmesh and iOS or Android export) are now available for free in Unity 5 personal edition.

Unity 5.0 was released in 2015 and has brought significant changes in both the way the software is licensed and its features. From Unity 5.0 onwards, the software comes into two main versions: the personal edition and the professional edition. For the former, all the features of the engine are available. In the latter, you gain access to additional features such as customizable splash screen, team licenses, or game performance reporting. In terms of features, Unity 5.0 introduces a wide range of features that will make your game look more realistic and polished, including global illumination (for improved illumination when using both static and dynamic objects), the Audio Mixer or export to WebGL.

As you can see, there is much to learn in Unity and we will focus on Unity's core technologies in this book series.

Level roundup

Summary

This chapter has described some of the reasons why you should use Unity and some of its core functionalities. You have also discovered the concept of game engine, the benefits brought by game engines, and how Unity can specifically make it easier for you to get started with game development.

2

Installing Unity and Becoming Familiar with the Interface

This chapter helps you to progressively become familiar with Unity by explaining and illustrating how to install this software, and how the different views and core features can be employed. You will also learn to create your first project and scene, using predefined objects such as boxes. After learning the features of the different views available in Unity, you will learn how to navigate through a scene (to look at objects), before creating your very first game with built-in objects and applying colors and textures.

After completing this section, you should be able to:

- Be more comfortable with Unity's interface.
- Understand the role and location of the different views in Unity.
- Understand the role of colliders.
- Add and configure cameras and lights.
- Know and use shortcuts to manipulate objects (e.g., move, scale, resize, duplicate, or delete) and move the view accordingly (e.g., pan or rotate).
- Use the Inspector view.
- Create and apply colors and textures to objects.
- Create and combine simple built-in shapes.
- Know how to search for and organize assets in your game efficiently.
- Navigate through your scene and see it from both first- and third-person views.

Downloading Unity

Now that you have had an overview of Unity and game engines, it is time for us to start using Unity. Unity is available for download, for free, from the Unity website and the next steps will show you how to download it:

1. Open the following link: <http://unity3d.com/unity/system-requirements>. This will help you to check that your computer complies with Unity's requirements.
2. Once you have checked the requirements, we can download Unity by opening the following page: <http://unity3d.com/get-unity/download?ref=personal>
3. Once the page is opened, a link to the current version of Unity is provided (i.e., the installer). This page will automatically detect whether your computer is running Mac OS X or Windows, and by clicking on the link, the corresponding installer will be downloaded to your computer (i.e., .dmg for MacOSX or .exe for Windows), as described on the next figure. Note that direct links to either version are available in Unity's archive (<http://unity3d.com/get-unity/download/archive>).



Figure 2-1: Downloading Unity from the official website

It is possible to download previous versions of Unity from Unity's archive (<http://unity3d.com/get-unity/download/archive>). This may be useful if you work from two different locations for your Unity project and if each computer has a different version of Unity installed. If this is the case, you will be able to open projects built with an older version of Unity with a new version of Unity (e.g., a project initially created with Unity 4.x can be opened with Unity 5.x); however, after opening a project that was created with an older version of Unity, this project is converted to the new version of Unity and can no longer be opened with the previous version (i.e., a project initially created with Unity 4.x, and then opened with Unity 5.x can't be opened anymore with Unity 4.x). You may also backup your project before any file conversion, for safety.

Once we have downloaded Unity, we can launch the installer and follow the onscreen instructions. Once the software is installed, you can progress to the next section.

Launching Unity

Once you have successfully installed Unity, we can now launch it. Upon the first time you open Unity, you may need to provide your email address, so that you can receive regular updates from the Unity team. This should be really useful to keep up-to-date with major announcements for this software. You may also be asked whether you would like to activate the Pro version; however, for the purpose of this tutorial, you only need to use the free version (i.e., personal edition).

After having provided your email details as well as choosing the free version of the software, we can start to enjoy Unity.

After launching Unity, the following window appears:



Figure 2-2: Launching Unity for the first time

The previous figure shows a window labeled Projects that lists all projects that you have been working on previously (i.e., empty for now). If you click on the label Get Started, you can access a welcome video from the Unity Team that briefly describes the new features introduced by this version. You may also notice three buttons at the bottom of the window, labeled Community, Documentation, and Tutorials, which will take you to the relevant resources on the Unity website also. (i.e., Unity forums, the official documentation for Unity, or official Unity tutorials).

Let's press the button labeled New Project to create a new project. The following window should appear:



Figure 2-3: Creating a new project

In this window, we will specify a name for our project, as well as a location.

