

Random Numbers, Files, and Onwards from Here

Random Numbers

- How can we make our program produce random values?
- It is impossible for a computer to algorithmically produce a *truly* random value.
- There are algorithms that will generate *pseudo-random* values.
 - These values are 'good enough' - they look random to a person, and are evenly distributed amongst the space of possible values.
 - But they are generated by an algorithm, so the precise sequence of random values could be repeated using the same algorithm.
 - A *seed* value is used when starting the pseudo-random algorithm - for the same seed, the same sequence of pseudo-random values will be produced.
 - If we set a different seed each time we run our program - for example, by setting the seed to the current time - we will get a different sequence of values.
- The Python module `random` allows us to generate random numbers.
 - Remember, being a module, we need to first bring it into our code using `import random`.
 - The documentation for this module can be found at <https://docs.python.org/3/library/random.html>.
- When the module starts, it is automatically seeded using the current system time.
- We can set our own seed using `random.seed()`.
 - `random.seed()` will seed the generator with the current system time.
 - `random.seed(x)` will seed the generator with the value of `x`.
- There are many different functions, given in the documentation, to extract different kinds of random values. Some useful ones are:
 - `random.random()` - Returns a random floating point value between 0.0 (inclusive) and 1.0 (exclusive).
 - `random.randint(a,b)` - Returns a random integer between `a` and `b` inclusive.
 - `random.shuffle(x)` - Shuffle the values in list `x`. (This modifies `x`, rather than returning a new list).
- Exercise:
 - Try generating some random numbers! Test how using the same seed leads to the same output.

Files

- We can read and write files using Python.
- This is a built-in feature of the language, and does not require a module.
- To open a file, call the `open` function:
 - `myfile = open(filename, mode)`
 - `filename` should be a string containing the filename.
 - `mode` should be a string containing one of several possible values, indicating how the file should be treated:
 - `r` - File is read-only.
 - `w` - File is write-only, and will be overwritten completely.
 - `a` - File is write-only, but data will be added to the end.

- For the remaining sessions, we will provide a list of longer challenges and exercises, for you to pick and choose what to practice with.

We hope you have enjoyed the course!