Name: PRAJVIN
Register Number:

**Ex. No: 02**

**Date:** 25.12.22

## Ex. No 2. Python programming using simple statements and expressions

**Aim**

The aim of this exercise is to write python script for the given program using simple statements and expressions and execute the same in Python IDLE (through Script or Development Mode).

**Problems Given**

1. Exchange the values of two variables
2. Circulate the values of n variables
3. Distance between two points

**Concepts Involved**

**Variable:** A variable is an identifier that refers to a value or variable is a memory location where a programmer can store a value. Example: roll_no, amount, name etc.

**Expression:** It is a combination of values, variables and operators.

**Statement:** A statement is a unit of code that the Python interpreter can execute.

**Reading Input:**

- Input often comes from the keyboard. It means the data entered by an end-user of the program.
- It has two key functions to deal with end user input called raw_input ( ) and input().
- raw_input:
    - The raw_input([prompt]) function reads one line from standard input and returns it as a string.
    - This prompts the users to enter any string and it would display same entered string on the screen.
    - raw_input function is not supported by higher versions of Python for Windows.
    - **Syntax:**

var_name=raw_input("<prompt input statement>")

**Example:**

book_name=raw_input("Enter your book name: ")

- input:
  - The input([prompt]) function is equivalent to raw_input, except that it assumes the input as a valid Python expression and returns the evaluated result. It has an optional parameter, which is the prompt string.

    **Syntax:**

    var_name=input("<prompt expression or statement>")

    **Example:**

    Name=input("Enter your name")

**Import Module:**

- Module is a built in file containing python definitions and statements with the .py extension, which implement a set of function.
- Modules are imported from other modules using the import command.
- When a module gets imported, it searches for the module and if found, python creates a module object. If the named module cannot be found, ModuleNotFoundError will be raised.
- Syntax: import modulename

  Example:

  import math

  >>>math.ceil(30.56)

  Output:

  31

  >>>math.floor(30.56)

  Output:

  30

Name: PRAJVIN
Register Number:

Script:

1. Exchange the values of two variables

In [1]:
```
#Exchanging the values of two variables
a=int(input("Enter value of a= "))
b=int(input("Enter value of b= "))
a,b=b,a
print("a,b=",a,b)
```

```
Enter value of a= 10
Enter value of b= 20
a,b= 20 10
```

2. Circulate the values of n variables

In [2]:
```
#Circulate the values of n variables
n=int(input("Enter the value of n="))
list1=[1,2,3,4]
print(list1[n:]+list1[:n])
```

```
Enter the value of n=1
[2, 3, 4, 1]
```

In [3]:
```
n=int(input("Enter the value of n="))
list1=[1,2,3,4]
print(list1[n:]+list1[:n])
```

```
Enter the value of n=2
[3, 4, 1, 2]
```

Name: PRAJVIN
Register Number:

In [4]:
```
n=int(input("Enter the value of n="))
list1=[1,2,3,4]
print(list1[n:]+list1[:n])
```

```
Enter the value of n=3
[4, 1, 2, 3]
```

In [5]:
```
n=int(input("Enter the value of n="))
list1=[1,2,3,4]
print(list1[n:]+list1[:n])
```

```
Enter the value of n= 4
[1, 2, 3, 4]
```

3. Distance between two points

In [6]:
```
#Distance between two points
import math
x1=int(input("Value of x1="))
x2=int(input("Value of x2="))
y1=int(input("Value of y1="))
y2=int(input("Value of y2="))
distance=math.sqrt(x2-x1)**2+(y2-y1)**2
print("The distance between two points is",distance)
```

```
Value of x1=10
Value of x2=20
Value of y1=30
Value of y2=40
The distance between two points is 110.0
```

Name: PRAJVIN
Register Number:

**Result:**

In this experiment, I have implemented the given problems by developing python programming using simple statements and the output was verified successfully.