TMTplus Introduction to Scientific Programming

Mahdi Farnaghi, Mahdi Khodadadzadeh & Robert Ohuru

March 2021

Answers 3

Conditionals

Ex 3.2

Explanation of the code:

code lines	predicted output	explanation
`def is_even(x):` `  return x % 2 == 0`	`No output`	`The first two lines define the is_even function. This function takes one number as argument and then returns the result as True or False, depending on the result of the comparison: x % 2 == 0` `This comparison is the even test. First it computes the remainder of the argument value by 2 and then checks if that remainder is 0 or not. If the remainder is 0 then the comparison is True otherwise is False.`
`a = is_even(22)` `print(a)`	`True`	`The first line assigns the result of the function is_even with the argument value of 22 to the variable a.` `The second lines prints the variable.`
`b = is_even(21)` `print(b)`	`False`	`A similar procedure is done for variable b with argument value of 21.`

Ex 3.3 The return statement stops the execution of a function and returns a value from that function. In python if there is no return statement in the function python will return None in the end of the function.

Ex 3.4 A Python script that reports a pass or fail depending on given mark.

```
mark = int(input("What␣is␣the␣mark?␣"))
if mark >= 55:
    result = "pass"
else:
    result = "fail"
print(result)
```

Ex 3.5 A Python script that reports a pass, fail or excellent depending on given mark.

```
mark = int(input("What␣is␣the␣mark?␣"))
category = ""
if mark >= 55:
    result = "pass"
    if mark < 80:
        category = "average"
    else:
        category = "exceptional"
else:
    result = "fail"
print(result, category)
```

Ex 3.6 A Python script as a function called day_of_week() that maps the number of the week to a string using nested conditionals:

```
def day_of_week(n):
  if n == 0:
    print("Monday")
  else:
    if n == 1:
        print("Tuesday")
    else:
        if n == 2:
            print("Wednesday")
        else:
            if n == 3:
                print("Thursday")
            else:
                if n == 4:
                    print("Friday")
                else:
                    if n == 5:
                        print("Saturday")
                    else:
                        if n == 6:
                            print("Sunday")
```

```
                        else:
                            print("Error!")

a = int(input("What␣is␣the␣day?␣"))
day_of_week(a)
```

A Python script as a function called day_of_week() that maps the number of the week to a string using chained conditionals: Ex 3.7

```
def day_of_week(n):
    if n == 0:
       print("Monday")
    elif n == 1:
       print("Tuesday")
    elif n == 2:
       print("Wednesday")
    elif n == 3:
       print("Thursday")
    elif n == 4:
       print("Friday")
    elif n == 5:
       print("Saturday")
    elif n == 6:
       print("Sunday")
    else:
        print("value␣not␣correct")

a = int(input("What␣is␣the␣day?␣"))
day_of_week(a)
```