JLUFE Fall 2021(Sep-Jan)

Homework Assignment Report

JILIN UNIVERSITY OF FINANCE AND ECONOMICS

College of Managment Science and Information Engineering

BSc in Data Science and Big Data Technology

(2021)

MODULE: Intelligent Technology

Homework Assignment: 04

Flow Control Statements

21/10/2021

Submitted by:

Milan(米兰) 0318021907632 (2005)
QQ: 3086215265 | Github ID: milaan9

Instructions:

1. I have added tips and required learning resources for each question, which helps you to solve the problems.
2. Finish the assignment on your **OWN**. **Any student find copying/sharing from classmates or internet will get '0' points!!!**
3. After Accepting this assignment from ➞ [GitHub Clasroom link (https://classroom.github.com/a/zeslyIXN)](https://classroom.github.com/a/zeslyIXN), Github will create private repository of the assignment in your GitHub Classroom account.
4. In your repository Clone ➞ Download ZIP in your computer.
5. Change your ➞ **College**, **Major**, **Name**, **Student number**, **Class number**, **QQ number** and **GitHub ID**
6. Once you finish the Assignment [convert your .ipynb file into PDF (https://github.com/milaan9/91_Python_Mini_Projects/tree/main/001_Convert_IPython_to_PDF)](https://github.com/milaan9/91_Python_Mini_Projects/tree/main/001_Convert_IPython_to_PDF) (both **.ipynb** and **.pdf** file will be required!)
7. To submit your assignment, go to GitHub Classroom repository and Add file ➞ Upload files ➞ Commit changes
   1. Replace the question (**.ipynb**) file with your solution (**.ipynb**) file.
   2. Also, upload (**.pdf**) converted file of your solution (**.ipynb**) file.

Python Assignment 04

Part A ➞ If-elif-else Statements Level 1, 2 & 3

> **Note:** Please create new cell for each question

Part A ➞ Level 1

> **Note:** Please create new cell for each question

1. Get two numbers from the user using `input()` prompt.

   - If `num_1` is greater than `num_2` return `num_1` is greater than `num_2` ,
   - if `num_1` is less `num_2` return `num_1` is smaller than `num_2` ,
   - else `num_1` is equal to `num_2` .

   -
     ```
     Enter number one: 9
     Enter number two: 6
     9 is greater than 6
     ```

In [ ]:

```
# Solution:
num1 = input("Enter number one:")
num2 = input("Enter number two:")
if num1 < num2 :
    print(num1 + "is less" + num2)
if num1 > num2 :
    print(num1 + " is greater than" + num2)
else:
    print(num1 + " is equal " + num2)
```

Part A ➞ Level 2

> **Note:** Please create new cell for each question

1. Write a code which gives grade to students according to theirs scores get from user `input()` :

   -
     ```
     80-100, A
     70-89, B
     60-69, C
     50-59, D
     0-49, F
     ```

2. Check if the season is `Autumn` , `Winter` , `Spring` or `Summer` .

   - If the user `input()` is:

- September, October or November, the season is Autumn.
- December, January or February, the season is Winter.
- March, April or May, the season is Spring
- June, July or August, the season is Summer

3. The following list contains some fruits:

- Taker user `input()` and if a fruit doesn't exist in the list add the fruit to the list and print the modified list. If the fruit exists print `('That fruit already exist in the list')`

  ```
  fruits = ['banana', 'orange', 'mango', 'pear']
  ```

In [16]:

```
score = int(input("Please enter the score:"))
if(90<=score<=100):
    print("Your grade is A")
elif(70<=score<=89):
    print("Your grade is B")
elif(60<=score<=69):
    print("Your grade is C")
elif(50<=score<=59):
    print("Your grade is D")
else:
    print("Your grade is E")
```

```
Please enter the score:65
Your grade is C
```

In [17]:

```
month = input("month:")
if month in ["September", "October" , "November"]:
    print("Autumn")
if month in ["December", "January" , "February"]:
    print("Winter")
if month in ["March", "April" , "May"]:
    print("Spring")
if month in ["June", "July" , "August"]:
    print("Summer")
```

```
month:June
Summer
```

In [18]:

```
fruit = input("fruit:")
fruits = ['banana', 'orange', 'mango', 'pear']
if fruit in fruits:
    print("That fruit already exist in the list")
else:
    fruits.append(fruit)
    print(fruits)
```

```
fruit:banana
That fruit already exist in the list
```

Part A ➞ Level 3

> **Note:** Please create new cell for each question

1. Here we have a person dictionary. Feel free to modify it!

-
  ```
  person={
  'first_name': 'Milaan',
  'last_name': 'Parmar',
  'age': 96,
  'country': 'England',
  'is_marred': True,
  'skills': ['Python', 'Matlab', 'R', 'C', 'C++'],
  'address': {
    'street': 'Space street',
    'zipcode': '02210'
  }
  }
  ```

    - Check if the person dictionary has `skills` key, if so print out the middle skill in the skills list.
    - Check if the person dictionary has `skills` key, if so check if the person has 'Python' skill and print out the result.
    - If a person skills has only Python and Matlab, print ('He knows machine learning'), if the person skills has Python, and R print ('He knows statistics'), if the person skills has C, and C++, Print ('He knows software development'), else print ('unknown title') - for more accurate results more conditions can be nested!
    - If the person is married and if he lives in England, print the information in the following format:

-
  ```
  Milaan Parmar lives in England. He is married.
  ```

