

Get Started with Unity

Fall 2026

9/14/2026

Kyoung Shin Park
Computer Engineering
Dankook University

Game Engine

Origins of Computer Game Engines

- ❑ Game engines arose in the mid-1990s.
- ❑ *Doom* by id provided a separation between:
 - core game components (such as the rendering system, collision detection system, audio system)
 - art assets (models, textures, animations)
 - rules of play
- ❑ *Quakes, Unreal, and Unreal Tournament* (all FPS games) were designed with the separation in mind
 - Sold licenses to their engine and tools
 - So of you may have done modding using these tools.
- ❑ It became generic enough that it was possible to implement a wide variety of very different games based on a common core set of components, *the game engine* (Unity3D and Unreal Engine 4).

Some Current Game Engines

- ❑ Quake family
 - Used to create many games
 - Has lineage that extends to modern games like Medal of Honor
 - Quake and Quake II engines source code are freely available
- ❑ Unreal Engine
 - Now at UE5.6 (2025.6.3)
 - Very rich tool set – Kismet
 - Large developers network
 - Good licensing model – good for small developers

More Game Engines

□ Unity

- Very feature rich
- Uses Javascript or C# for scripting
- Large community support
- Great for cross-platform development

□ Source Engine

- Games like Half-life 2 and its sequels, Team Fortress 2, and Portal
- Very powerful with good graphics capabilities and a good toolset

□ DICE's Frostbite

- Used to create games like Battlefield 4
- FrostEd – asset creation tool

Even More Game Engines

❑ CryEngine

- Originally developed as a demo for Nvidia
- Used to develop numerous games – starting with Far Cry

❑ Sony PhyreEngine

- Uses to create games for the Sony platforms
- Numerous titles have been written with this engine

❑ Microsoft XNA and MonoGame

- Based on C# - easy to use
- Used for Xbox and PC games
- Not longer supported – replaced by MonoGame

2D Game Engines

- ❑ Designed for non-programmers to build apps for Android and iPhone
- ❑ Examples include
 - Multimedia Fusion 2
 - Game Salad Creator
 - Scratch

Best Game Engines

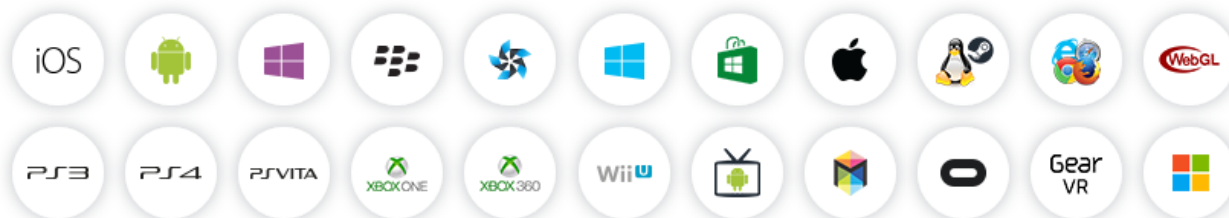
- ❑ GameDev Academy Best Game Engines
 - <https://gamedevacademy.org/best-game-engines/>
- ❑ Top 10 Game Engines – Ulab - SumDU
 - <https://ulab.sumdu.edu.ua/top-10-game-engines>
- ❑ Examples include
 - Unity
 - Unreal
 - Godot
 - Phaser
 - GameMaker
 - CryEngine
 - AppGameKit
 - RPG Maker
 - Amazon Lumberyard
 -

Get Started with Unity

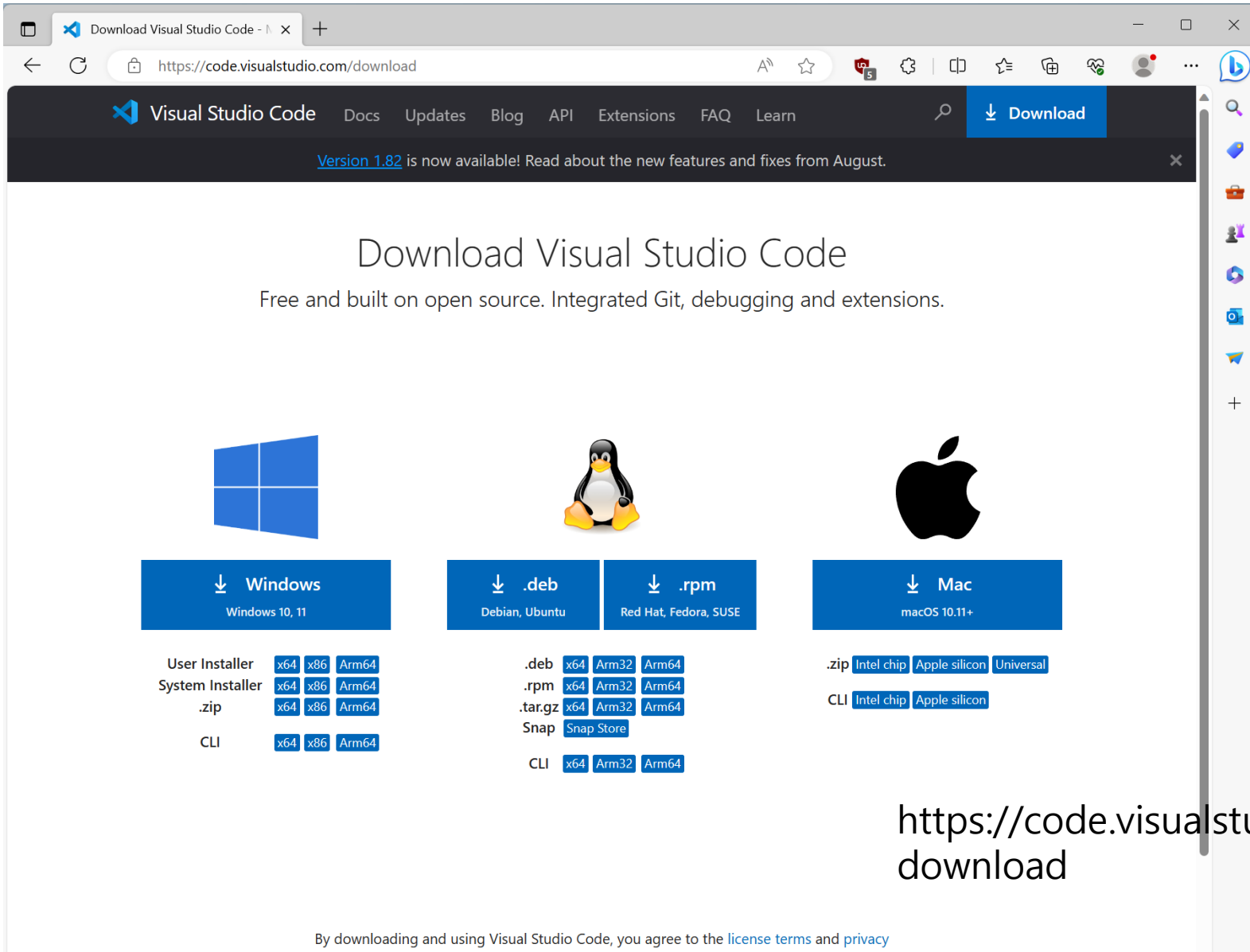
Unity

- ❑ Unity3D is a widely-used **cross-platform** game development system.
- ❑ It consists of a **game engine** and an **integrated development environment (IDE)**.
- ❑ It can be used to develop games for many different platforms, PCs, consoles, mobile devices and deployment on the Web.
- ❑ Tutorials at <https://learn.unity.com/>

 **BUILD ONCE DEPLOY ANYWHERE**



Visual Studio Code



The screenshot shows the Visual Studio Code download page in a web browser. The browser's address bar displays the URL <https://code.visualstudio.com/download>. The page header includes the Visual Studio Code logo, navigation links (Docs, Updates, Blog, API, Extensions, FAQ, Learn), and a prominent blue 'Download' button. A notification banner states: 'Version 1.82 is now available! Read about the new features and fixes from August.' The main heading is 'Download Visual Studio Code', followed by the tagline 'Free and built on open source. Integrated Git, debugging and extensions.' Below this, three operating system icons (Windows, Linux/Tux, and Mac) are shown. Under the Windows icon, a blue button labeled 'Windows' (Windows 10, 11) leads to a list of download options: User Installer, System Installer, .zip, and CLI, each with buttons for x64, x86, and Arm64 architectures. Under the Linux icon, two blue buttons labeled '.deb' (Debian, Ubuntu) and '.rpm' (Red Hat, Fedora, SUSE) are shown. The '.deb' button leads to options for .deb, .rpm, .tar.gz, Snap, and CLI, with architecture buttons (x64, Arm32, Arm64) for the first four. The '.rpm' button leads to a 'Snap Store' button. Under the Mac icon, a blue button labeled 'Mac' (macOS 10.11+) leads to options for .zip (Intel chip, Apple silicon, Universal) and CLI (Intel chip, Apple silicon).

