CHAPTER 18

#### **HELLO WORLD: YOUR FIRST PROGRAM**







- Hello World?
- Creating a Unity Project



- Hello World?
- Creating a Unity Project
  - The Unity Project Folder



- Hello World?
- Creating a Unity Project
  - The Unity Project Folder
- MonoDevelop: Unity's Code Editor



- Hello World?
- Creating a Unity Project
  - The Unity Project Folder
- MonoDevelop: Unity's Code Editor
- Attaching Scripts to GameObjects



- Hello World?
- Creating a Unity Project
  - The Unity Project Folder
- MonoDevelop: Unity's Code Editor
- Attaching Scripts to GameObjects
- Start() and Update()



- Hello World?
- Creating a Unity Project
  - The Unity Project Folder
- MonoDevelop: Unity's Code Editor
- Attaching Scripts to GameObjects
- Start() and Update()
- GameObject Prefabs and Instantiation



- Hello World?
- Creating a Unity Project
  - The Unity Project Folder
- MonoDevelop: Unity's Code Editor
- Attaching Scripts to GameObjects
- Start() and Update()
- GameObject Prefabs and Instantiation
- The HelloWorld Project



 Hello World is often the first program written by anyone learning a new programming language.



- Hello World is often the first program written by anyone learning a new programming language.
- Outputs "Hello World!" to the Console



- Hello World is often the first program written by anyone learning a new programming language.
- Outputs "Hello World!" to the Console

E Cauch		
Clear Collapse Clear on play Error pause	Open Player Log	→= Open Editor Log
Hello World!     UnityEngine.MonoBehaviour:print(Object)		
Hello World!		





The code of HelloWorld.cs is very simple:



The code of HelloWorld.cs is very simple:





From the menu bar, choose File > New Project...



- From the menu bar, choose File > New Project...
- Choose the location for your project folder



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field
    - Click the Save button

- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field
    - Click the Save button
  - Windows



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field
    - Click the Save button
  - Windows
    - Click the *Browse…* button



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field
    - Click the Save button
  - Windows
    - Click the *Browse…* button
    - Navigate to the right location



- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field
    - Click the Save button
  - Windows
    - Click the *Browse…* button
    - Navigate to the right location
    - Click the *New Folder* button and give the new folder the name of your project.

- From the menu bar, choose File > New Project...
- Choose the location for your project folder
  - Mac OS X
    - Click the Set... button
    - Navigate to the right location
    - Type the project name into the Save As field
    - Click the Save button
  - Windows
    - Click the *Browse…* button
    - Navigate to the right location
    - Click the *New Folder* button and give the new folder the name of your project.
    - Click the Select Folder button

$\mathbf{O}$ $\mathbf{O}$	Project Wizard (	4.3.4f1)	
(	Open Project Creat	te new Project	
Project Director	'Y:		
/Users/jrgibs	on/New Unity Project 1	Set.	
Import the follo	wing packages:		
Character (	Controller.unityPackage		
Glass Refra	ction (Pro Only).unityPacl	kage	
📄 Image Effe	ts (Pro Only).unityPackag	je	
Light Cook	ies.unityPackage		
Light Flares	s.unityPackage		
Particles.ur	nityPackage		
Physic Mate	erials.unityPackage		
Projectors.	unityPackage		
Scripts.unit	tyPackage		
Skyboxes.u	InityPackage		
Chandend A			_
Set up defaults	for: 3D ‡	Create Proje	ect

Jnity - Project Wizard (4.3.4f1)	×
Open Project Create New Project	
Project Location:	
C:\Users\jrgibson\Documents\New Unity Project	Browse
Import the following	~
Character Controller.unityPackage	<b>A</b>
Light Cookies.unityPackage	
Light Flares.unityPackage	=
Particles.unityPackage	
Physic Materials.unityPackage	
Projectors.unityPackage	
Scripts.unityPackage	
Skyboxes.unityPackage	
•	•
Setup defaults for: 3D 🔹	Create



#### Set up defaults for 3D

Open Project Create new Project	Open Project Create New Project	
Project Directory:          /Users/jrgibson/New Unity Project 1       Set         Import the following packages:	Project Location: C:\Users\jrgibson\Documents\New Unity Project	Browse
Create Project	Import the following         Character Controller.unityPackage         Light Cookies.unityPackage         Light Flares.unityPackage         Particles.unityPackage         Physic Materials.unityPackage         Projectors.unityPackage         Scripts.unityPackage         Skvboxes.unityPackage         Import the following         Import the following         Setup defaults for:	Create



X

- Set up defaults for 3D
- Click the Create Project or Create button

Open Project Create new Project Project Directory: /Users/jrgibson/New Unity Project 1 Set	Open Project       Create New Project         Project Location:       C:\Users\jrgibson\Documents\New Unity Project	Browse
Import the following packages:         Character Controller.unityPackage         Glass Refraction (Pro Only).unityPackage         Image Effects (Pro Only).unityPackage         Light Cookies.unityPackage         Light Flares.unityPackage         Particles.unityPackage         Physic Materials.unityPackage         Scripts.unityPackage         Skyboxes.unityPackage         Skyboxes.unityPackage         Create Project	Import the following  Character Controller.unityPackage Light Cookies.unityPackage Light Flares.unityPackage Particles.unityPackage Physic Materials.unityPackage Scripts.unityPackage Scripts.unityPackage Skvboxes.unitvPackaae III Setup defaults for: 3D	Create

- Set up defaults for 3D
- Click the Create Project or Create button
- Appendix A contains detailed instructions

O     O     Project Wizard (4.3.4f1)	Unity - Project Wizard (4.3.4f1)	×
Open Project Create new Project Project Directory: /Users/jrgibson/New Unity Project 1 Set	Open Project       Create New Project         Project Location:       C:\Users\jrgibson\Documents\New Unity Project	Browse
<ul> <li>Character Controller.unityPackage</li> <li>Glass Refraction (Pro Only).unityPackage</li> <li>Image Effects (Pro Only).unityPackage</li> <li>Light Cookies.unityPackage</li> <li>Light Flares.unityPackage</li> <li>Particles.unityPackage</li> <li>Physic Materials.unityPackage</li> <li>Projectors.unityPackage</li> <li>Skyboxes.unityPackage</li> <li>Create Project</li> </ul>	Import the following  Character Controller.unityPackage Light Cookies.unityPackage Particles.unityPackage Particles.unityPackage Physic Materials.unityPackage Scripts.unityPackage Skvboxes.unitvPackage Skvboxes.unitvPackage III Setup defaults for: 3D	Create



The Project pane shows the contents of the Assets folder inside your Unity project folder


## **Creating a Unity Project**

- The Project pane shows the contents of the Assets folder inside your Unity project folder
  - Right-click in the Project pane and choose Reveal in Finder (or Show in Explorer) from the pop-up menu



# **Creating a Unity Project**

 The Project pane shows the contents of the Assets folder inside your Unity project folder

 Right-click in the Project pane and choose Reveal in Finder (or Show in Explorer) from the pop-up menu

Create *	(Q)   4   9 me_0	Main Car
	Create	•
	Reveal in Finder Open Delete	
	Import New Asset Import Package Export Package Find References In Scene Select Dependencies	•
	Refresh Reimport	ЖR
	Reimport All	
	Sync MonoDevelop Project	ct
		-



## **Creating a Unity Project**

 The Project pane shows the contents of the Assets folder inside your Unity project folder

 Right-click in the Project pane and choose Reveal in Finder (or Show in Explorer) from the pop-up menu

Scene_0	Main Car	000	🔲 Hello World	
Create	•			▼ Q
Reveal in Finder		FAVORITES	Name	<ul> <li>Date Added</li> </ul>
Open 🔹		A iraibson	🔻 🚞 Assets	Today 7:35 PM
Delete		jigibson	Scene_0.unity	Today 11:09 PM
Import New Asset		Desktop	🕨 🚞 Library	Today 7:35 PM
Import Package	•	Documents	ProjectSettings	Today 7:35 PM
Export Package		Downloads	🕨 🚞 Temp	Today 11:06 PM
Find References In Scene		Drophox		
Select Dependencies				
Refresh	₩R	📇 All My Files		
Reimport		AirDrop		
Reimport All		Applications		▶ 🚞 Unity Projects ▶ 🚞 Hello Wor
Sync MonoDevelop Project		5 items, 29.69 GB available		



• Unity uses MonoDevelop for code editing



- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team



- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team
- To open MonoDevelop, double-click any C# script in your Project pane



- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team
- To open MonoDevelop, double-click any C# script in your Project pane
  - This will launch MonoDevelop



- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team
- To open MonoDevelop, double-click any C# script in your Project pane
  - This will launch MonoDevelop
  - Though the launch process takes some time



- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team
- To open MonoDevelop, double-click any C# script in your Project pane
  - This will launch MonoDevelop
  - Though the launch process takes some time
- You must save a document in MonoDevelop for it to recompile and update in Unity



- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team
- To open MonoDevelop, double-click any C# script in your Project pane
  - This will launch MonoDevelop
  - Though the launch process takes some time
- You must save a document in MonoDevelop for it to recompile and update in Unity
- On Windows, Microsoft Visual Studio may be used

- Unity uses MonoDevelop for code editing
  - MonoDevelop is a separate program developed by a different team
- To open MonoDevelop, double-click any C# script in your Project pane
  - This will launch MonoDevelop
  - Though the launch process takes some time
- You must save a document in MonoDevelop for it to recompile and update in Unity
- On Windows, Microsoft Visual Studio may be used
  - Instructions for this can be found online











## **Attaching Scripts to GameObjects**



#### To work in Unity, a C# script must be attached to a GameObject



## **Attaching Scripts to GameObjects**



#### This makes the script a *component* of the GameObject





 You make use of Start() and Update() in the HelloWorld project



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}
    - Called every frame



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}
    - Called every frame
    - This can happen over 200 times per second!



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}
    - Called every frame
    - This can happen over 200 times per second!
  - void Awake() {...} (not used in HelloWorld, but important)



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}
    - Called every frame
    - This can happen over 200 times per second!
  - void Awake() {...} (not used in HelloWorld, but important)
    - Called once



- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}
    - Called every frame
    - This can happen over 200 times per second!
  - void Awake() {...} (not used in HelloWorld, but important)
    - Called once
    - Called at the moment the GameObject is created

- You make use of Start() and Update() in the HelloWorld project
  - void Start() {...}
    - Called once
    - Called immediately before the first Update() is called
  - void Update() {...}
    - Called every frame
    - This can happen over 200 times per second!
  - void Awake() {...} (not used in HelloWorld, but important)
    - Called once
    - Called at the moment the GameObject is created
    - Guaranteed to be called before Start()



 A prefab is a mold from which GameObject instances can be made



- A prefab is a mold from which GameObject instances can be made
  - Created by dragging a GameObject from the Hierarchy pane into the Project pane



- A prefab is a mold from which GameObject instances can be made
  - Created by dragging a GameObject from the Hierarchy pane into the Project pane
  - Can be assigned to a script variable in the Inspector pane



- A prefab is a mold from which GameObject instances can be made
  - Created by dragging a GameObject from the Hierarchy pane into the Project pane
  - Can be assigned to a script variable in the Inspector pane

• public GameObject gameObjectPrefab;



- A prefab is a mold from which GameObject instances can be made
  - Created by dragging a GameObject from the Hierarchy pane into the Project pane
  - Can be assigned to a script variable in the Inspector pane
     public GameObject gameObjectPrefab;
  - Then, an instance of the prefab can be created in code


#### **GameObject Prefabs and Instantiation**

- A prefab is a mold from which GameObject instances can be made
  - Created by dragging a GameObject from the Hierarchy pane into the Project pane
  - Can be assigned to a script variable in the Inspector pane
    public GameObject gameObjectPrefab;
  - Then, an instance of the prefab can be created in code

• Instantiate( gameObjectPrefab );



#### **GameObject Prefabs and Instantiation**

- A prefab is a mold from which GameObject instances can be made
  - Created by dragging a GameObject from the Hierarchy pane into the Project pane
  - Can be assigned to a script variable in the Inspector pane
    •public GameObject gameObjectPrefab;
  - Then, an instance of the prefab can be created in code

•Instantiate( gameObjectPrefab );

 This is used in HelloWorld to create thousands of instances of a Cube GameObject prefab





Output "Hello World!" to the Console pane



- Output "Hello World!" to the Console pane
  - Once using Start()



#### Output "Hello World!" to the Console pane

- Once using Start()
- Many times using Update()



#### Output "Hello World!" to the Console pane

- Once using Start()
- Many times using Update()

Create a Cube prefab that reacts to gravity & physics



- Output "Hello World!" to the Console pane
  - Once using Start()
  - Many times using Update()
- Create a Cube prefab that reacts to gravity & physics
- Instantiate an instance of the Cube prefab



- Output "Hello World!" to the Console pane
  - Once using Start()
  - Many times using Update()
- Create a Cube prefab that reacts to gravity & physics
- Instantiate an instance of the Cube prefab
  - Once using Start()

