

## Solutions 2A

Summer 2017

---


### Task 1 - Defining Functions

This is the first task in which we now expect pupils to save their work to a file, run it, and then test it works by typing into the interpreter.



```
q1.py - /Users/graham/Desktop/q1.py (3.6.0)
def square(n):
    return n ** 2
|
```

Ln: 3 Col: 0



```
Python 3.6.0 Shell
Python 3.6.0 (default, Dec 24 2016, 18:27:04)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/graham/Desktop/q1.py =====
>>> square(2)
4
>>> square(3)
9
>>> |
```

Ln: 10 Col: 4

## Task 2 - Calling Functions

The pupils now need to reuse their existing square function from question 1 in order to build a function that also adds one to the result. Appreciating abstraction is an important skill, but takes some time to develop.

```
q2.py - /Users/graham/Desktop/q2.py (3.6.0)
def square(n):
    return n ** 2

def square_add(n):
    return square(n) + 1
|
```

Ln: 6 Col: 0

```
Python 3.6.0 Shell
Python 3.6.0 (default, Dec 24 2016, 18:27:04)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/graham/Desktop/q2.py =====
>>> square_add(2)
5
>>> square_add(4)
17
>>>
```

Ln: 10 Col: 4

## Task 3 - Ifs and Functions

Pupils are now tasked with adding an elif block to the code given to them. They also have to recognize that there is no need for any other modifications to the code in order to achieve the task.

Note in particular how IDLE does not print the *None* for us by default. We can show it by calling print ourselves though.



```
def example(n):  
    if n > 10:  
        return True  
    elif n < 10:  
        return False
```

The screenshot shows a code editor window titled 'q3.py - /Users/graham/GitHub/Builder/q3.py (3.6.0)'. It contains a Python function 'example(n)' with two conditional branches: 'if n > 10: return True' and 'elif n < 10: return False'. The status bar at the bottom right indicates 'Ln: 6 Col: 0'.



```
Python 3.6.0 (default, Dec 24 2016, 18:27:04)  
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] on darwin  
Type "copyright", "credits" or "license()" for more information.  
>>>  
===== RESTART: /Users/graham/GitHub/Builder/q3.py =====  
>>> example(8)  
False  
>>> example(9)  
False  
>>> example(10)  
True  
>>> example(11)  
True  
>>> example(12)  
True  
>>> print(example(10))  
None  
>>>
```

The screenshot shows a Python 3.6.0 Shell window. It displays the output of several function calls: 'example(8)' returns 'False', 'example(9)' returns 'False', 'example(10)' returns 'True', 'example(11)' returns 'True', and 'example(12)' returns 'True'. Finally, 'print(example(10))' is called, which outputs 'None'. The status bar at the bottom right indicates 'Ln: 12 Col: 4'.

---

## Task 4 - All Together Now

The final task is designed to test the understanding of functions. Once the idea is grasped, this task is straightforward, however, may prove to be difficult initially.

```
q4.py - /Users/graham/Desktop/q4.py (3.6.0)

def square(n):
    return n ** 2

def example(n):
    if n > 10:
        return square(n)
    elif n < 10:
        return False

Ln: 9 Col: 0
```

```
Python 3.6.0 Shell
Python 3.6.0 (default, Dec 24 2016, 18:27:04)
[GCC 4.2.1 Compatible Apple LLVM 8.0.0 (clang-800.0.42.1)] on darwin
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /Users/graham/Desktop/q4.py =====
>>> example(9)
False
>>> example(10)
>>> example(11)
121
>>> example(12)
144
>>> |

Ln: 13 Col: 4
```