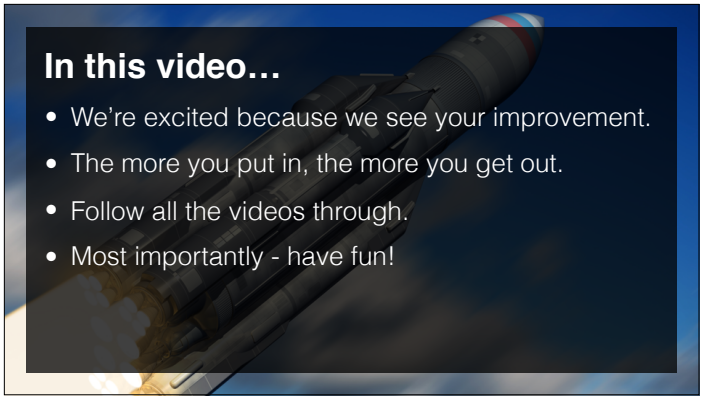

Welcome To The Course


In this video…
• We're excited because we see your improvement.
• The more you put in, the more you get out.
• Follow all the videos through.
• Most importantly - have fun!


How To Use This Course

In this video…

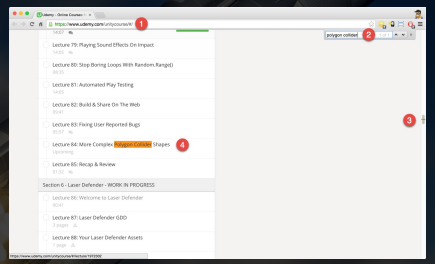
- Get Unity downloading now, it's big.
- Challenge 1: Do a whole section alone.
- Challenge 2: Large challenge videos.
- Challenge 3: Mini-challenges.

Making use of these PDF slides

- Think of the trees before you print.
- You can click on links in PDF.
- Also, remember to use search.

An overview of the interface…

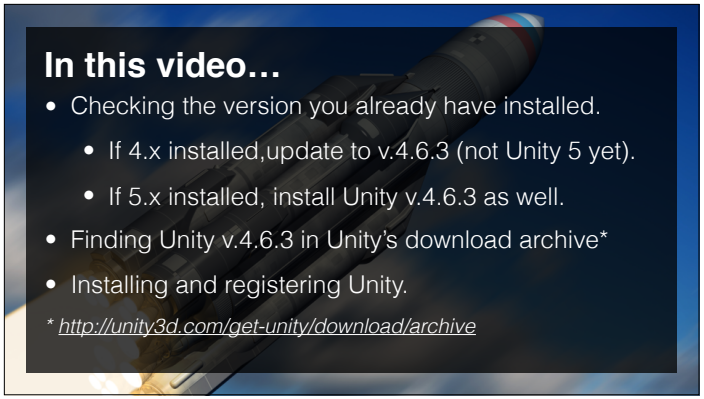
How to search for things



In summary…

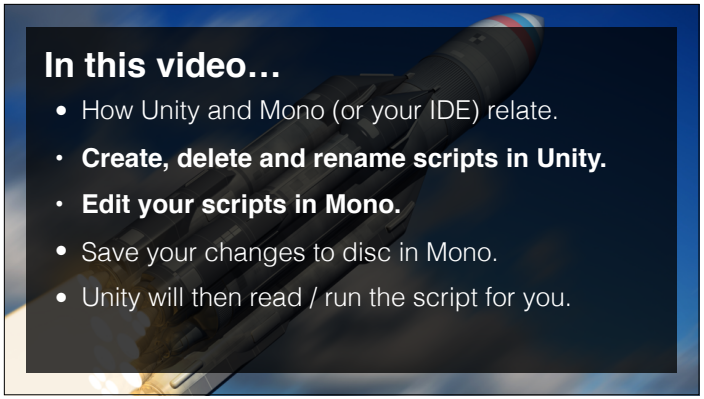
- Section notes are for reference, no action needed.
- Please do **at least** the mini-challenges.
- Assets are near the start of each section.
- Post feedback against specific lecture.
- Consider HD video, and Closed Captions (CC)