- Output "Hello World!" to the Console pane
  - Once using Start()
  - Many times using Update()
- Create a Cube prefab that reacts to gravity & physics
- Instantiate an instance of the Cube prefab
  - Once using Start()
  - Many times using Update()



- Output "Hello World!" to the Console pane
  - Once using Start()
  - Many times using Update()
- Create a Cube prefab that reacts to gravity & physics
- Instantiate an instance of the Cube prefab
  - Once using Start()
  - Many times using Update()
    - This will create a cascade of thousands of Cube instances



- Output "Hello World!" to the Console pane
  - Once using Start()
  - Many times using Update()
- Create a Cube prefab that reacts to gravity & physics
- Instantiate an instance of the Cube prefab
  - Once using Start()
  - Many times using Update()
    - This will create a cascade of thousands of Cube instances
  - Over other physically-modeled objects



🗇 🖸 👌 Scene_2.unity - Hello World - PC, Mac & Linux Standalone			
C + C X *Center © Cobal Eavers • Came Dev •			
E Project	THIE Hierarchy	# Scene	···· O Impector
Cruss * (0	Cruze * (SyAll	Textured # AGE # 🕸 🗐 Gizenes * (87A)	
Cube Prefab	Capsule		<u>×</u>
Cubespanner  HelloWorld	Cube PrefabiClonel	, , , , , , , , , , , , , , , , , , ,	
ØScene_0	Cube Prefabi(Clone)	S	
@Scene_1	Cube Prefab(Clone)		
€QScene_2	Cube PrefabiClone)	. 1	a think
	Cube PrefabiClone)	•	•
	Cube Prefab(Clone)		
	Cube Prefab(Clone)		4.15
	Cube PrefabiClone)		14°
	Cube PrefabiClone)		14 j
	Cube Prefab(Clone)	and the second sec	
	Cube PrefabiClonel		
	Cube PrefabiCione)		
	Cube Prefab(Clone)		$\sim n \sim$
	Cube Prefab(Clone)		
	Cube PrefabiClone)		
	Cube PrefabiClone)	•	
	Cube Prefab(Clone)		
	Cube PrefabiClone)	•	
	Cube PrefabiClone)		
	Cube Prefabi(Clone)		
	Cube PrefabiClone) Cube PrefabiClone)	· · ·	
	Cube Prefab(Clone)	C Gave	
	Cube Prefabi(Clone)		dominate on may stors [ Crimes
	Cube PrefabiClone) Cube PrefabiClone)	50	adistics
	Cube Prefab(Clone)	Graphics:	290.4 FP5 (3.4m)
	Cube PrefabiClone)	Draw Calls: 7 Saved by	batching: 74
	Cube PrefabiClone)	Triu: 2.4k. Vertu: 2.8k Used Textures: 1 - 22k	
	* * * * * * * *	Render Textures: 0 - 0	B switches: 0
Console	Constanting from Manual Street	Screen: 684x428 = 3.3 VRAM usage: 3.3 M8 to	3.4 MB (0/ 1.00 CB
Cent Catality Cent in but Inter Install	Open Paper Log Open Eallor Log	V80 Tetal: 10 - 92.5 K8	
		Visible Skinned Meshes	: O Animations: 0
		Network: (no players ce	www.trab
			-

#### The final HelloWorld scene



 Hello World is a common first program to make in any new language



- Hello World is a common first program to make in any new language
- Unity projects are stored as many files in project folders on your hard drive



- Hello World is a common first program to make in any new language
- Unity projects are stored as many files in project folders on your hard drive
- MonoDevelop is used to edit code for Unity



- Hello World is a common first program to make in any new language
- Unity projects are stored as many files in project folders on your hard drive
- MonoDevelop is used to edit code for Unity
- Scripts must be attached to GameObjects to run



- Hello World is a common first program to make in any new language
- Unity projects are stored as many files in project folders on your hard drive
- MonoDevelop is used to edit code for Unity
- Scripts must be attached to GameObjects to run
- Start(), Update(), and Awake() are called at different times and have different uses



- Hello World is a common first program to make in any new language
- Unity projects are stored as many files in project folders on your hard drive
- MonoDevelop is used to edit code for Unity
- Scripts must be attached to GameObjects to run
- Start(), Update(), and Awake() are called at different times and have different uses
- GameObject prefabs can be instantiated many times