

# Assignment 1A

Summer 2017

---

## Task 1 - Hello York

This assignment will teach you how to open IDLE and print your name.

**Open a new document in IDLE and print Hello York to the screen.** Does it run? Make sure you know how to save and open the files you make. Ask if you need help.

**Top Tip:** The IDLE executes lines of code one at a time, as you type them in after the 'prompt', which looks like three arrows to the right (>>>). So when you type your code there and hit 'Enter', it will run on the next line, and the prompt will appear again for your next line.

## Task 2 - Results May Vary

Sometimes, remembering things is pretty important. Variables will let us do that, which you'll be doing in this assignment. Suppose you start an interactive session and **paste in this line to initialize the variables x, y, z:**

```
x = 3; y = 6; z = 7;
```

What will Python print as the value of the following expressions? Try to **find the result manually** and then **compare your result by typing the expression in the interpreter:**

```
x * 2
```

```
x ** 2
```

```
x * z
```

```
y / x
```

---

## Task 3 - Order of Operations

Recall from mathematics, that arithmetic expressions are evaluated according to BIDMAS. Python evaluates arithmetic expressions in a *similar* way.

**Try evaluating the following expressions** and make sure you understand the results:

`2 + 3 * 5`

`1 - 3 + 5`

`2 * 3 ** 2`

`4 / 2 + 8`

`1 + 2 * 3`

`4 - 3 - 1`

**Extension:** Try evaluating the example from the slides. What happens?

## Task 4 - Boolean Expressions

Sometimes we need to represent truth values. In this assignment, we'll be learning about booleans. **Make sure you understand how these evaluate for different values for p and q:**

`p and q`

`not (p and q)`

`(not p) and q`

## Task 5 - Strings

Python allows us to manipulate strings. You can **assign a string to a variable**:

```
word = "Test 123"
```

And then **find its length**:

```
len(word)
```

We can also **repeat it 3 times, and concatenate an exclamation mark**:

```
(word * 3) + "!"
```