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Version 1.82 is now available! Read about the new features and fixes from August.

Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.



↓ Windows
Windows 10, 11

User Installer [x64](#) [x86](#) [Arm64](#)
System Installer [x64](#) [x86](#) [Arm64](#)
.zip [x64](#) [x86](#) [Arm64](#)
CLI [x64](#) [x86](#) [Arm64](#)



↓ .deb
Debian, Ubuntu

↓ .rpm
Red Hat, Fedora, SUSE

.deb [x64](#) [Arm32](#) [Arm64](#)
.rpm [x64](#) [Arm32](#) [Arm64](#)
.tar.gz [x64](#) [Arm32](#) [Arm64](#)
Snap [Snap Store](#)
CLI [x64](#) [Arm32](#) [Arm64](#)



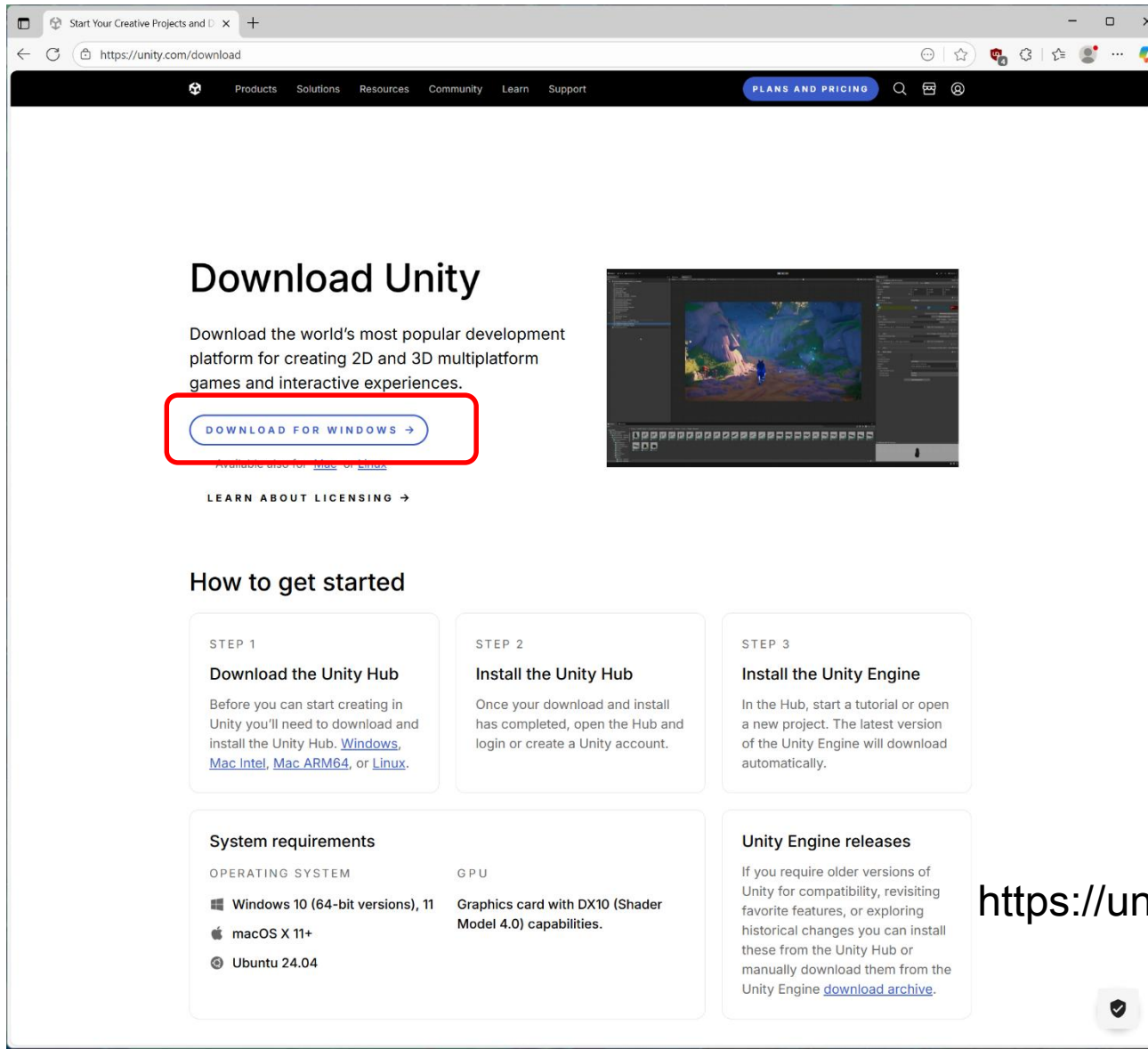
↓ Mac
macOS 10.11+

.zip [Intel chip](#) [Apple silicon](#) [Universal](#)
CLI [Intel chip](#) [Apple silicon](#)

<https://code.visualstudio.com/download>

By downloading and using Visual Studio Code, you agree to the [license terms](#) and [privacy](#)

Download Unity Hub



The screenshot shows the Unity Hub download page in a web browser. The browser's address bar displays 'https://unity.com/download'. The page features a dark navigation bar with links for 'Products', 'Solutions', 'Resources', 'Community', 'Learn', and 'Support', along with a 'PLANS AND PRICING' button. The main content area has a large heading 'Download Unity' and a subtext 'Download the world's most popular development platform for creating 2D and 3D multiplatform games and interactive experiences.' A red rectangle highlights the 'DOWNLOAD FOR WINDOWS →' button. To the right is a screenshot of the Unity Hub interface showing a 3D scene. Below the main content is a 'How to get started' section with three steps: 'STEP 1 Download the Unity Hub', 'STEP 2 Install the Unity Hub', and 'STEP 3 Install the Unity Engine'. At the bottom, there are sections for 'System requirements' (listing Windows 10, macOS X 11+, and Ubuntu 24.04) and 'Unity Engine releases' (providing a link to the download archive).

Start Your Creative Projects and D x +

https://unity.com/download

Products Solutions Resources Community Learn Support

PLANS AND PRICING

Download Unity

Download the world's most popular development platform for creating 2D and 3D multiplatform games and interactive experiences.

[DOWNLOAD FOR WINDOWS →](#)

Available also for: [Linux](#) or [macOS](#)

[LEARN ABOUT LICENSING →](#)

How to get started

STEP 1

Download the Unity Hub

Before you can start creating in Unity you'll need to download and install the Unity Hub. [Windows](#), [Mac Intel](#), [Mac ARM64](#), or [Linux](#).

STEP 2

Install the Unity Hub

Once your download and install has completed, open the Hub and login or create a Unity account.

STEP 3

Install the Unity Engine

In the Hub, start a tutorial or open a new project. The latest version of the Unity Engine will download automatically.

System requirements

OPERATING SYSTEM	GPU
Windows 10 (64-bit versions), 11	Graphics card with DX10 (Shader Model 4.0) capabilities.
macOS X 11+	
Ubuntu 24.04	

Unity Engine releases

If you require older versions of Unity for compatibility, revisiting favorite features, or exploring historical changes you can install these from the Unity Hub or manually download them from the Unity Engine [download archive](#).