- In the section Project Name: type a name of your choice, for example myFirstProject.
- In the Location section: click on the three dots located to the right of the label Location and select where on your hard drive you would like to save the project.
- 2D/3D: for this project, we will be using a three-dimensional environment; therefore, we will click on the 3D icon. As we will see later, you can create both 2D and 3D games with Unity, and projects are setup accordingly.
- Assets/Packages: we will leave this option as default, as we will identify later the assets that will be necessary for our first project.
- Once you have entered this information, you can now click on the button labeled Create Project.

When Unity starts-up, a window labeled Unity Editor Update Check appears. This window, illustrated below, is there to check whether you have the latest version of Unity and to let you know of any recent updates available. If an update is necessary, you can install it; if you would prefer not to see this message displayed every time you start Unity, you can uncheck the corresponding box labeled Check for Updates accordingly.



Figure 2-4: Automatically checking for Unity updates

Unity provides links to official forums and documentation from the main (i.e., top) menu: Help | Unity Forums or Help | Unity Manual.

Understanding and becoming familiar with the interface

As for any major version of Unity, once you open this software for the first time, a project is already loaded and ready to play. This project usually illustrates and brings forward the most interesting features available in the latest version. In Unity 5.x, this project is called Viking Village and includes a very interesting and fun level to play and to discover. All projects delivered in versions 1, 2, 3 and 4 of Unity are also available through the Assets Store, which is available from Unity by selecting Window | Assets Store from the main menu. At this point in time, although some demo levels have been released for Unity 5.0, an official playable level has not been released yet. This is the reason why we will be using the demo project called Viking Village in the next sections.

After launching Unity, we can notice that it includes several windows organized in a (default) layout. Each of these windows includes a label (usually in the top-left corner) of the window, and all can be moved around, if necessary, by either changing the layout (Window | Layouts | ...) or by dragging and dropping the corresponding tab for a window (this will move the view to where you would like it to appear within the window). In the default layout, the following views appear onscreen (as described in the next screenshot, clockwise from the top left corner):

1. The Hierarchy window (the corresponding shortcut is CTRL+4): this window or view lists all the objects currently present in your scene; these could include, for example, basic shapes, 3D characters, or terrains. This view also makes it possible to identify a hierarchy between objects; for example, we can see in this view if some objects have children or parents (we will explore this concept later).
2. The Scene view (CTRL+1): this window displays the content of a scene (or the item listed in the Hierarchy view) so that you can visualize them and modify them accordingly using the mouse (e.g., move, scale, etc.).
3. The Game view (CTRL+2): this window makes it possible to visualize the scene as it will appear in the game (i.e., through the lenses of the active camera).
4. The Inspector view (CTRL+3): this window displays information (i.e., properties) on the object currently selected.
5. The Console window (SHIFT+CTRL+C): this window displays messages either printed from the code by the user (using keywords) or by Unity. These include warnings or error messages related to your project or code.
6. The Project window (CTRL+5): this window includes all the assets available and used for your project. These include 3D models, sounds, or textures.

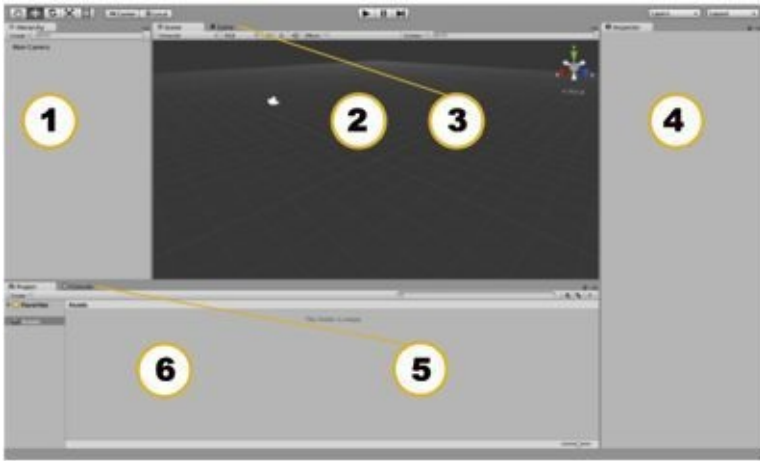


Figure 2-5: Main windows and views in Unity

The scene view

We will use this view to create and visualize the scene for our game. When you create a project, you can include several scenes within. A scene is comparable to a level, and scenes that are included in the same project can share similar resources, so that assets are imported once and shared across (or used in) all scenes. The Scene and Game views are displayed in the same window, and both are represented by a corresponding tab. By default, the Scene view is active; however, it is possible to switch to the Game view by clicking on the tab labeled Game. For example, if we click successively on the Game and Scene tabs, we can see the view from both the perspectives of your eyes (i.e., scene view) and the active camera present in the scene (Game view) as illustrated in the next figures.