Installing Unity

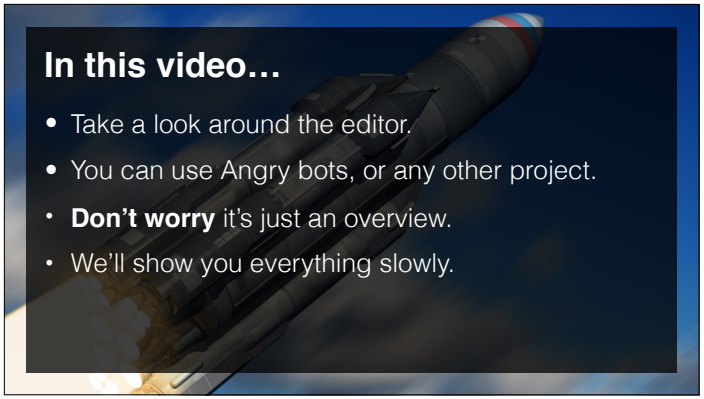

In this video…
• Checking the version you already have installed.
• If 4.x installed,update to v.4.6.3 (not Unity 5 yet).
• If 5.x installed, install Unity v.4.6.3 as well.
• Finding Unity v.4.6.3 in Unity's download archive*
• Installing and registering Unity.
* http://unity3d.com/get-unity/download/archive

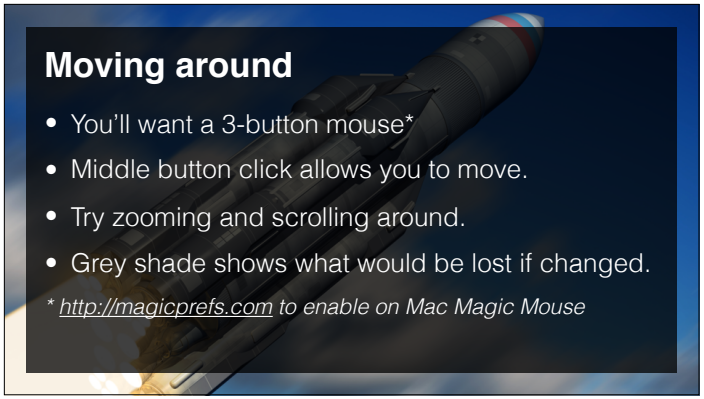

Get Unity 4.6.X Running
• Your very first mini-challenge!
• If you have problems try…
• Checking you have 10GB+ free space.
• Rebooting your machine.
• Installing again.

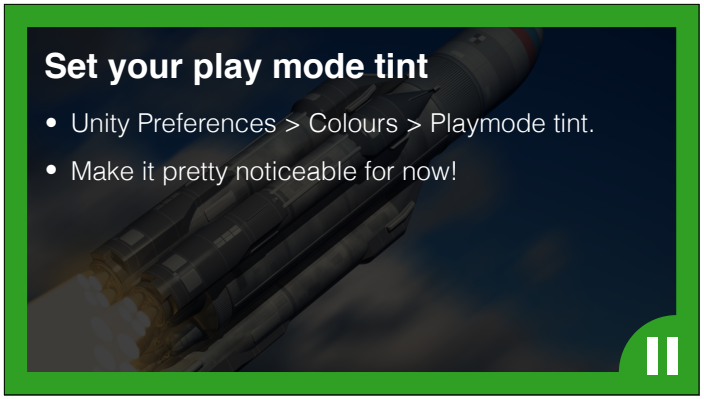

Unity, Your IDE and Your Files


In this video…
How Unity and Mono (or your IDE) relate.
Create, delete and rename scripts in Unity.
Edit your scripts in Mono.
Save your changes to disc in Mono.
Unity will then read / run the script for you.


Introducing The Unity Editor


In this video…
Take a look around the editor.
You can use Angry bots, or any other project.
Don’t worry it’s just an overview.
We’ll show you everything slowly.


Moving around
• You'll want a 3-button mouse*
• Middle button click allows you to move.
• Try zooming and scrolling around.
• Grey shade shows what would be lost if changed.
* http://magicprefs.com to enable on Mac Magic Mouse


Set your play mode tint
• Unity Preferences > Colours > Playmode tint.
• Make it pretty noticeable for now!


MonoDevelop 101

Where to find MonoDevelop Preferences

- In Windows go to Tools > Options*
- In MacOS go to MonoDevelop > Preferences.

**Thanks to Thelmo.*

Setting-up MonoDevelop

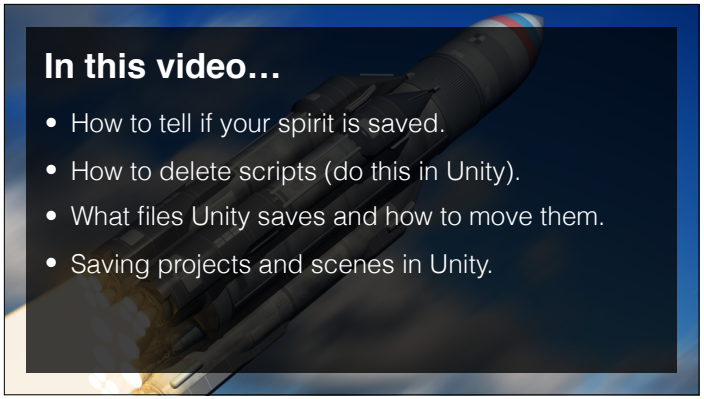
- Go into MonoDevelop Preferences.
- Text Editor > Behaviour > Indentation > Automatic
- Take a look at the keyboard shortcuts.

MonoDevelop doesn't start on Win8.1

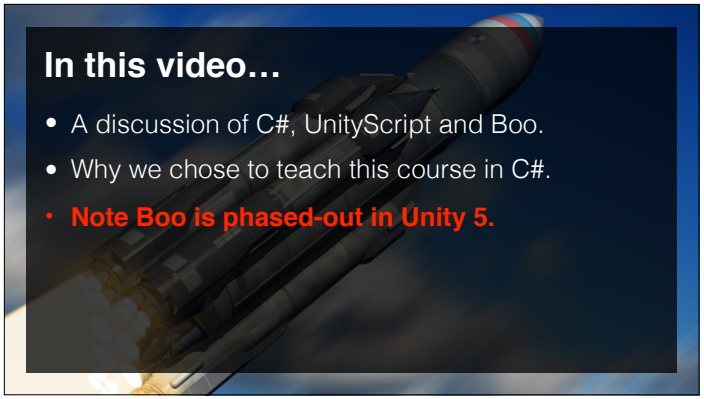
- You may be able to solve the problem by downloading a new version of **glibsharpglue-2.dll** in your **Unity\Monodevelop\bin** Folder.
- Find out more on the Unity Answers forum.

Thanks to Efim!

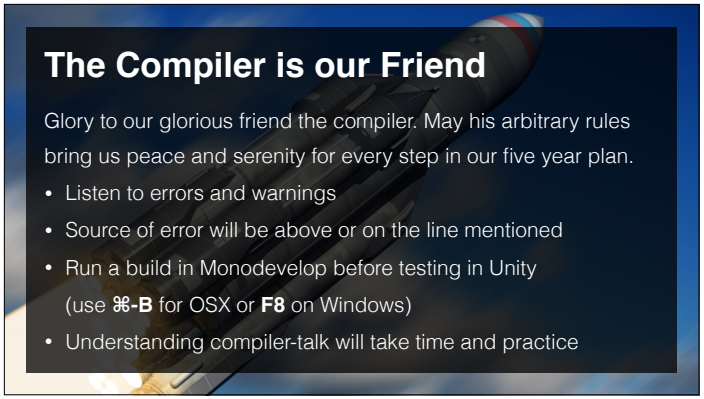

Saving & Closing Your Project


In this video…
• How to tell if your spirit is saved.
• How to delete scripts (do this in Unity).
• What files Unity saves and how to move them.
• Saving projects and scenes in Unity.


How The 3 Languages Relate

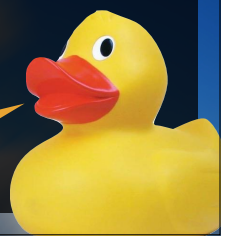

In this video…
• A discussion of C#, UnityScript and Boo.
• Why we chose to teach this course in C#.
• Note Boo is phased-out in Unity 5.


