Day Objectives

Introduction To Juypter Notebook

BOLD

ITALIAN

*TEXT

1.ordered list 1

2.ordered list 2

3.ordered list 3

Python Basics

Arthematic operations

In [1]:

```
##1

'''divija
divija1
divijag2''' #Using For Multi-line Comments
#->for single line comment
```

Out[1]:

```
'divija\ndivija1\ndivijag2'
```

In [2]:

```
##2

##variable assignment
n=369
n
```

Out[2]:

```
369
```

In [3]:

```
##3

n1=n
n
```

Out[3]:

```
369
```

In [8]:

```
##4

n2=n1
n1
```

Out[8]:

```
369
```

In [10]:

```
##5

##multi variables holding single value
n1=n2=n3=n
n2
```

Out[10]:

```
369
```

In [15]:
```
##6

n=12
n
n1=10
n1##here we will get n1 as output (whatever we give atlast only that will be prin
```

Out[15]:
```
10
```

In [14]:
```
##7

n##can use same variable in different cells
```

Out[14]:
```
12
```

In [16]:
```
##8

n=12
print(n)
n1=10
n1##if we need to print all outputs then use 'print'(for last value we need not g
```

```
12
```

Out[16]:
```
10
```

In [17]:
```
##9

n=12
n1=10
n1
print(n)
```

```
12
```

In [18]:
```
##10

n=1
n=2
n=3
print(n)
n1=2
print(n1)
```

```
3
2
```

In [34]:

```
##11

#type conversion: converting one datatype into another
x=10
print(x)#default type int
type(x)
y=str(x)
print(type(y))
print(float(x))
#type(x)
```

```
10
<class 'str'>
10.0
```

In [35]:

```
##12

#type conversion: converting one datatype into another
x=12
type(x)
float(x)
type(x)
```

Out[35]:

```
int
```

In [36]:

```
##13

#type conversion: converting one datatype into another
x=12
type(x)
float(x)
#type(x)
```

Out[36]:

```
12.0
```

In [37]:
```
#basic operations
##add
x=1
y=2
print
print(x+y) ##directly also we can get output by giving values, but the correct way
##sub
k=-3
j=-4
print(k-j)
##mult
c=((x+y)*(k+j))
##division
print(c/2)#single modular division operator gives float value
print(c//2)#double single modular division operator gives int value
##modular
print(c%2)
c
```

```
3
1
-10.5
-11
1
```

Out[37]:
```
-21
```

In [38]:
```
print(2+3)
print(1723687687*2869878668)#directly we can get output by giving values
```

```
5
4946774523215560916
```

In [39]:
```
a='k'
type(a)##string value directly cannot be coneverted into int,we must give assign
```

Out[39]:
```
str
```

In [40]:
```
a='k'
int(str(a))
type(a)##here it didnot convert since we gave string value(str cannot be converte
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-40-2ece08e9b1d0> in <module>()
      1 a='k'
----> 2 int(str(a))
      3 type(a)

ValueError: invalid literal for int() with base 10: 'k'
```

In [41]:

```
u="1"
print(u)
y=int(u)
print(y)#converting string to int datatype
```

```
1
1
```

Indentation In Python

Indentation is nothing but space or tab

Conditional statements

if,if else,if elif else,nested if

In [43]:

```
## If Condition
## Write a program to know how to use if condition
n=3
if(n%2==0):
    n=n+1
    print("n is even no.")
else:
    print("n is not even no.")
```

```
n is not even no.
```

In [ ]: