Day objectives

- Basics of python
    - comments
    - keywords
    - variables
    - operators
- Conditional statements
    - if
    - if..else
    - if...elif..else
    - nested..if
- Control statements or Iterative statements
    - for
    - while
    - nested for and nested while

- Comments
    - Used for holding particular lines of code
    - We have two types of comments
        - single line comment
            - #
        - multi line comment
            - """ data """
- Keywords
    - We can call it as reserved words
    - To defined code
    - keywords cannot used as a variable name,function name
- Variables
    - Which is used for store values a specific memory location
    - Rules for declaring variables
        - keywords cannot used as a variable name
        - Variable should not starts with integers
        - Don't mention spaces
        - Variable should not starts with special characters
        - but it starts only letter (_)
        - Don't used function names as a varaible name
- Data Types
    - Data type specifies which type of value of a variable
    - int (intergers)
    - str (for character or names)
    - float (for decimal values)

In [22]:

```
# print("APSSDC")
print('msg')
""" print('Good morning')
print('Name')
print('Hello world') """
print('hi')
```

```
msg
hi
```

In [19]:

```
print('name')
"""print('hello')
print("man")"""
print('mam')
```

```
name
mam
```

In [ ]:

```
import keyword
print(keyword.kwlist)
```

In [4]:

```
print(len(keyword.kwlist))
```

```
35
```

In [6]:

```
n1 = 12
n2 = 15
print(n1)
print(n2)
type(n1)
```

```
12
15
```

Out[6]:

```
int
```

In [7]:

```
t = 3.56
type(t)
```

Out[7]:

```
float
```

In [8]:

```
y = 'hello'
type(y)
```

Out[8]:

```
str
```

In [9]:

```
1n = 5
2n = 67
print(1n)
print(2n)
```

```
  File "<ipython-input-9-cd06edd6d22b>", line 1
    1n = 5
     ^
SyntaxError: invalid syntax
```

In [10]:

```
_h = 6
print(_h)
```

```
6
```

In [23]:

```
n1 = n2 = n3 = 7  ## multi variable assignment with same value
print(n1)
print(n2)
print(n3)
```

```
7
7
7
```

In [24]:

```
a,b,c = 123,234,456 ## multi varaibale assignment with multi values
print(a)
print(b)
print(c)
```

```
123
234
456
```

- Type casting
  - Changing one kind of data type into another kind of data type

In [30]:

```
a = 12
type(a)
res = float(a)
type(res)
```

Out[30]:

```
float
```

In [36]:

```
x = int(input('enter first number: '))
y = int(input('enter second number: '))
print(x)
print(y)
type(x)
```

```
enter first number: 3
enter second number: 4
3
4
```

Out[36]:

```
int
```

In [41]:

```
## Dynamic way of programming

x = int(input('enter first number: '))
y = int(input('enter second number: '))
print(x+y)
```

```
enter first number: 6
enter second number: 3
9
```

In [40]:

```
## Static way of programming

x = 8
y = 2
print(x+y)
```

```
10
```

In [33]:

```
t = input('enter input')
type(t)
```

```
enter input56
```

Out[33]:

```
str
```

Operators

- Used to perform mathematical operations and for adding values
- Arithmetic operators(+,-,,/,%,//(floor division),*(power))
- Assignment operators(=, +=, -=, *= , /=)
- Comparison operators(==, != , <,>,<=,>=)
- Logical operators(and, or, not)
- Identity operators(is, is not)
- Membership operators(in, not in)
- Bistwise operators(&,|,^,<<,>>)

In [42]:

```
a = int(input('enter a number: '))
b = int(input('enter another number: '))
print(a+b)
print(a-b)
print(a*b)
print(a/b)
print(a//b)
print(a%b)
print(a**b) ## 11**5
```

```
enter a number: 11
enter another number: 5
16
6
55
2.2
2
1
161051
```

In [45]:

```
s = 10
b = 20

s += b  ## s = s+b  s = 10+20
print(s)

s -= b  ## s = s-b  s = 30 - 20
print(s)

s *= b
print(s)
```

```
30
10
200
```

In [52]:
```
## comparison operators

c = 10
d = 10
print(c == d) ## 10 == 20
print(c < d ) ## 10<20
print(c > d) ## 10>20
print(c <= d)
print(c >= d)
print(c != d)
```

```
True
False
False
True
True
False
```

In [56]:
```
## Logical operators

a = 6
print(a>4 and a>5)  ## 6>4 and 6<5
print(a>4 or a<5) ## 6>4 or 6<5
print(not(a>4 or a<5))
```

```
True
True
False
```

In [58]:
```
## Identity operators

a = 'mango'
b = 'Mango'
print(a is b)
print(a is not b)
```

```
False
True
```

In [60]:
```
## Membership operators

a = 'Nandini'
print('n' in a)
print('d' not in a)
```

```
True
False
```

In [61]:

```
## Bitwise operators

a = 10  ## 1010
b = 2 ##   0010
print(a&b)
print(a|b)
print(a^b)
```

```
2
10
8
```

In [62]:

```
a = 10
a<<1
```

Out[62]:

```
20
```

In [ ]:

In [ ]:

