A Practical Introduction to Python

Lesson 3 of 3 - Rock, Paper, Scissors

Lesson Description

This unit of work is about students exploring and using textual programming in a way that lets them do successful things. It is not about making sure that they understand the syntax for a while loop or an if statement.

In this lesson students will try to complete a half-finished game of Rock, Paper, Scissors. They will have to add code one section at a time. It is a tough lesson and the students should appreciate that even partial success in this task is a worthy achievement.

You may find the odd student wants to try coding "Rock, Paper, Scissors, Lizard, Spock". If you don't understand the reference then you should try watching The Big Bang Theory.

Lesson Objectives

All students should learn:
• How to add code to an existing program
• How to generate a random number

Most students should learn:
• How to use if/elif statements
• The need for care with indentation

Some students might learn:
• How to use nested if statements (if statements within if statements)
• How to create an attractive user interface

Resources

• Software: Python 3.x
• Software: EasyGUI (easygui.sourceforge.net)

• rockPaperScissors.py
• Rock Paper Scissors instructions

Lesson Outline

Starter

Try playing a game of Rock, Paper, Scissors online. One of the first links in Google will take you to a great AI player that will actually show you the thought processes and how it learns and analyses your playing style (perhaps best explored in a different lesson).

You'd be amazed how many students don't know how to play the game.

Task 1

Students should download the resources and check how the program works so far. Initially the computer will always choose "Rock" and they may notice that the interface is a bit dull.

Task 2

This lesson should be a little more self-paced, with a set of printed (or online) instructions rather than just relying on you at the front. This will allow those that struggle to take their time and those who fly through to move on at their own pace.

Who says I can't differentiate?

Students should initially code the random number generator for the computer player.

Next they should use an if/elif statement to convert this number into a choice.

After that is the slightly complex task of working out the result of the round.

Extension

There are various potential extensions. Improving the user interface with appropriate images would be a good one, as would keeping a running score.

Plenary

I always struggle with plenaries, you'll probably do a better job than I would ;-)