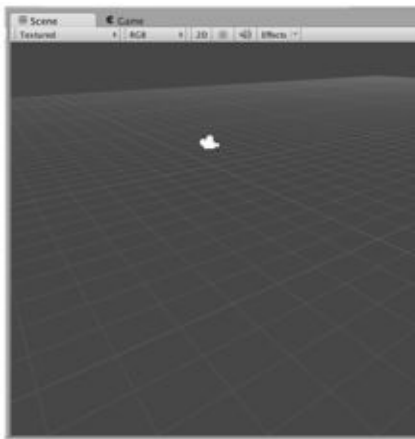


Figure 2-6: The Scene view



Figure 2-7: The Game view

Note that you can also rearrange the layout to be able to, for example, see both the Scene and Game views simultaneously. We could drag and drop the Game tab beside the Console tab to obtain the layout described on the next figure.

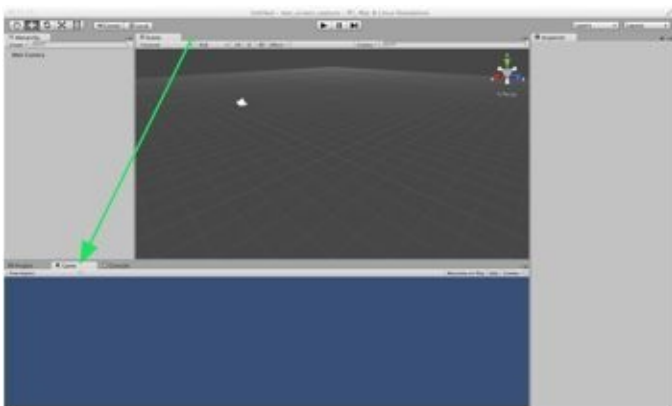


Figure 2-8: Changing the layout to display both Game and Scene views

Discovering and navigating through the scene

So that you can navigate easily in the current scene, several shortcuts and navigation modes are available. These make it possible to navigate through your scene just as you would in a First-Person Shooter or to literally “Fly” through your scene. You can also zoom-in and zoom-out to focus on particular areas or objects, look around (i.e., using mouse look) or pan the view to focus on a specific part of the scene. The main modes of navigation are provided in the next table; however, we will look into these in more detail in the next section as we will be experimenting with them to explore (and modify) an existing scene.

Table 1: Navigation shortcuts

Navigation	Key or Mouse Combination
Activate Fly Mode	Keep MRB (Mouse Right Button) pressed.
Accelerate	Press Shift (in walk mode).
Move in four directions (left, right, forward and back)	Press W, A, S, or D.
Float Up and Down	Press Q or E (in fly mode).
Look around	Press ALT and drag the mouse left, right, forward or back.
Zoom in/out	Move the mouse wheel.
Pan the view	Press Q (to activate the hand tool) then drag and drop the mouse.

For example, in the default navigation mode, you can “walk” through the scene using the arrow keys (up, down, left and right). In the “flight” mode, which can be activated by pressing and holding the Mouse Right Button (MRB), we can navigate using the W, A, S

and D keys and also look around us by dragging the mouse in the direction we would like to look (i.e., left, right, up and down) and also float up and down using the keys Q and E. As you can see, both modes are very useful to navigate through your scene and to visualize all its elements. In addition, you can also choose to display the scene along a particular axis (x, y, or z) using the gizmo that is displayed in the top-right corner of the Scene view as described on the next figure.



Figure 2-9: Gizmo

The gizmo available in the Scene view includes three axes that are color-coded: x (in red), y (in green) and z (in blue). By clicking on any of these axes (or corresponding letters), the scene will be seen accordingly (i.e., through the x-, y-, or z-axis).

If you are not familiar with 3D axes: x, and z usually refer to the width and depth, while y refers to the height. By default, in Unity, the z-axis is pointing towards the screen if the x-axis is pointing to the right and the y-axis is pointing upwards. This is often referred as a left-handed coordinate system.

Also note that by clicking on the middle of the gizmo (white box), we can switch between isometric and perspective views.

In addition to the navigation tools, Unity also offers ways to focus on a particular object by rotating around a specific point (i.e., by pressing the ALT key and dragging the mouse to the left, right, up or down), or double-clicking on an object (i.e., in the Scene or Hierarchy view), so that the camera in the Scene view is focused on this object (this can also be achieved by selecting the object in either the Scene or Hierarchy view and pressing SHIFT+ F), or by zooming-in and out (i.e., scrolling the mouse wheel forward or back).

While the shortcuts and keys described in this section should get you started with Unity and make it possible for you to navigate through your scene easily, there are, obviously, many more shortcuts that you could use, but that will not be presented in this book. Instead, you may look for and find these in the official documentation that is available both offline (using the top menu: Help | Unity Manual then select the sections Unity Overview | Unity Basics | Learning the Interface | Scene View) and online (<http://docs.unity3d.com/Documentation/Manual/SceneViewNavigation.html>). When using the documentation, you can also search for particular words as illustrated on the next figure.