Conditional statements

- To execute specific block or code when the given condition is true or false
- Types of conditional statements
  - if --> to check single condtions in a program
  - if..else --> to check two condtions in a program
  - if..elif..else --> to check more that two condtions
  - nested --> conditions inside conditions

if

Syntax:

if(condition): statement/logic

indendation: - four spaces or one tabspace

In [65]:

```
## sum of two number is greater than 20 or not
## input: 12 14 -->26
## output: 26 is greater than 20

x = int(input('enter first number: '))
y = int(input('enter second number: '))
if(x+y > 20):
    print(x+y,'is greater than 20')
```

```
enter first number: 9
enter second number: 3
```

if...else

Syntax:

if(condition): statements/logic else: staements/logic

In [68]:

```
## EVEN OR ODD

num = int(input('enter a number: '))
if(num%2 == 0):
    print(num,'is Even')
else:
    print(num,'is Odd')
```

```
enter a number: 5
5 is Odd
```

In [67]:

```
4%2 == 0
```

Out[67]:

```
True
```

In [69]:

```
5%2 == 0
```

Out[69]:

```
False
```

In [71]:

```
## Check the given number is existed in the given range or not
## input: 6
## range --> 1--10
## Existed
## Not existed

n = int(input('enter number: '))
if(n>=1 and n<=10): ## 6>=1 and 6<=10
    print('Existed')
else:
    print('Not existed')
```

```
enter number: 78
Not existed
```

if..elif..else

Syntax:

if(condition): statements/logic elif(condition): statements/logic

--

-- else: statements/logic

In [73]:

```
## Mathematical operations
## +,-,*,/

a = int(input())
b = int(input())
c = input()
if(c == '+'):
    print(a+b)
elif(c == '-'):
    print(a-b)
elif(c == '*'):
    print(a*b)
elif(c == '/'):
    print(a/b)
else:
    print('Invalid character')
```

```
9
5
@
Invalid character
```

nested if..elif..else

Syntax:

if(condition): statements if(condition): statements/logic

--

In [78]:

```
## 1 -- 100 (user_ids)
    ## password (12345)
        ## Successfully entered
    ## Wrong password
## Invalid use_id

u_id = int(input('enter user_id: '))
if(u_id>=1 and u_id<=100):
    pwd = int(input('enter password: '))
    if(pwd == 12345):
        print('Successfully entered')
    else:
        print('Wrong passowrd')
else:
    print('Invalid user')
```

```
enter user_id: 6
enter password: 4
Wrong passowrd
```

Tasks

1. Number spelling(0 -- 9)
    - input: 5
    - output: Five
2. Days
    - input: 2
    - output: Monday
3. Big number of 3 number
4. Vote eligibility
    - input: 23
    - output: Eligible
5. Check the given character is vowel or consonant

Type *Markdown* and LaTeX: α^2

In [ ]:

```

```

Iterations/Loops

- for
    - which is used to execute a block of code into multiple times
- while
- nested for
- nested while

In [79]:

```
print('apssdc')
print('apssdc')
print('apssdc')
print('apssdc')
print('apssdc')
```

```
apssdc
apssdc
apssdc
apssdc
apssdc
```

Syntax of for loop:

for variable in range(start,end,step): statements/logic

In [80]:

```
for i in range(1,10,1):
    print(i)
```

```
1
2
3
4
5
6
7
8
9
```

In [81]:

```
for i in range(1,10):
    print(i)
```

```
1
2
3
4
5
6
7
8
9
```

In [82]:

```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

In [83]:

```
for i in range(1,10):
    print(i,end=" ")
```

```
1 2 3 4 5 6 7 8 9
```

In [84]:

```
for i in range(1,10):
    print(i,end="&")
```

```
1&2&3&4&5&6&7&8&9&
```

In [85]:

```
for i in range(10,0,-1):
    print(i,end=' ')
```

```
10 9 8 7 6 5 4 3 2 1
```

In [86]:

```python
for i in range(1,501):
    print('apssdc',end=' ')
```

```
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc a
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc
apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc ap
ssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apssdc apss
dc apssdc
```

In [88]:

```
## sum of n natural numbers

sr = int(input('enter start value: ')) ## 1
er = int(input('enter end value: ')) ## 10
s = 0
for i in range(sr,er+1): ## (1,11)
    s = s+i ## s += i

print(s)
```

```
enter start value: 78
enter end value: 123
4623
```

In [96]:

```
## print even series and odd series

sr = int(input('enter start value: ')) ## 1
er = int(input('enter end value: ')) ## 10
e_sum = o_sum = e_count = o_count = 0

print('Even numbers are: ')
for i in range(sr,er+1):
    if(i%2 == 0):
        e_sum += i
        e_count += 1
        print(i,end=' ')
print('\nOdd numbers are: ')
for i in range(sr,er+1):
    if(i%2 != 0): ## if(i%2 == 1)
        o_sum += i
        o_count += 1
        print(i,end=' ')
print('\nEven numbers sum is: ',e_sum)
print('Odd number sum is: ',o_sum)
print('Even numbers count is: ',e_count)
print('Odd numbers count is: ',o_count)
```

```
enter start value: 1
enter end value: 20
Even numbers are: 
2 4 6 8 10 12 14 16 18 20 
Odd numbers are: 
1 3 5 7 9 11 13 15 17 19 
Even numbers sum is:  110
Odd number sum is:  100
Even numbers count is:  10
Odd numbers count is:  10
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: