Section Notes

Number Wizard (Console)

Intro

Scripting is often intimidating when you start off. It's difficult to know what you're looking at and easy to get stuck. Which is exactly why this is where we'll start. Getting the most uncomfortable bit out of the way first will let you relax into the rest of the course. After this project, you will feel a lot more confident that you can handle scripting in Unity, and have built a solid understanding of the basic concept. We'll also show you where to go for more information on what's available in Unity.

Number Wizard is a simple guess-the-number game, where the player secretly picks a number and the computer attempts to guess with as few attempts as possible.

What We're Going To Do

- Create a guess-the-number game that is played by using the keyboard
- Write our first script

What You'll learn

- Basic scripting concepts
- Getting user keyboard input

Extra Credits

There are many ways you can make this simple console game better…

- Make the messages more interesting or colourful?
- Guess a random number instead of half the range to make the game feel more natural?
- Make the Player choose the number range?

If you have some programming experience already, and found the section to be familiar, why don't you try to change your game to use a recursive version of the Binary Search algorithm?

Further Reading

So what's the difference between functions and methods anyway?

As you start learning about programing, you'll hear the words *function*, *method* and *procedure* or perhaps even *member function* used almost interchangeably. In addition, because all programming languages are slightly different, each term will mean slightly different things depending on the context in which they are used. If you want to get more specific the following are good guidelines:

- A **function** is a piece of code that has a name in your program and that you may call by name. For example, *miles_to_inches.* It may take some input and will return a value. In the case of *miles_to_inches*, it would take a distance in miles and return it in inches. All the data that a function will use will be explicitly passed to it or will come from the containing context.
- A **Method** is a piece of code that is attached to an object. It will be passed to the object it belongs to implicitly[1]. In C# and Unity Script, you can access the current object explicitly using the *this* keyword. A method will be able to change its object data too. Almost all of the code you will write in this course is in fact a method, even though the object the code belongs to will not be obvious.
- A **Procedure** is a piece of code that executes a series of commands, but does not necessarily return a value. In most modern languages, including all the ones used in this course, the distinction between Procedures and Functions does not exist, as functions allow you to execute commands anyway.

Additional resources

- The Unity Manual is a great resource that covers a lot of ground. It's highly recommended you take a look at it as you go along.
- The Unity Scripting Reference covers every corner of Unity Scripting, and will tell you exactly how to use every feature. Getting familiar with this resource will make looking things up much easier.
- The Course Community is where you'll be able to ask questions and share ideas with other students.

[1] some languages do this explicitly.