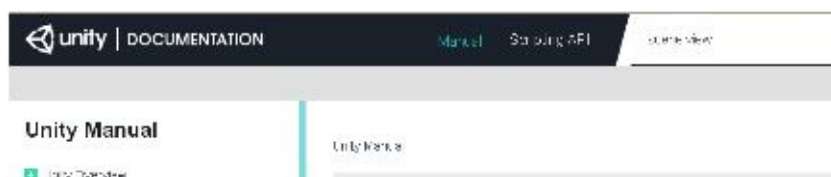


Figure 2-10: Using Unity's manual

The hierarchy view

As indicated by its name, this view lists and displays the name of all objects included in the scene (in alphabetical order by default) along with the type of relationship or hierarchy between them. You may notice that before you add any object to the scene, a camera is already present in the scene so that it can be viewed in the Game view through its lenses.

This view offers several advantages when we need to manage all the scene objects quickly and perform organizational changes (e.g., find objects based on their name, duplicate objects, amend objects' names, amend the properties of several objects simultaneously, or change the hierarchy between objects).

For example, on the following figure, we can see that the scene includes four objects: a camera and three cubes.

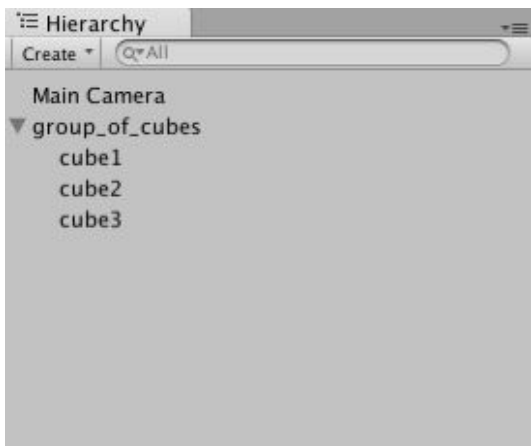


Figure 2-11: Creating a hierarchy between objects

We can also notice that all cubes are grouped under a “folder” (in Unity, this can be created as an empty object in the scene), which means that:

1. All three cubes are children of the object called `group_of_cubes`.
2. The object `group_of_cubes` is the parent of the three cubes.
3. If a transformation (i.e., scale or rotate) is applied to the parent (e.g., group of cubes) it will also be applied to the children (i.e., `cube1`, `cube2`, and `cube3`).

To change the hierarchy of the scene and make some objects children of a particular object, we only need to drag these objects atop the parent object.

The project view

This view includes and displays all assets employed in your project (and across scenes), these include: audio files, textures, scripts (e.g., scripts written in JavaScript, C# or Boo), materials, 3D models, scenes, or packages (i.e., zipped resources for Unity). All these assets, once present in the Project view, can be shared across scenes.

In other words, if we create a project and then a scene, and import assets for our game, these assets will be available from any other scene within the same project.

As for the Hierarchy view, built-in folders and search capabilities are included to ease the management of all your assets.

By default, the Project view includes two windows divided vertically (left and right columns). As illustrated on the next figure, the left window includes a folder called assets and a series of “smart” folders (i.e., the content of these folders varies dynamically) called Favorites. The right window displays the content of the folder selected on the left-hand side.

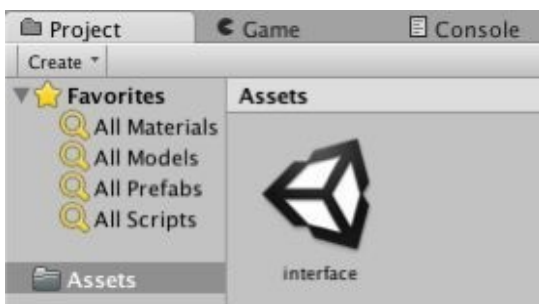


Figure 2-12: The project view

In the previous figure, the content of the folder Assets is displayed and consists of the current scene labeled Interface. By clicking on any of the smart folders (e.g., All Materials, All Models, All Prefabs, or All Scripts) Unity will filter the assets to display only the relevant ones accordingly (e.g., materials, models, prefabs, or scripts). This can speed-up the process of accessing specific assets and can be done (as for many of the functionalities present in Unity) in different ways. For example, you may notice a search window to the left of the Project view as illustrated in the next figure.



Figure 2-13: Searching for assets in the project


The search window in the Project folder can be used to search assets by their name or by their type, as illustrated in the next figure, by clicking on this icon .



Figure
As we
we are
specifi
examp
Materi

