Control Structures

January 31, 2022

1 IT Academy - Data Science Itinerary

1.1 S02-T03: Control Structures

1 :

`Create a program that rates a numeric variable based on the Failed / Passed / Notable / Excell`

- grade list:
  - grade: >= 9 ===> excellent
  - grade: beetwen 7 and 8 ===> Notable
  - grade: beetwen 5 and 6 ===> Passed
  - grade: < 5 ===> Failed

we will write the function based in the grade list

[1]:
```
def grade_classification(grade):
    """ grade_classification() is a function that takes a numerical grade as␣
 ↪input and returns a rate based on the Failed / Passed / Notable / Excellent␣
 ↪scale. """
    if type(grade) == int or type(grade) == float:

        if grade < 0 or grade > 10:
            return ("Invalid grade")

        elif grade >= 9:
            return ("Excellent")

        elif grade >= 7:
            return ("Notable")

        elif grade >= 5:
            return ("Passed")

        if grade < 5:
            return ("Failed")
    else:
        print("Invalid grade. Enter a numerical grade")
```

testing the function:

[2]:
```
import random

random_grade = random.sample(range(10), 7)
print(random_grade)

for grade in random_grade:
    print(grade, grade_classification(grade))
```

```
[2, 5, 3, 7, 8, 1, 0]
2 Failed
5 Passed
3 Failed
7 Notable
8 Notable
1 Failed
0 Failed
```

2 :

Create a program that asks for two numbers. It should show a message saying if the first one is

[3]:
```
def check_numbers():
    """ check_numbers() is function that requests the user to enter two␣
↪interger numbers and compares them
    saying if the first one is bigger, the second one is bigger or they are the␣
↪same"""

    num_1 = input("enter your first number: ")

    while num_1.isdigit() == False:
        print(num_1," is not a interger number, write a number")
        num_1 = input("Try again! enter your first number: ")


    num_2 = input("enter your second number: ")

    while num_2.isdigit() == False:
        print(num_2," is not a interger number, write a number")
        num_2 = input("Try again! enter your second number: ")

    if int(num_1) == int(num_2):
        print("The first number",num1,"is the same as the second number",num_2)

    elif int(num_1) > int(num_2):
        print("The first number",num_1,"is bigger than the second number",␣
↪num_2)

    else:
```

```
        print("The second number",num_2,"is bigger than the first number",num_1)
```

[4]:
```
check_numbers()
```

```
enter your first number: 3
enter your second number: 6
The second number 6 is bigger than the first number 3
```

3 :

Create a program that asks the user for a name and number. If the number is 0, it should displa
"Joan Joan Joan".

[5]:
```
def name_repeated():
    """name_reapeated() is a fuction that requests the user to input a name,␣
↪and then input a number.
    The name is displayed repeatedly for the number of times the user indicates.
↪"""
    name = input("Enter your name: ")

    number = input("Enter a number: ")
    while number.isdigit() == False:
        print(number," is not a interger number, write a number")
        number = input("Try again! enter a number: ")

    number = int(number)

    if number <=0:

        while number <= 0:
            number = int(input("Please enter a positive number greater than␣
↪zero: "))
    print ((name + " " )* number)
```

[6]:
```
name_repeated()
```

```
Enter your name: Harry
Enter a number: 8
Harry Harry Harry Harry Harry Harry Harry Harry
```

4 :

Create a program that given any list, tells the user if it is symmetrical or not.
If so, tell the user how many items it has.

[7]:
```
def is_symmetrical(list):
    """ is_symmetrical(list) is a fuction takes a list as input and reports
    whether or not the list is symmetric.
    If symmetric, it also states the number of items in the list"""
```

```
    if list == list[::-1]:
         print("The list is symmetric, and it has", len(list), "items.")
    else:
        print("The list is not symmetric.")
```

[8]:
```
list_1 = ["a","b","c"]
list_2 = ["a","b","a"]
```

[9]:
```
is_symmetrical(list_1)
```

```
The list is not symmetric.
```

[10]:
```
is_symmetrical(list_2)
```

```
The list is symmetric, and it has 3 items.
```

5 :

Create a program that given a list, tells the user how many numbers match thier position. For example, in the list [3,4,2,0,2,3,6], numbers 2 and 6 coincide.

[11]:
```
def index_list_match(my_list):
    """index_list_match(list) is function that takes a list as input and␣
 ↪reports
    how many numbers in the list match their index position """

    i = 0
    match_items = []

    for item in my_list:
        if my_list.index(item) == item:

            match_items.append(item)
            i += 1
    if i > 0 :
        print ("The list has", i, "numbers equal to its index")
        print ("items that match are:", match_items)
    else:
        print("No elements equal to index found")
```

[12]:
```
a = [0,1,2,3,4,5]
index_list_match(a)
```

```
The list has 6 numbers equal to its index
items that match are: [0, 1, 2, 3, 4, 5]
```

[13]:
```
b = [2,3,4,5,"a",5,6,7]
index_list_match(b)
```

```
The list has 2 numbers equal to its index
items that match are: [6, 7]
```

[ ]: