MVA Microsoft Virtual Academy

Functions

Microsoft

Christopher Harrison | Content Developer
Susan Ibach | Technical Evangelist

Repetition

- One of the problems with code is you're frequently doing the same thing over and over again
  - The same few lines of code
  - The same tasks
  - The same operations

    - Again, and again, and again…

Repetition

- One of the problems with code is you're frequently doing the same thing over and over again
  - The same few lines of code
  - The same tasks
  - The same operations

    - Again, and again, and again…

Repetition

- One of the problems with code is you're frequently doing the same thing over and over again
  - The same few lines of code
  - The same tasks
  - The same operations

    - Again, and again, and again…

What if we could just create button that does the work?

- Then, just press that button

MVA Microsoft Virtual Academy

Introducing functions

Microsoft

What is a function?

- Function:
  - (Noun) A reusable section of code with a name that does something
  - Sometimes called a method

- You have already used functions!

  print

  open

  write

  close

Why create functions?

- Code reuse
  - You are doing the same thing over and over again
- Simplify your code
  - Functions have names to define what they do
  - Breakdown complex blocks of code
- Easier to make changes
  - If it's only been written once, you only have to update it once

How do you create a function?

- Use the keyword def
  - Short for define
- Give your function a name
  - You may also have parameter names (we will explain those shortly)
- Write the code in the body of the function

```
def printMessage():
    print('Hello World')
    return
```

How do you call a function?

- Simply use its name

```
def printMessage():
    print('Hello World')
    return

printMessage()
```

DEMO

Creating and calling functions

MVA Microsoft Virtual Academy

Parameters

Microsoft

I'd like to the function to be dynamic

- In our examples the functions we created only did one thing
  - Sometimes that's exactly what we need!
- But sometimes we need some flexibility
  - Create custom messages to be displayed
  - Provide two numbers for a calculation
  - Print to the screen, and optionally write to a file

To create a function that accepts data you use parameters

- A parameters is a piece of data passed into a function
- You've already used parameters!

```
print('Hello World')
```

- Inside the function, parameters behave like variables
  - It's a good idea to give them meaningful names

```
def printMessage(message):
    print(message)
    return

printMessage('Hello world!')
```

What about multiple parameters?

- Simply add them in, separated by commas

```
def displayMessage(greeting, name):
    message = greeting + ', ' + name
    print(message)
    return

displayMessage('Hi', 'Christopher')
```

DEMO

Add input paramters

MVA Microsoft Virtual Academy

Returning data

Microsoft

Functions return data using the keyword return

- Specify the value or data you want to pass back after the return keyword
- You can reuse names in different functions

```
def getMessage(name):
    message = 'Hello, ' + name
    return message

def printMessage(message):
    print(message)
    return

output = getMessage('Christopher')
printMessage(output)
```

Wait a minute...

- Did you just use the same name twice?

- After a while, you can't always use different names
- Functions are like containers for names
  - You can use the same name in different functions

DEMO

Returning values

Your challenge....

- Create a function to simplify writing to files.
- Set the function to accept parameters
  - one for text
  - one for the name of a file
- Add the code that will write the text out to the file.

Congratulations

- You can now save time coding by putting routine statements into functions

Microsoft