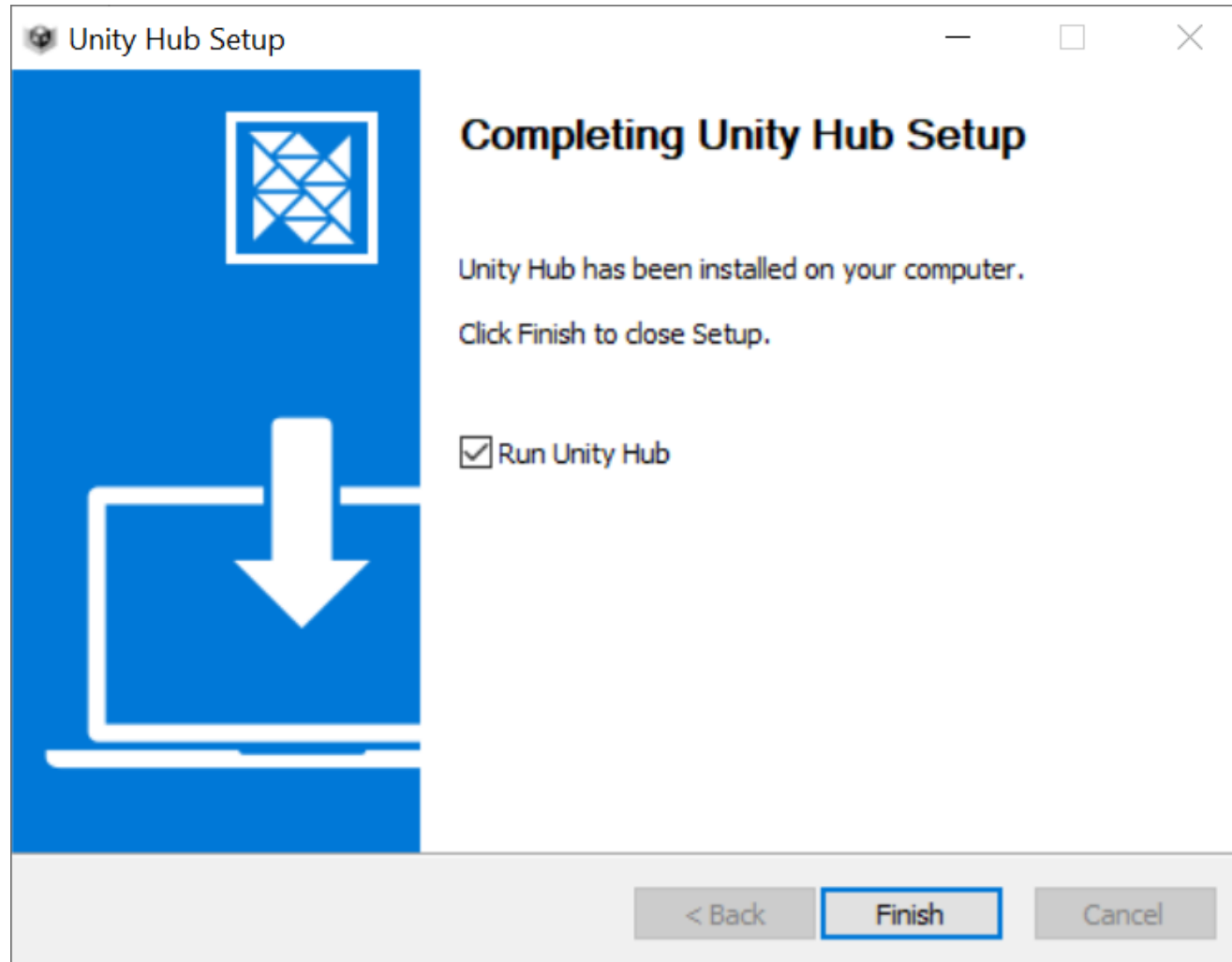
<https://unity.com/download>

Install Unity Hub

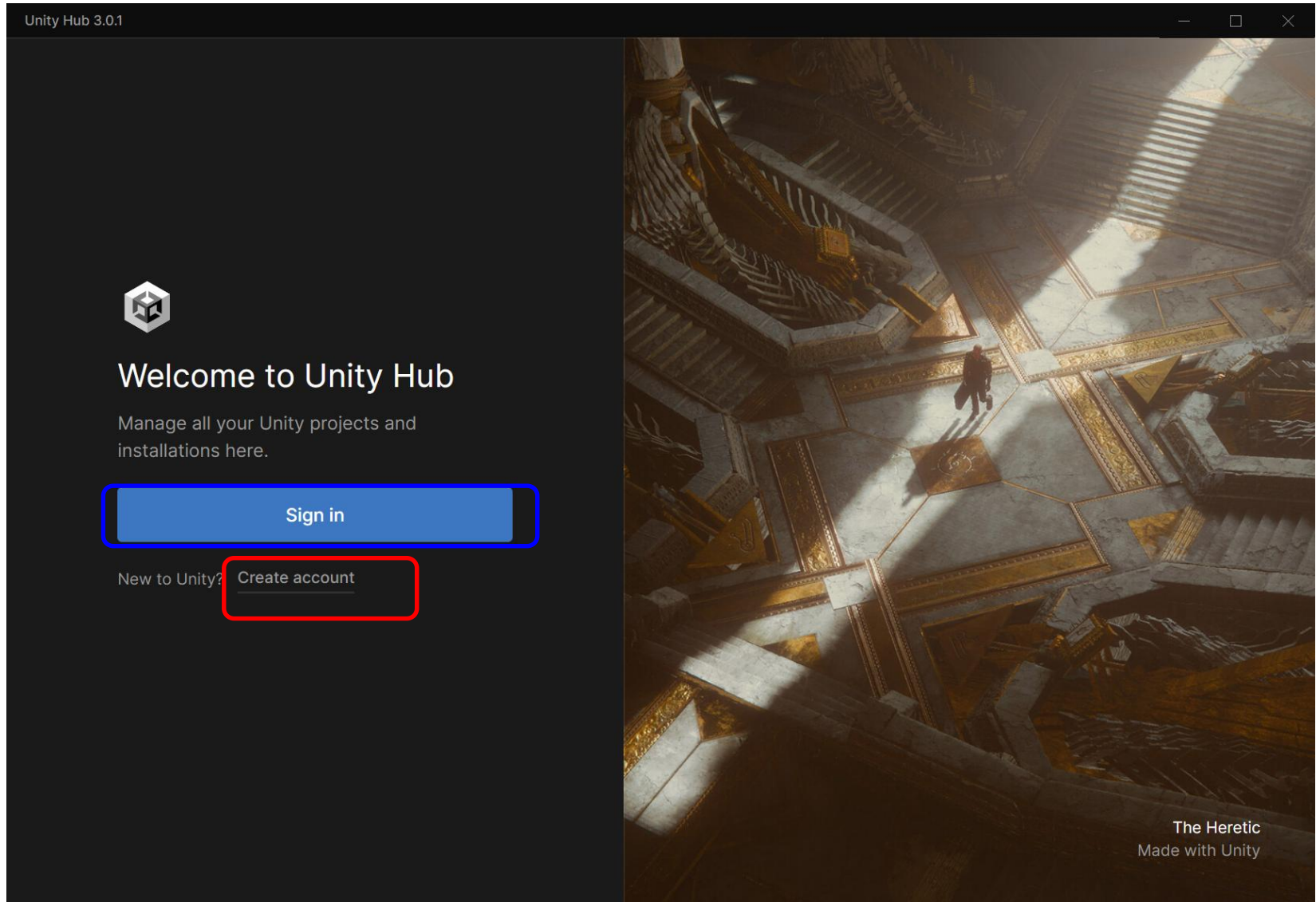


UnityHubSetup.exe

Run Unity Hub



Sign in (or Create account)



Install Unity

The screenshot shows the Unity Hub 3.10.0 application window. On the left is a dark sidebar with navigation options: 'Projects', 'Installs' (highlighted), 'Learn', 'Community', and 'Downloads'. The main area has a top notification bar stating 'Hub V3.14.0 is now available and will install after restarting.' with 'Restart now' and 'Dismiss' buttons. Below this is the 'Installs' section, which includes tabs for 'All', 'Official releases', and 'Pre-releases', and a search bar. The 'UNITY 6.1' section displays a card for 'Unity 6.2 (6000.2.1f1) LTS' with a settings gear icon, the file path 'C:\Program Files\Unity\Hub\Editor\6000.2.1f1\Editor\Unity.exe', and platform tags 'UWP' and 'Windows'. Below this is the 'OTHER VERSIONS' section, showing two more cards: 'Unity (2022.3.51f1) LTS' and 'Unity (2022.3.47f1) LTS', both with their respective file paths and platform tags.

Unity Hub 3.10.0

Hub V3.14.0 is now available and will install after restarting.
[See Release Notes](#) Restart now Dismiss

Installs

Locate Install Editor

All Official releases Pre-releases

UNITY 6.1

 **Unity 6.2 (6000.2.1f1)** LTS 
C:\Program Files\Unity\Hub\Editor\6000.2.1f1\Editor\Unity.exe
UWP Windows

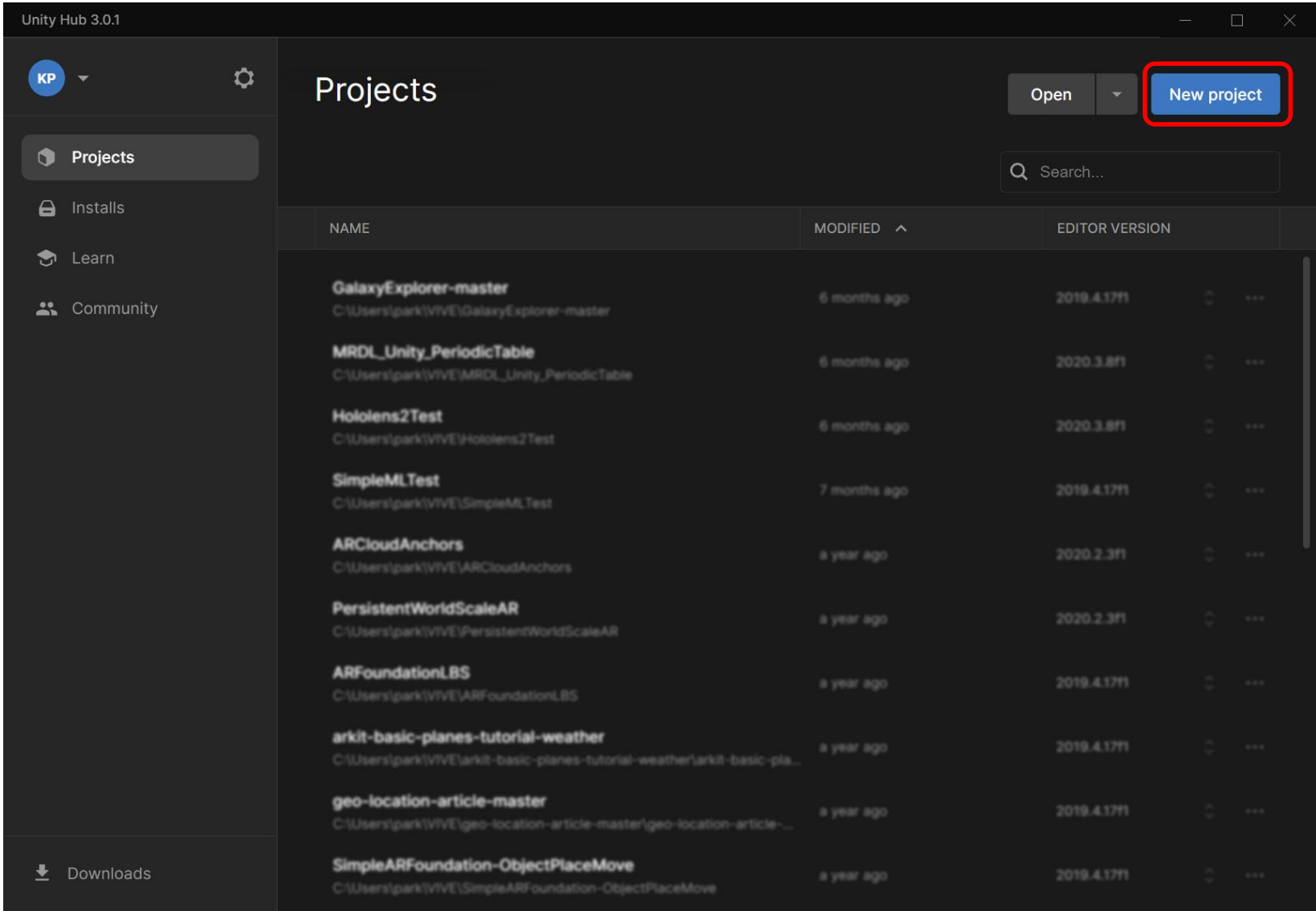
OTHER VERSIONS

 **Unity (2022.3.51f1)** LTS 
C:\Program Files\Unity\Hub\Editor\2022.3.51f1\Editor\Unity.exe
Android UWP Windows

 **Unity (2022.3.47f1)** LTS 
C:\Program Files\Unity\Hub\Editor\2022.3.47f1\Editor\Unity.exe

Install 6000.2.1f1 (LTS)

Create a New Project



The screenshot shows the Unity Hub 3.0.1 interface. On the left is a sidebar with navigation options: Projects (selected), Installs, Learn, and Community. The main area is titled 'Projects' and contains a table of existing projects. In the top right of the main area, there are 'Open' and 'New project' buttons. The 'New project' button is highlighted with a red rectangle. Below the buttons is a search bar. The table lists projects with columns for NAME, MODIFIED, and EDITOR VERSION. Each row also includes icons for refreshing and deleting the project.

NAME	MODIFIED	EDITOR VERSION
GalaxyExplorer-master C:\Users\park\VIVE\GalaxyExplorer-master	6 months ago	2019.4.17f1
MRDL_Unity_PeriodicTable C:\Users\park\VIVE\MRDL_Unity_PeriodicTable	6 months ago	2020.3.8f1
Hololens2Test C:\Users\park\VIVE\Hololens2Test	6 months ago	2020.3.8f1
SimpleMLTest C:\Users\park\VIVE\SimpleMLTest	7 months ago	2019.4.17f1
ARCloudAnchors C:\Users\park\VIVE\ARCloudAnchors	a year ago	2020.2.3f1
PersistentWorldScaleAR C:\Users\park\VIVE\PersistentWorldScaleAR	a year ago	2020.2.3f1
ARFoundationLBS C:\Users\park\VIVE\ARFoundationLBS	a year ago	2019.4.17f1
arkit-basic-planes-tutorial-weather C:\Users\park\VIVE\arkit-basic-planes-tutorial-weather\arkit-basic-pla...	a year ago	2019.4.17f1
geo-location-article-master C:\Users\park\VIVE\geo-location-article-master\geo-location-article-...	a year ago	2019.4.17f1
SimpleARFoundation-ObjectPlaceMove C:\Users\park\VIVE\SimpleARFoundation-ObjectPlaceMove	a year ago	2019.4.17f1

Unity Interface

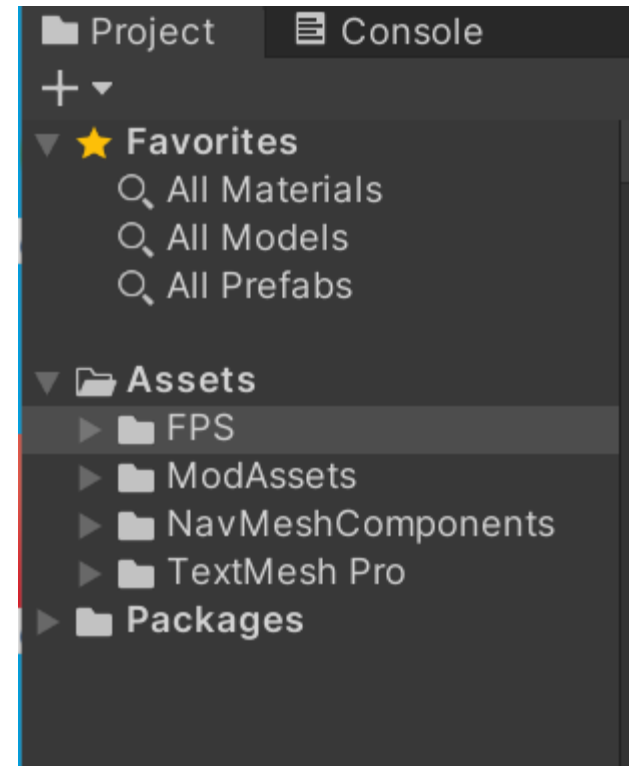
Unity Interface

- Project
- Scene View
- Game View
- Hierarchy
- Inspector
- Console



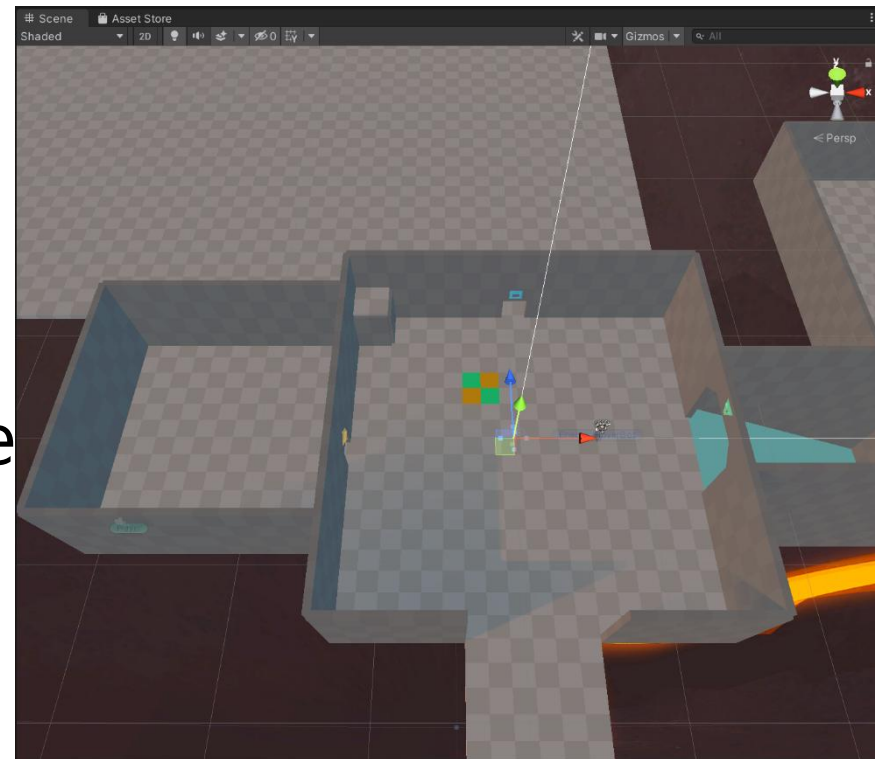
Project Window

- ❑ The project window contains all of the assets that are available for you to use.
- ❑ Typically, these are organized into folders, for example, according to the **asset type** (models, materials, audio, prefabs, scripts, etc.).



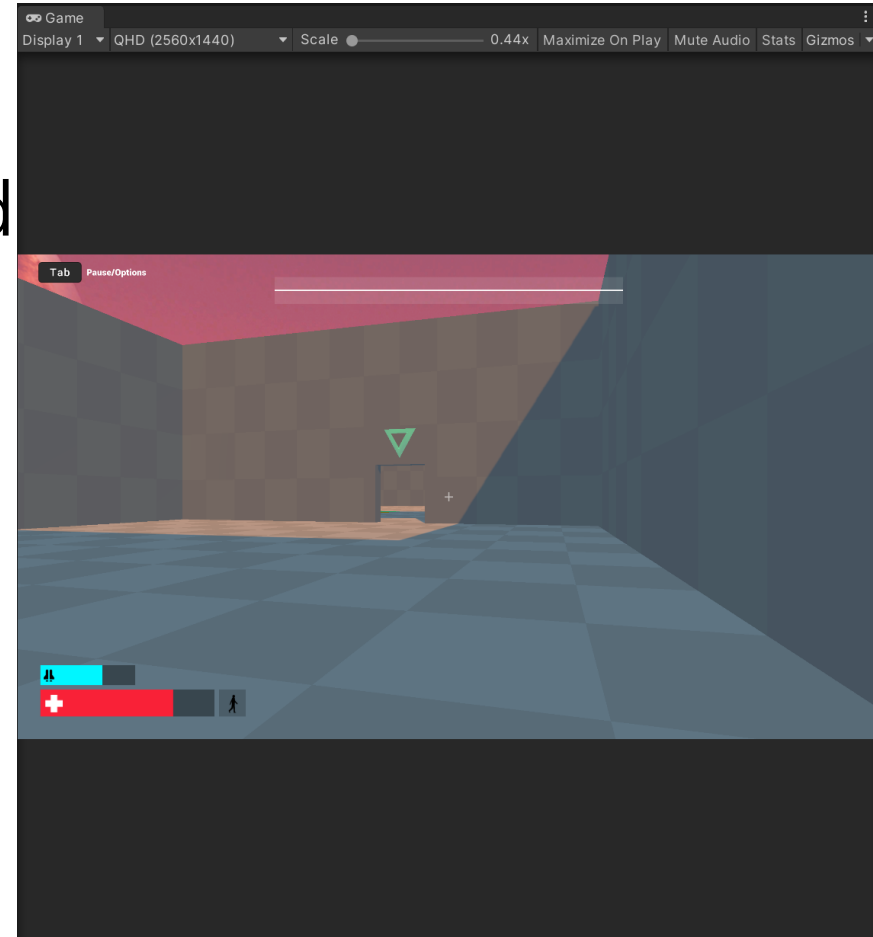
Scene View

- ❑ This scene window shows all the elements of the current scene.
- ❑ Most editing of the scene is done through the scene view, because it provides access to low-level and hidden aspects of the objects.



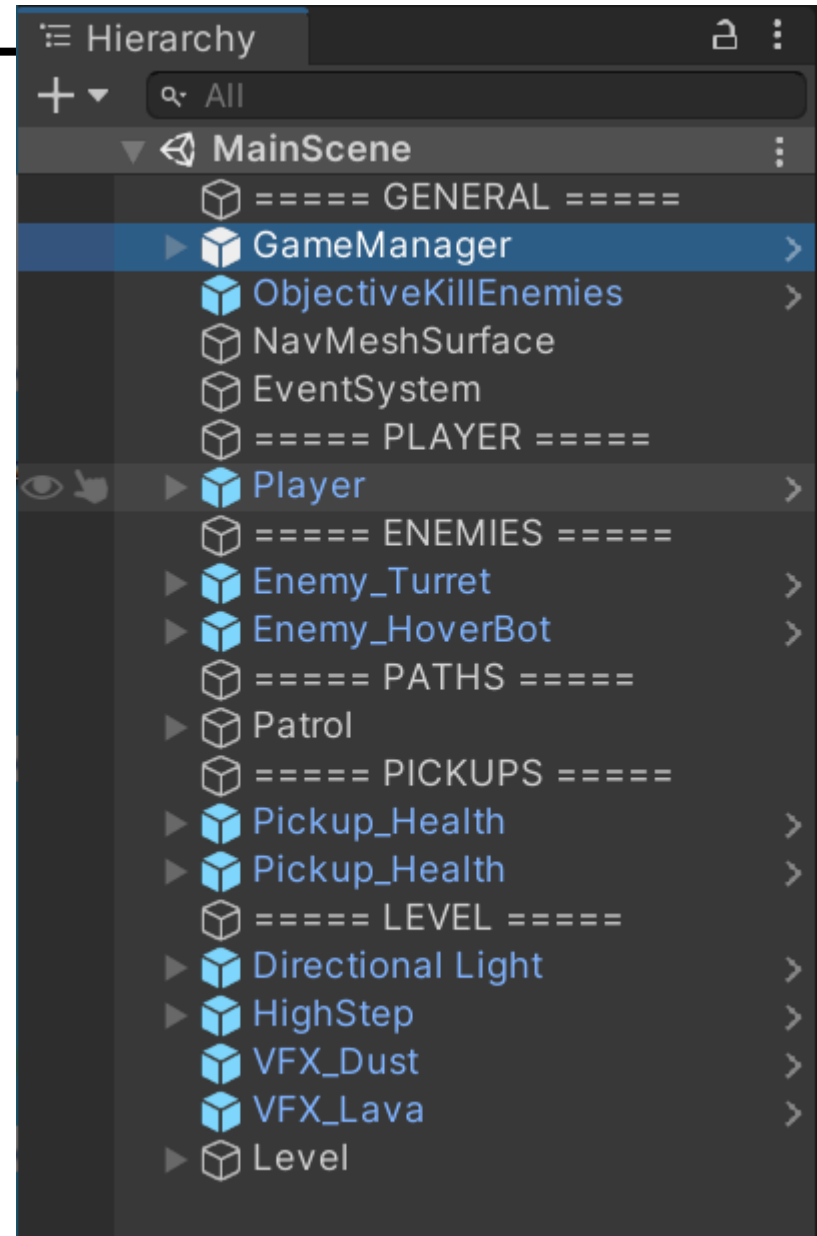
Game View

- This game window shows the elements of the scene as they would appear to the player.



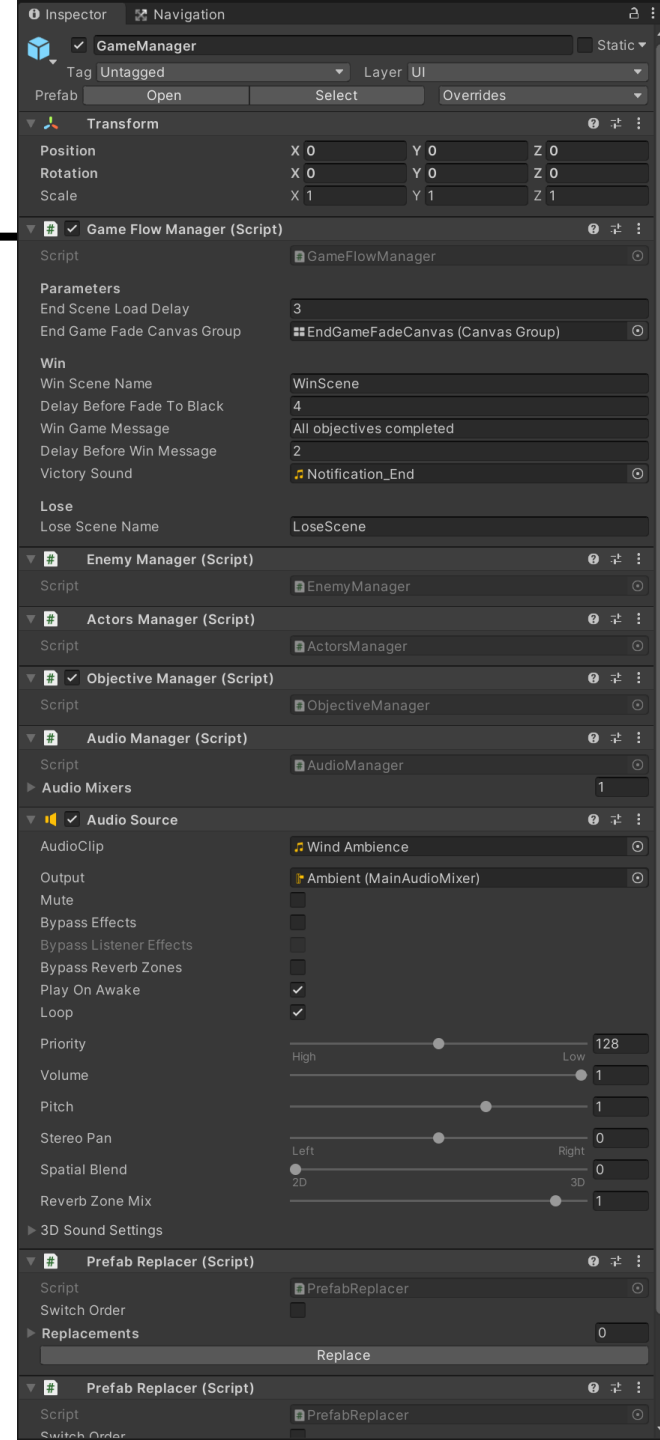
Hierarchy

- ❑ This window shows all the game objects that constitute the current scene.
- ❑ Game objects are stored hierarchically in a tree structure.



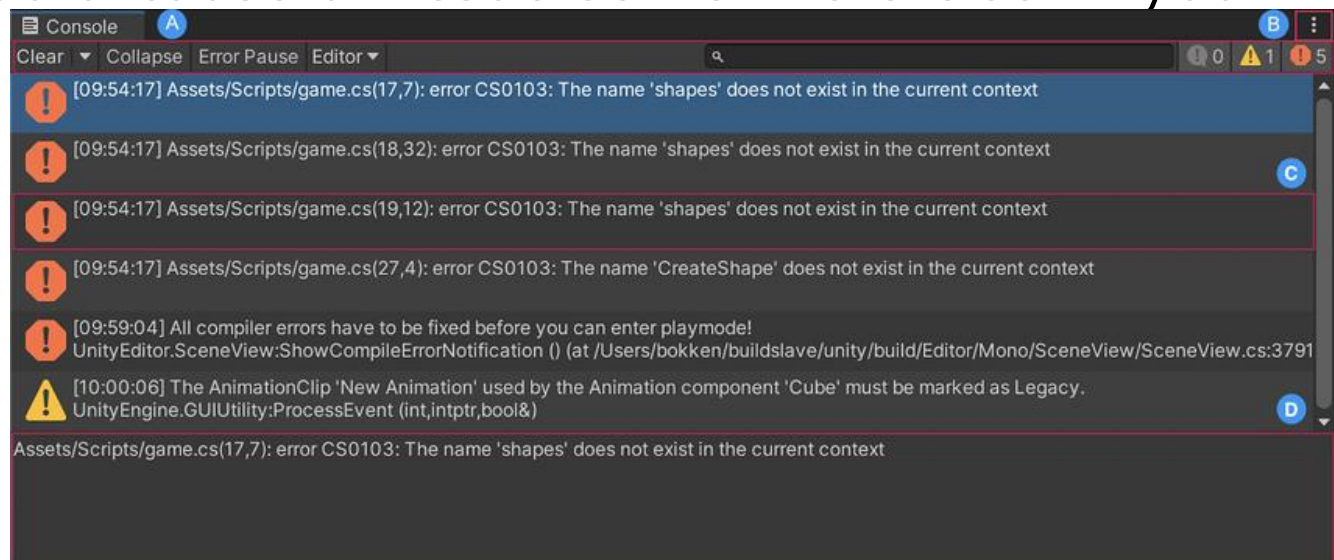
Inspector

- At any time there is an active **game object** (which the designer selects by clicking on the object or on its entry in the hierarchy).
- This window provides all the **component** information associated with this object.



Console

- ❑ The console window displays errors, warnings, and other messages the Editor generates.
- ❑ These errors and warnings help you find issues in your project, such as script compilation errors.
- ❑ They also alert you to actions the Editor has taken automatically, such as replacing missing meta files, which could cause an issue somewhere else in your project.