In [20]:

```
# Solution:
person={
'first_name': 'Milaan',
'last_name': 'Parmar',
'age': 96,
'country': 'England',
'is_marred': True,
'skills': ['Python', 'Matlab', 'R', 'C', 'C++'],
'address': {
 'street': 'Space street',
 'zipcode': '02210'
}
}
if 'skills' in person.keys():
    print(person["skills"])
if 'skills' in person.keys():
    if 'Python' in person["skills"]:
        print("Yes")
    else:
        print("No")
    if 'Matlab' in person["skills"] and "Python" in person["skills"]:
        print('He knows machine learning')
    if 'R' in person["skills"] and "Python" in person["skills"]:
        print('He knows statistics')
    if 'C' in person["skills"] and "C++" in person["skills"]:
        print('He knows software development')
    else:
        print('unknown title')
if person["is_marred"] == True and person['country'] == 'England':
    print(person['first_name']+" "+person['last_name']+" "+"lives in"+" "+person['country']+"."+"He
```

```
['Python', 'Matlab', 'R', 'C', 'C++']
Yes
He knows machine learning
He knows statistics
He knows software development
Milaan Parmar lives in England.He is married
```

Part B ➞ Loops Level 1, 2 and 3

> **Note:** Please create new cell for each question

Part B ➞ Level 1

> **Note:** Please create new cell for each question

1. Iterate 0 to 10 using `for` loop, do the same using `while` loop.
2. Iterate 10 to 0 using `for` loop, do the same using `while` loop.
3. Write a code so we get on the output the following square by taking `input()` from user:

-
  ```
  # = # = # = # = #
  # = # = # = # = #
  # = # = # = # = #
  # = # = # = # = #
  # = # = # = # = #
  # = # = # = # = #
  # = # = # = # = #
  # = # = # = # = #
  ```

4. Use nested loops to create the following by taking `input()` from user:

```
      #
     ###
    #####
   #######
  #########
 ###########
#############
```

5. Print the following using loops by taking `input()` from user:

-
  ```
  0 x 0 = 0
  1 x 1 = 1
  2 x 2 = 4
  3 x 3 = 9
  4 x 4 = 16
  5 x 5 = 25
  6 x 6 = 36
  7 x 7 = 49
  8 x 8 = 64
  9 x 9 = 81
  10 x 10 = 100
  ```

6. Iterate through the list, `['Python', 'Numpy', 'Pandas', 'Scikit', 'Pytorch']` using a `for` loop and print out the items.
7. Use `while` loop to iterate from 0 to 100 and print the sum of all numbers.

-
  ```
  The sum of all numbers is 5050.
  ```

8. Use `for` loop to iterate from 0 to 100 and print the sum of all evens and the sum of all odds.

-
  ```
  The sum of all evens is 2550. And the sum of all odds is 2500.
  ```

In [ ]:

```

```

In [22]:

```
# Solution:
count = 0
for n in range(10):
    while(count < 10):
        count+=1
    count = 10
    while(count > 0):
        count-=1
for n in range(8):
    print("# = # = # = # = #")
```

```
# = # = # = # = #
# = # = # = # = #
# = # = # = # = #
# = # = # = # = #
# = # = # = # = #
# = # = # = # = #
# = # = # = # = #
# = # = # = # = #
```

In [27]:

```
n = input("high:")
n = int(n)
for m in range(n+1):
    for k in range(n-m):
        print(" ", end="")
    for k in range(2*m-1):
        print("#", end="")
    print()
```

```
high:7

      #
     ###
    #####
   #######
  #########
 ###########
#############
```

In [35]:

```
n = input("number:")
n = int(i)
for m in range(i):
    z = m^2
    z = str(z)
    m = str(m)
    print(m+"x"+m+"="+z)
```

```
number:7
0x0=2
1x1=3
2x2=0
3x3=1
4x4=6
5x5=7
6x6=4
```

In [31]:

```
list = ['Python', 'Numpy', 'Pandas', 'Scikit', 'Pytorch']
for n in range(len(list)):
    print(list[n])
```

```
Python
Numpy
Pandas
Scikit
Pytorch
```

In [36]:

```
count = 0
sum = 0
while(count<=100):
    sum = sum + count
    count = count + 1
print(sum)
```

```
5050
```

In [54]:

```
i=0
sum = 0
for i in range(0,101):
    if(i%2==0):
        sum=sum + i
print(sum)
```

```
2550
```

Part B ➞ Level 2

> **Note:** Please create new cell for each question

1. Use `for` loop to find fibonacci numbers from from 0 to 100 and print only even numbers from it. Also, find how many even numbers are in it.
2. Use `while` loop to find fibonacci numbers from from 0 to 100 and print only odd numbers from it. Also find how many odd numbers are in it.

Part B ➞ Level 3

> **Note:** Please create new cell for each question

1. Go to the data folder and use the **countries_data.py (https://github.com/milaan9/02_Python_Datatypes/blob/main/countries_data.py)** file. Loop through the countries and extract all the countries containing the word `land` .
2. This is a fruit list, `['banana', 'orange', 'mango', 'lemon']` reverse the order using loop.
3. Go to the data folder and use the **countries_details_data.py (https://github.com/milaan9/03_Python_Flow_Control/blob/main/countries_details_data.py)** file.
   A. What are the total number of languages in the data
   B. Find the ten most spoken languages from the data
   C. Find the 10 most populated countries in the world

In [58]:

```
# Solution:
a1= 1
a2= 1
print(a1)
print(a2)
for an in range(0,100):
    if an == a2 + a1:
        print(an)
        a1,a2 = a2,an
```

```
1
1
2
3
5
8
13
21
34
55
89
```

In [57]:

```
a1 = 0
a2 = 1
an = 1
while an < 100:
    print(an)
    an = a2 + a1
    a1,a2 = a2,an
```

```
1
1
2
3
5
8
13
21
34
55
89
```

In [ ]:

```
```