How To Debug Small Programs


The Compiler is our Friend
Glory to our glorious friend the compiler. May his arbitrary rules bring us peace and serenity for every step in our five year plan.
• Listen to errors and warnings
• Source of error will be above or on the line mentioned
• Run a build in Monodevelop before testing in Unity (use ⌘-B for OSX or F8 on Windows)
• Understanding compiler-talk will take time and practice

The Rubber Duck Sensei

- The four steps of rubber duck debugging: www.rubberduckdebugging.com



Assume Nothing!

- Most bugs are due to incorrect assumptions
- List you assumptions before and after a method is called
- Prove your assumptions (copious amounts of **print()** are called for)
- You can write the assumptions down in the code as comments, or better yet as tests

Remember the Mars Climate Orbiter: en.wikipedia.org/wiki/Mars_Climate_Orbiter

The Minimum Viable Test Case

- Remove code until you have the minimum possible amount of code that exhibits the problem
- Remove code step by step and check for the issue every time
- At some point, you'll remove the line of code that causes the issue.
- Less code is also much easier to reason about.

Let's do science!

Use the scientific method for Debugging

1. Make an hypothesis
2. Develop a testable prediction
3. Test the prediction
4. Repeat until tests concur with hypothesis

Further Resources

- Eric Lippert's great *How to debug small programs* tinyurl.com/howto-debug
- John Regehr's *Scientific debugging:* tinyurl.com/scientific-debugging

How To Ask Good Questions

Why this video

- Making it easier for someone to answer means you're more likely to have your problem solved
- The process will often solve problems before the question is even asked
- The attitude isn't just appropriate for game development or programming, but for life in general

Good admin

- Mark your question as **[Help]** and **[Solved]** at the beginning of the title
- Use a clear and descriptive title
- Use full sentences with correct grammar and punctuation

Write a good description

- Describe your problem thoroughly, format it as a bug report:
  - Observed behaviour
  - Expected behaviour
  - Steps to reproduce
  - List of things you've tried out already

Include all relevant information

- Include errors and warning in full
- Include version information
- Use Screenshots generously to show the problem. Take screenshots of both the entire window and the specific issue.

Use a code hosting service

- Code formatting on general purpose forums leaves much to be desired
- Use the right tool for the job:
  - http://pastebin.com
  - https://gist.github.com/
- Post complete files, point people to the appropriate line
- Screenshots are a reasonable alternative, especially if you

Create a Minimum Viable Test Case

- Remove code that is not relevant to your problem
- Make sure it still runs and exhibits the problem
- This has several effects
  - Often solves the problem by highlighting the issue
  - Makes reasoning about the problem much easier
  - Makes it easier for others to find the problem

The Golden Rule

- Write a question in the way you'd like to be asked
- Don't ask until you've tried 3 different things to solve the problem yourself

The Secret

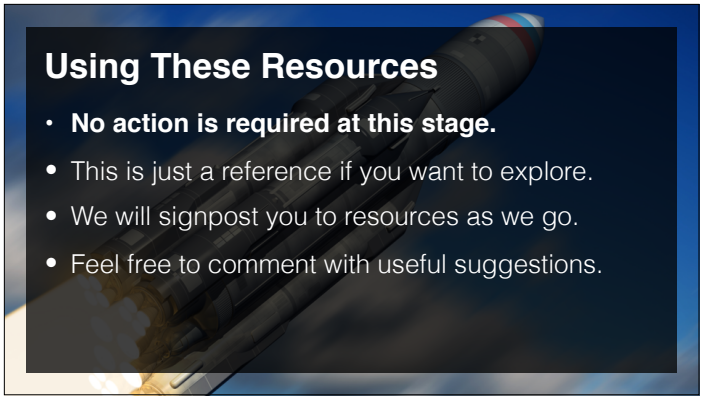
The effort spent asking a better question solves the problem 75% of the time!

Asking better questions makes you a better programmer!

Further Resources

- Stack Overflow's *How to ask a good question*: http://stackoverflow.com/help/how-to-ask
- Jon Skeet's *Writing the perfect question*: http://tinyurl.com/stack-hints
- Eric S. Raymond's *How To Ask Questions The Smart Way:* https://tinyurl.com/smart-questions


Useful Resources


Using These Resources
• No action is required at this stage.
• This is just a reference if you want to explore.
• We will signpost you to resources as we go.
• Feel free to comment with useful suggestions.