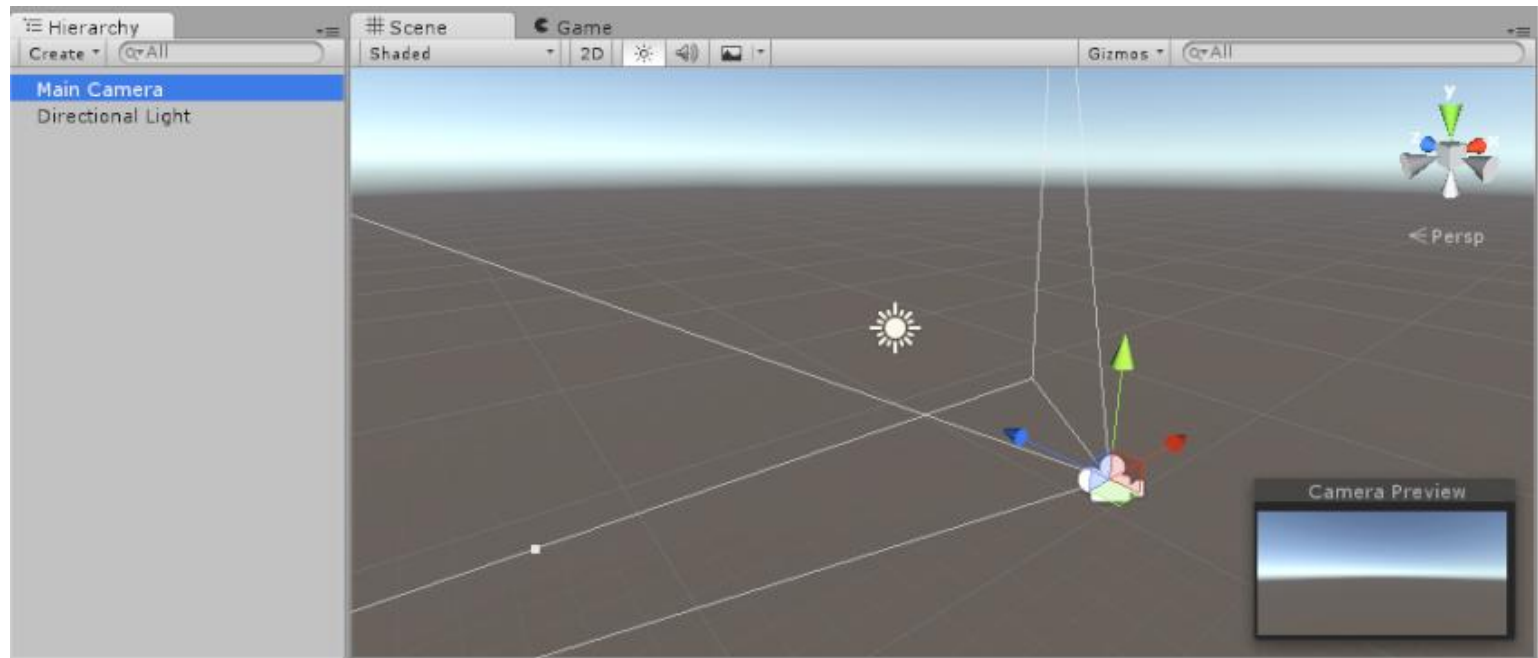


Unity Basics

Scene

□ Scenes

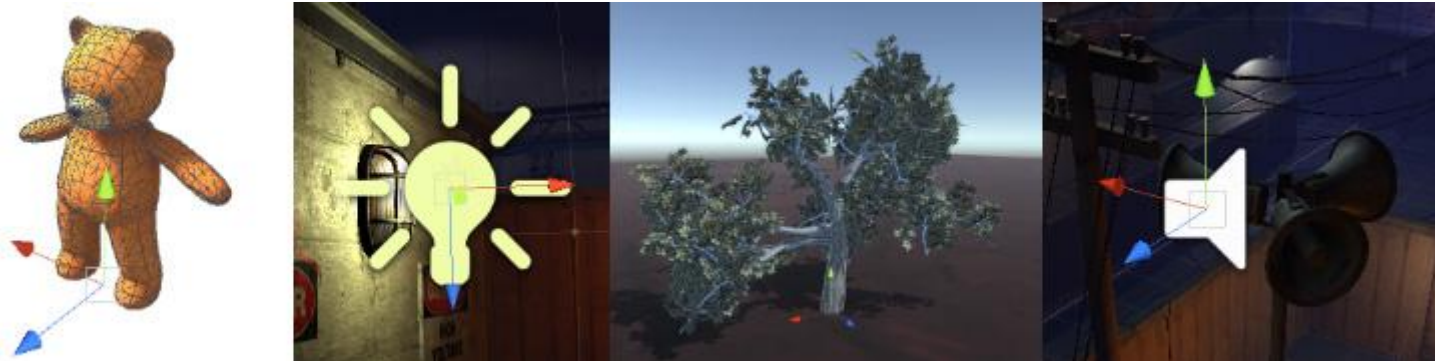
- A scene contains a collection of game objects that constitute the world that the player sees at any time.
- Below example shows a sample scene that contains only a **Camera** and a **Light**.



GameObjects

□ GameObjects

- The game objects are all the things that constitute your scene.
- **GameObjects** are the fundamental objects in Unity that represent characters, props and scenery. They do not accomplish much in themselves but they act as containers for **Components**, which implement the functionality.

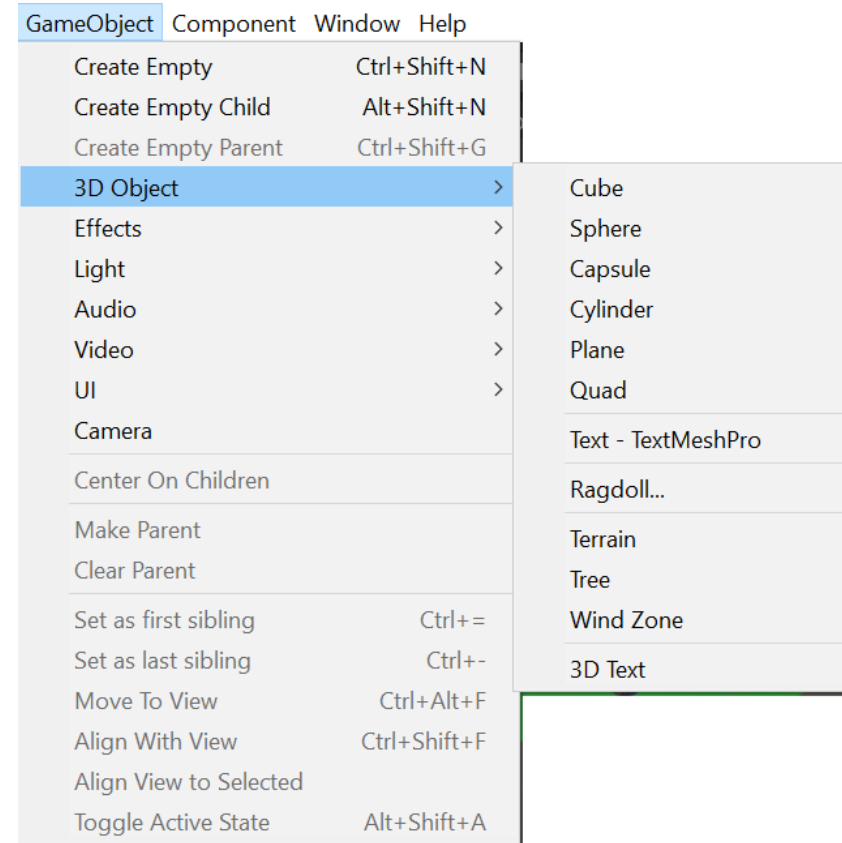


Four different types of GameObject: an animated character, a light, a tree, and an audio source

GameObjects

□ GameObject

- Empty
- 3D Object – Cube, Sphere,
- Light – Directional Light,
- Audio
- Video
- Effect - Particle System
- UI
- Camera



Components

□ Components

- Each **GameObject** is defined by a collection of associated elements, called **Components**.
- The set of components that are associated with a game object depend on the nature of object. For example, a light source object is associated with color and intensity of the light source. A camera object is associated with various properties of how the projection is computed (wide-angle or telephoto).
- The various components that are associated with a game object can be viewed and edited in the **Inspector window**.

Components

□ Components

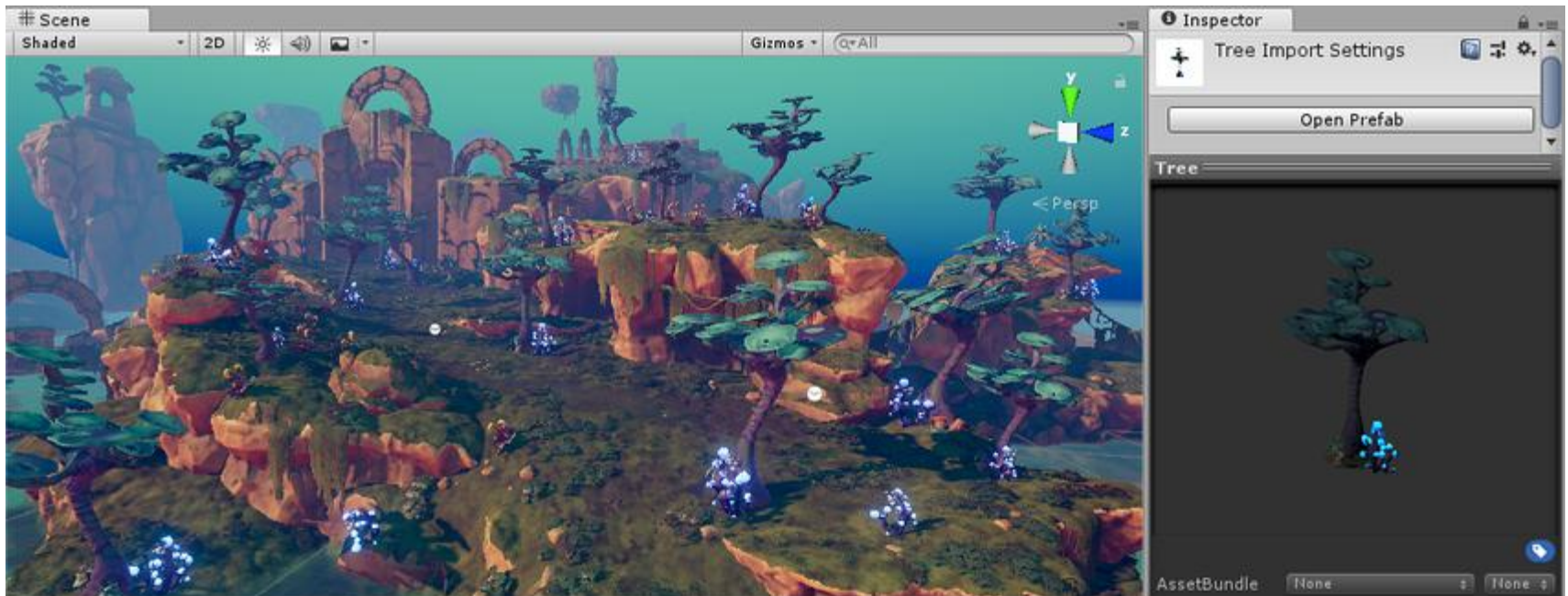
- Mesh
- Effects
- Physics
- Physics2D
- Navigation
- Audio
- Video
- Rendering
- Tilemap
- Layout
- Playables
- Miscellaneous
- Scripts
- UI
- Event

Component	Window	Help
Add...		Ctrl+Shift+A
Mesh		>
Effects		>
Physics		>
Physics 2D		>
Navigation		>
Audio		>
Video		>
Rendering		>
Tilemap		>
Layout		>
Playables		>
Miscellaneous		>
Scripts		>
UI		>
Event		>

Prefabs

❑ Prefabs

- Unity's Prefab system allows you to create, configure, and store a GameObject complete with all its components, property values, and child GameObjects as a **reusable Asset**.
- The **Prefab** Asset acts as a template from which you can create new Prefab instances in the Scene.



Assets

□ Assets

- An asset is any **resource** that will be used as part of an object's component.
- Examples include **meshes** (for defining the shapes of objects), **materials** (for defining shapes), **physics materials** (for defining physical properties like friction), and **scripts** (for defining behaviors).

Scripts

□ Scripts

- A script is a **chunk of code** that defines the **behavior of game objects**. Scripts are associated with game objects.
- There are various types of scripts classes, depending on the type of behavior being controlled.
- Because interactive game programming is **event-driven**, a typical script is composed as a collection of functions, each of which is invoked in response to a particular event. (E.g., A function may be invoked when this object collides with another object.)
- Typically, each of these functions performs some simple action (e.g., moving the game object, creating/destroying game objects, triggering events for other game objects), and then returns control to the system.

IDE (Visual Studio) for Script

Menu structure for Visual Studio:

- Edit
 - Undo (Ctrl+Z)
 - Redo (Ctrl+Y)
 - Select All (Ctrl+A)
 - Deselect All (Shift+D)
 - Select Children (Shift+C)
 - Select Prefab Root (Ctrl+Shift+R)
 - Invert Selection (Ctrl+I)
 - Cut (Ctrl+X)
 - Copy (Ctrl+C)
 - Paste (Ctrl+V)
 - Paste As Child (Ctrl+Shift+V)
 - Duplicate (Ctrl+D)
 - Rename
 - Delete
 - Frame Selected (F)
 - Lock View to Selected (Shift+F)
 - Find (Ctrl+F)
 - Play (Ctrl+P)
 - Pause (Ctrl+Shift+P)
 - Step (Ctrl+Alt+P)
 - Sign in...
 - Sign out
 - Selection >
 - Project Settings...
 - Preferences...
 - Shortcuts...
 - Clear All PlayerPrefs
 - Graphics Tier >
 - Grid and Snap Settings...
- Assets
- GameObject
- Component
- Window

Visual Studio Preferences - External Tools

External Script Editor: Visual Studio Community 2022 [17.1.0]

Visual Studio Editor v2.0.14 enabled

Generate .csproj files for:

- Embedded packages ☒
- Local packages ☒
- Registry packages ☐
- Git packages ☐
- Built-in packages ☐
- Local tarball ☐
- Packages from unknown sources ☐
- Player projects ☐

Regenerate project files

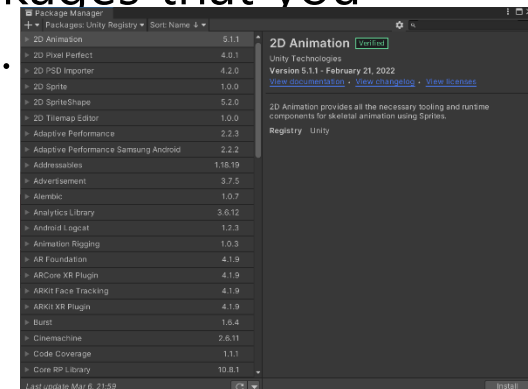
Image application: Open by file extension

Revision Control Diff/Merge: WinMerge

Packages

□ Packages

- A **package** is an aggregation of game objects and their associated meta-data.
- They are related objects (models, scripts, materials, etc.). Here are some examples:
 - a collection of shaders for **rendering water effects**
 - particle systems for **creating explosions**
 - models of race cars for **a racing game**
 - models of trees and bushes to **create a woodland scene**
- Unity provides a number standard packages for free, and when a new project is created, you can select the packages that you would like to have imported into your project.



Reference

- Unity Manual
<https://docs.unity3d.com/Manual/UnityOverview.html>