print("Hello, World!")
What will we cover?

- Variables;
- Doing calculations;
- Mixing numbers and strings;
- Introducing Python Turtle!
○ Define simple problems, and describe and follow a **sequence of steps** and decisions (algorithms) needed to solve them (**ACTDIP010**)
○ Implement simple digital solutions as visual programs with algorithms involving branching (decisions) and **user input** (**ACTDIP011**)
○ Recognise **different types of data** and explore how the same data can be represented in different ways (**ACTDIK008**)

**What does this cover?**
1 More about Variables
Using Multiple Variables

- New variables can be created from other variables

```python
firstname = input('What is your first name? ')
lastname = input('What is your last name? ')
fullname = firstname + ' ' + lastname
print('Hello, ' + fullname)
```
Test it out!
Try the first question now!
Python the Calculator
We can use Python to do calculations for us.

To calculate the number of seconds in a day

\[(60 \text{ seconds per minute} \times 60 \text{ minutes per hour} \times 24 \text{ hours per day})\] :

```python
print(60 * 60 * 24)
```
## Python the Calculator

<table>
<thead>
<tr>
<th>Name</th>
<th>Calculator</th>
<th>Python</th>
</tr>
</thead>
<tbody>
<tr>
<td>addition</td>
<td>+</td>
<td>+</td>
</tr>
<tr>
<td>subtraction</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>multiplication</td>
<td>×</td>
<td>*</td>
</tr>
<tr>
<td>division</td>
<td>÷</td>
<td>/</td>
</tr>
</tbody>
</table>
Numbers and Strings are different

- Variables help keep track of the calculations and avoid “magic numbers”.

```python
secs_min = 60
mins_hour = 60
hours_day = 24
secs_day = secs_min*mins_hour*hours_day
print(secs_day)
```
Numbers and Strings
Asking for numbers from the user

- `input` reads in a string from the user, regardless of whether the user enters a digit or a letter.

```python
a = input('Enter a number: ')
b = input('Enter another number: ')
print(a + b)
```

This would be similar to:
```python
print('5' + '6')
```
Instead of being added together as we might want, the strings are concatenated.

```
print('5' + '6')
```

→ 56

We need a way of converting the strings to numbers!
Asking for numbers from the user

- `input` reads in a string from the user, regardless of whether the user enters a digit or a letter.

```python
a = input('Enter a number: ')
b = input('Enter another number: ')
print(a + b)
```

This would be similar to:

```python
print('5' + '6')
```
Using `int` to convert a string to an integer

- We can use the `int` function to convert strings to *integers* (whole numbers).

```python
a = int(input('Enter a number: '))
b = int(input('Enter another number: '))
print(a + b)
```
Teacher aside: introducing functions

Now is quite a good time to have a discussion about functions. Students have already been using them, and now they can talk about how the work, consider:

- the `int` function takes an *argument* of a string;
- it *returns* an integer;
- nesting function calls (input *inside* `int`)
Test it out!

Try the second question now!
3 Python Turtle

The Turtle Moves!
The turtle Module

- A module is an additional library of code that can be imported for use in a program.
- It lets you use other variables and functions that are defined in that module without having to start from scratch.
The turtle Module

- In these questions, we’ll use programming to control a virtual turtle to draw things on the screen!
- `turtle` is a Python module, so we need to import its functions by putting this statement at the top of each program:

```python
from turtle import *
```
Turtles are best when moving!

It’s best to work through these notes directly.
Moving around

moves, it draws a line, so you can draw shapes (and art)!

```
from turtle import *
forward(100)
right(120)
forward(100)
right(120)
forward(100)
```

Here we've drawn a triangle with 60° angles, and 100 turtle steps on each side. Since the sides are equal length, we've drawn an equilateral triangle.
Any Questions?

Find me at:
@groklearning
nicky@groklearning.com