Functions

Exercises

Week 4

Prior to attempting these exercises ensure you have read the lecture notes and/or viewed the video, and followed the practical. You may wish to use the Python interpreter in interactive mode to help work out the solutions to some of the questions.

Download and store this document within your own filespace, so the contents can be edited. You will be able to refer to it during the test in Week 6.
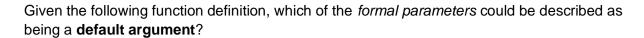
Enter your answers directly into the highlighted boxes.

For more information about the module delivery, assessment and feedback please refer to the module within the MyBeckett portal.

_____

_____

What must be done before a function that is not *built-in* to Python can be used in a program?

Answer:

Defining function by using def

_____

Given the following `import` statement, how would a call to the `sin()` function be made?

```
import math
```

Answer:

math.sin()

_____

Given the following `import` statement, how would a call to the `sqrt()` function be made?

```
from math import sqrt
```

Answer:

sqrt(number)

_____

What is the name of the common library that is available with all Python distributions?

Answer:

Python standard library

_____

What keyword is used in Python to define a new function?

Answer:

def

_____

Write some Python code that defines a function called `print_header(msg)`. This should output the value provided by the '`msg`' parameter to the screen (prefixed by five asterisk '`*****`') characters.

Answer:

```
def print_header(msg):
    print("*****", msg , "*****")
```

_____

In the answer box below give an example of what the **docstring** may look like for the
`print_header(msg)` function.

Answer:

""" print a header with 5 asterisk """

_____

Where within a function definition should a **docstring** appear?

Answer:

Immediately after function definition

_____

What statement should appear within a function's code block to cause a specific value to be
passed back to the caller of the function?

Answer:

Return statement

_____

Write some Python code that defines a function called `find_min(a,b)` that returns the
smallest of the two given parameter values.

Answer:

```
def find_min(a,b):
   If a>b:
     return a
   else:
     return b
```

_____

Given the following function definition, which of the *formal parameters* could be described as being a **default argument**?

```
def shouldContinue(prompt, answer=False):
     # function body...
```

Answer:

Answer is default parameter and False is a default argument

Provide two example calls to the above function, one which provides a value for the *default argument*, and one that does not.

Answer:

A= shouldContinue("hello") #default argument
B= shouldContinue("hello", True) #no default argument

_____

State why following function definition would **not** be allowed.

```
def do_something(prefix="Message", prompt, answer=False):
     # function body...
```

Answer:

default parameters must follow non-default parameters in a function signature.

_____

What single character is placed directly before the name of a *formal parameter*, to indicate that a variable number of actual parameters can be passed when the function is called?

Answer:

Asterisk (*)

_____

What commonly used built-in function, which displays output on the screen, can take a **variable number** of arguments?

Answer:

Print()

_____

Is it valid for a function's parameter name to be prefixed by two asterisk characters '`**`' as shown below?

```
def send_output(**details):
    # function body...
```

Answer:

> Yes, it is valid

If present, what does this prefix indicate?

Answer:

> It indicates that it can collect any additional keyword arguments passed to the function and bundle it to a dictionary

_____

What is the name given to a small 'anonymous' function that must be defined using a single expression?

Answer:

> lambda

Give an example of such a function that calculates the *cube* of a given number (i.e. the value of the number raised to the power of three) -

Answer:

> cube_lambda = lambda x: x ** 3
> result = cube_lambda(number)
> print(f"cube of {number} is: {result}")

_____

Exercises are complete

Save this logbook with your answers. Then ask your tutor to check your responses to each question.