Python Lab - Assignment No :- 01
Decision Making and Control Statements

Date : 05/05/2021

Aim : Python programs to handle decision making and control statements.

Theory :

- **Variables :**

- **What is a Variable in Python?**
- A Python variable is a reserved memory location to store values.
- In other words, a variable in a python program gives data to the computer for processing.
- Every value in Python has a data type.
- Different data types in Python are Numbers, List, Tuple, Strings, Dictionary, etc.

- **Rules for Python variables :**
- A variable name must start with a letter or the underscore character.
- A variable name cannot start with a number.
- A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
- Variable names are case-sensitive (age, Age and AGE are three different variables).
- **Eg.** : Name, _abc, var1, etc.

- **Python Variable Types : Local & Global**

- There are two types of variables in Python, *Global variable and Local variable*.
- **Global Variables** : When you want to use the same variable for rest of your program or module you declare it as a global variable.
- **Local Variables** : While if you want to use the variable in a specific function or method, you use a local variable while Python variable declaration.
- **Eg**. :

```
car = BMW          #globalVariable
def func1():       #someFunction
year = 1960        # localVariable
print(car)
print(year)
```

Python Lab - Assignment No :- 01
Decision Making and Control Statements

➢ **Conditional Statements in Python :**

- **What is an 'if' statement in Python?**

- 'if' statements in python are known as decision making statements or conditional statements, since it helps us execute a statement 'if' the test expression evaluated is True.

  Syntax : if test expression:

  statement(s)

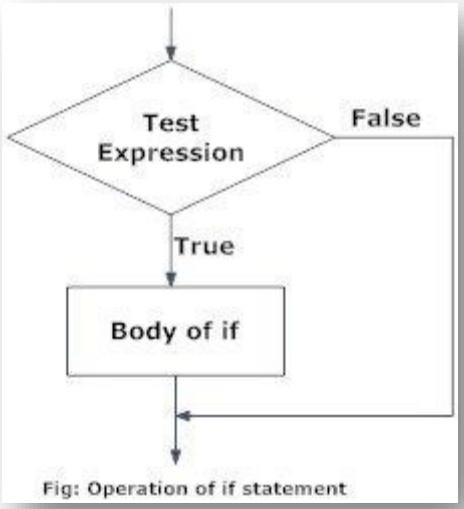

Fig: Operation of if statement

- If the expression is false, the statements are not executed.
- The body of the 'if' statement is indicated by an indentation and the first unindented line marks it's end.
- Non-zero values are interpreted as True while 'none' and '0' as False.

- **If..elif..else Statements :**

- elif is short term for else if, it allows us to check for multiple statements.
- The condition for the 'if' block is checked and evaluated first, if it is False, the elif block is evaluated and so on, and finally if all the conditions are False the statements in the 'else' block are executed.
- The *else* statement is an optional statement and there could be at most only one **else** statement following **if**.
-

```
Syntax : if expression1:
             statement(s)
         elif expression2:
             statement(s)
         elif expression3:
             statement(s)
         else:
             statement(s)
```

- **Code :**

1. Write a program that prompts the user to input a year and determine whether the year is leap or not.

```
print("Welcome to Leap Year Calculator")

year = int(input("Enter a valid year : "))

if (year%4) == 0:
```

```
  print(f"{year} is a leap year!")

else:

  print(f"{year} is not a leap year!")
```

- **Output :**

```
PS C:\Users\Nidhi\PythonS11> & C:/Users/Nidhi/AppData/Local/Microsoft/WindowsApps/PythonSoftwareFoundation
.Python.3.9_qbz5n2kfra8p0/python.exe c:/Users/Nidhi/PythonS11/17_Lab1.1.py
Welcome to Leap Year Calculator
Enter a valid year : 2020
2020 is a leap year!
PS C:\Users\Nidhi\PythonS11> & C:/Users/Nidhi/AppData/Local/Microsoft/WindowsApps/PythonSoftwareFoundation
.Python.3.9_qbz5n2kfra8p0/python.exe c:/Users/Nidhi/PythonS11/17_Lab1.1.py
Welcome to Leap Year Calculator
Enter a valid year : 2021
2021 is not a leap year!
PS C:\Users\Nidhi\PythonS11>
```

 # 2. Write a program that prompts the user to input a positive integer and the program displays the sum of the digits of the number.

```
num = int(input("Enter a postive integer :"))

temp = num

sum = 0

while temp > 0:

  digi = temp%10

  sum = sum + digi

  temp = int(temp/10)

print("Sum of digits of entered number = ",sum)
```

- **Output :**

```
PS C:\Users\Nidhi\PythonS11> & C:/Users/Nidhi/AppData/Local/Microsoft/WindowsApps/PythonSoftwareFoundation
.Python.3.9_qbz5n2kfra8p0/python.exe c:/Users/Nidhi/PythonS11/17_Lab1.2.py
Enter a postive integer :80391
Sum of digits of entered number =  21
PS C:\Users\Nidhi\PythonS11> & C:/Users/Nidhi/AppData/Local/Microsoft/WindowsApps/PythonSoftwareFoundation
.Python.3.9_qbz5n2kfra8p0/python.exe c:/Users/Nidhi/PythonS11/17_Lab1.2.py
Enter a postive integer :53662
Sum of digits of entered number =  22
```

 # 3. Write a program to print first 25 numbers of a Fibonacci series.

```
#print Fibonacci series for 1st 25 numbers.


print("Welcome to Fibonacci Calculator!")
fibin = int(input("Enter the number of terms required : " ))
num1 = 0
num2 = 1
#sum = num1+num2
count = 0
if fibin < 0:
  print("Invalid input..enter a positive integer")
elif fibin == 1:
  print("Fibonacci series for "f"{fibin} terms : ")
  print(num1)
else:
  print("Fibonacci series for "f"{fibin} terms : ")
  while count < fibin:
    print(num1)
    n = num1+num2
    num1 = num2
    num2 = n
    count += 1
```

- **Output :**

```
PS C:\Users\Nidhi\PythonS11> & C:/Users/Nidhi/AppData/Local/Microsoft/WindowsApps/PythonSoftwareFoundation
.Python.3.9_qbz5n2kfra8p0/python.exe c:/Users/Nidhi/PythonS11/17_Lab1.3.py
Welcome to Fibonacci Calculator!
Enter the number of terms required : 25
Fibonacci series for 25 terms :
0
1
1
2
3
5
8
13
21
34
55
89
144
233
377
610
987
1597
2584
4181
6765
10946
17711
28657
46368
```

Conclusion :

- Variables are referred to "envelop" or "buckets" where information can be maintained and referenced. Like any other programming language Python also uses a variable to store the information.
- Variables can be declared by any name or even alphabets like a, aa, abc, etc. following the rules.
- Variables can be re-declared even after you have declared them for once.
- Types of variables in Python or Python variable types : Local & Global.
- An **else** statement can be combined with an **if** statement. An **else** statement contains the block of code that executes if the conditional expression in the if statement resolves to 0 or a FALSE value.
- The **elif** statement allows you to check multiple expressions for TRUE and execute a block of code as soon as one of the conditions evaluates to TRUE.
