Introduction to Unity

Games courses at Falmouth

Games courses at Falmouth

- **Studio-based**: 100% coursework, mainly team-based game development projects

Games courses at Falmouth

- **Studio-based**: 100% coursework, mainly team-based game development projects
- BA Game Development

Games courses at Falmouth

- **Studio-based**: 100% coursework, mainly team-based game development projects
- BA Game Development
  - Six routes: animation, art, audio, design, programming, writing

Games courses at Falmouth

- **Studio-based**: 100% coursework, mainly team-based game development projects
- BA Game Development
  - Six routes: animation, art, audio, design, programming, writing
- BSc Computing for Games

Games courses at Falmouth

- **Studio-based**: 100% coursework, mainly team-based game development projects
- BA Game Development
  - Six routes: animation, art, audio, design, programming, writing
- BSc Computing for Games
- BA Game Art

What is Unity?

What is Unity?

- A **game engine**

What is Unity?

- A **game engine**
- Two parts:

What is Unity?

- A **game engine**
- Two parts:
  - **Editor**: used by developers to create the game

What is Unity?

- A **game engine**
- Two parts:
  - **Editor**: used by developers to create the game
  - **Player**: packages the game to run on the customer's machine

Why Unity?

Why Unity?

- It's **cross platform**

Why Unity?

- It's **cross platform**
  - Editor runs on Windows and Mac

Why Unity?

- It's **cross platform**
  - Editor runs on Windows and Mac
  - Games can run on Windows, Mac, Linux, Android, iOS, all major consoles, and web via HTML5

Why Unity?

- It's **cross platform**
  - Editor runs on Windows and Mac
  - Games can run on Windows, Mac, Linux, Android, iOS, all major consoles, and web via HTML5
- It's relatively (!) **easy to use**

Why Unity?

- It's **cross platform**
  - Editor runs on Windows and Mac
  - Games can run on Windows, Mac, Linux, Android, iOS, all major consoles, and web via HTML5
- It's relatively (!) **easy to use**
- Very flexible, can be **programmed** using C#

Why Unity?

- It's **cross platform**
  - Editor runs on Windows and Mac
  - Games can run on Windows, Mac, Linux, Android, iOS, all major consoles, and web via HTML5
- It's relatively (!) **easy to use**
- Very flexible, can be **programmed** using C#
- Has **asset pipelines** for artists, animators, musicians etc.

Why Unity?

- It's **cross platform**
  - Editor runs on Windows and Mac
  - Games can run on Windows, Mac, Linux, Android, iOS, all major consoles, and web via HTML5
- It's relatively (!) **easy to use**
- Very flexible, can be **programmed** using C#
- Has **asset pipelines** for artists, animators, musicians etc.
- **Asset store** to download sprites, 3D models, animations, C# scripts, ...

Made with Unity

https://madewith.unity.com/

Key concepts

Key concepts

- A game is made up of one or more **scenes**

Key concepts

- A game is made up of one or more **scenes**
- Each scene contains a number of **game objects**

Key concepts

- A game is made up of one or more **scenes**
- Each scene contains a number of **game objects**
- Each game object has several **components** which affect how it looks, moves and behaves

Key concepts

- A game is made up of one or more **scenes**
- Each scene contains a number of **game objects**
- Each game object has several **components** which affect how it looks, moves and behaves
- Programmers can create custom components called **behaviours**

Key concepts

- A game is made up of one or more **scenes**
- Each scene contains a number of **game objects**
- Each game object has several **components** which affect how it looks, moves and behaves
- Programmers can create custom components called **behaviours**
- NB: behaviours can create or delete game objects, or change scenes...

Key concepts

- A game is made up of one or more **scenes**
- Each scene contains a number of **game objects**
- Each game object has several **components** which affect how it looks, moves and behaves
- Programmers can create custom components called **behaviours**
- NB: behaviours can create or delete game objects, or change scenes...
- No main loop, but behaviours have an `Update` method which is called each frame

What to make?

http://www.ludocraft.com/gigster/index.html

What to make?

What to make?

- Think simple — **don't overscope**!

What to make?

- Think simple — **don't overscope**!
- Work in **teams**

What to make?

- Think simple — **don't overscope**!
- Work in **teams**
- Focus on **core mechanics** — what will the player do in your game, and why is that fun?

What to make?

- Think simple — **don't overscope**!
- Work in **teams**
- Focus on **core mechanics** — what will the player do in your game, and why is that fun?
- Think about your **unique selling points (USPs)** — what will make your game different from all the others?

What to make?

- Think simple — **don't overscope**!
- Work in **teams**
- Focus on **core mechanics** — what will the player do in your game, and why is that fun?
- Think about your **unique selling points (USPs)** — what will make your game different from all the others?
- Aim for a **minimum viable product** — don't worry about polishing yet

Unity tutorial

http://bit.ly/unity_roll_a_ball