LEEDS
BECKETT
UNIVERSITY

Introduction to Programming

Week 3

Tony Jenkins

`a.m.jenkins@leedsbeckett.ac.uk`

OBJECTIVE

This week we start to explore how programs process data.

Data is stored in *variables* - values that can be manipulated in a program.

Week 3
Pre-requisites

You should have viewed or read the lecture:

"Variables and Types"

in the Week 2 folder on MyBeckett.

You should be able to access the Python Interpreter on the PCs in the labs.

You should be able to run PyCharm, enter a program, run it, and find the output.

You should be able to store your code on GitHub, and push and pull any changes.

Week 3

Pre-requisites

Create, now, a *private* repo that you will use for your programming work.

Give it a serious name - we are going to keep this one!

Include `README`, `.gitignore` and even `LICENSE` if you want to.

You may want to share it with your tutor.

Clone it so that you can access it from your IDE.

Create a folder for this week's work.

Variables and Values

A Quick Recap

Variables store values.

Values stored in variables in a program have a *type*.

Programming languages provide a set of *primitive data types.*

Python provides four primitive types:

- ➢ Integers.
- ➢ Floating point numbers.
- ➢ Strings.
- ➢ Booleans.

```
2         # The integer 2.
2.0       # The floating-point number 2.
'2'       # The ASCII character 2.
'Two'     # A string of characters.

True      # Boolean truth value.
```

Variables and Values

A Quick Recap

Variables have names (correctly called identifiers).

Programming languages have rules for what can be used as an identifier.

Python variable identifiers are traditionally written in `snake_case`.

(Compare to Java which favours `camelCase`.)

It is important to follow conventions like this - your code will be read by others more often that it is run.

```
2        # The integer 2.
2.0      # The floating-point number 2.
'2'      # The ASCII character 2.
'Two'    # A string of characters.

True     # Boolean truth value.

bold_knights = 3
name = 'Sir Robin of Camelot'
required_shrubbery = True
```

Variables and Values

A Quick Recap

A variable is created when a value is assigned to it.

The value determines the variable's type.

The `type` function lets us find out what type a variable currently has.

```
2        # The integer 2.
2.0      # The floating-point number 2.
'2'      # The ASCII character 2.
'Two'    # A string of characters.

True     # Boolean truth value.

bold_knights = 3
name = 'Sir Robin of Camelot'
required_shrubbery = True

>>> type(required_shrubbery)
<class 'bool'>
>>> type(bold_knights)
<class 'int'>
```

Variables and Values

A Quick Recap

Variables may also be assigned the results of expressions.

The type is determined by the expressions.

The usual operators work.

Boolean operators also exist.

Some operators work with strings too.

Strings can also be sliced.

```
swallow_speed_mph = 15
swallow_speed_kph = swallow_speed_mph * 1.6

brave = True
from_camelot = True

brave_knight = brave or from_camelot

order = 'eggs' + ' ' + 'spam'
order = 'eggs' + ' ' + 'spam ' * 3

order = order[:-1]
```

Lists

A Compound Type

A programming language will also provide some more complex data types.

These can be used to store related primitive values, and allow for neater processing.

Python supports a few as standard: like *lists*.

```
>>> knights = []

>>> knights.append ('Robin')
>>> knights.append ('Galahad')

>>> knights
['Robin', 'Galahad']

>>> len(knights)
2

>>> 'Bedevere' in knights
False
```

Libraries

Useful Point

We are starting to meet some of the built-in features of Python.

But we are scratching the surface.

A common "newbie" mistake is to start coding something by hand where they is already a built-in way to do it.

If you find yourself coding something that should be simple, but isn't, stop and think.

Program
Exam Marks

A student has five marks on five exams, graded 0 to 100.

Write a program that reads these five marks, and outputs the highest, lowest, and mean.

Note that there are many ways to do this. There is no single correct solution, but some solutions are better than others!

Abstraction

Key Skill

Programming is not just about learning a language.

A key skill is *abstraction*, which basically means spotting that a new problem is actually something you've seen before.

Obviously as you gain experience the number of "new" problems you encounter reduces.

Program

Temperatures

A weather station takes readings throughout the day, from 8am to 8pm, at three hour intervals.

The temperature readings are stored with the scale "C" or "F" as the last character, although all readings are taken in Celsius ("C").

Write a program to find the maximum, minimum, range, and average temperatures through a day.

DRY
Key Idea

The programs from today contain code that is repeated.

This is a bad thing. It is boring to type, error-prone, and cumbersome to maintain.

DRY = DON'T REPEAT YOURSELF

WET = WASTE EVERYONE'S TIME
WET = WRITE EVERYTHING TWICE
WET = WE ENJOY TYPING

Program
Temperatures

A weather station takes readings throughout the day, from 8am to 8pm, at three hour intervals.

The temperature readings are stored with the scale "C" or "F" as the last character, although all readings are taken in Celsius ("C").

Write a program to find the maximum, minimum, range, and average temperatures through a day.

NEXT

We will look at control statements, and specifically about how they will enable us to write DRY code.

Thank you

LEEDS
BECKETT
UNIVERSITY