##Markdown Basics

- **Bold**
- **Italic**
- ***IB***
- Normal Text
  - sublist 1
  - sublist 2

1. Order list of elements 1
2. Order list of element 2

- ☐ option 1
- ☑ option 2

I get 10 times more traffic from [Google] 1 (http://google.com/) than from [Yahoo] 2 (http://search.yahoo.com/) or [MSN] 3 (http://search.msn.com/)

printf("hello Markdown")

Jupyter Logo (logo.png)

In [ ]:
```
1
```

Python Basics

Python Version 3.7

Python comments

print("Hello Mr.Beam") #basic output print("Hello Python","!",end=" ") print("Good Afternoon")

In [ ]:
```
1
```

Assignment

In [5]:
```
1 n1=123456           #single variable assignment
2 n2=n3=n4=n1         #multivariable assignment
3
4 a,b,c=123,234,345   #multivariable assignment with differnt values
5
6                     #if you give a,b,c output only c value will come.
7
8 #print(a,b,c)
9 n1
```

Out[5]: 123456

Data Types &Conversions

- int
- float
- string

In [14]:
```
type(a)
s1="python"
type(s1)
f1=12.345
type(f1)
int(f1)
float(str(int(f1)))


```

Out[14]:
```
12.0
```

Airthmetic Operations

- +
- -
- *
- /
- %
- **

In [18]:
```
n1%11

n3=n2**123

type(n3)
len(str(n3))


atoms=10**82
len(str(atoms))
type(str(atoms))

122321**9
```

Out[18]:
```
6130687873308026945890176790042303730066739281
```

In [ ]:
```

```

Conditionals:

In [21]:
```
if atoms < 10 ** 76:
    print("TRUE")
else:
    print("false")

```

```
false
```

In [2]:
```
n = 123
if n % 2 == 0:
    print("even")
else:
    print("odd")

```

```
odd
```

Type *Markdown* and LaTeX: α^2

In [1]:
```
### find the greast of three numbers:
n1 = int(input("enter the first number"))
n2 = int(input("enter the second number"))
n3 = int(input("enter the third number"))
if n1>n2 and n1 > n3:
    print (n1,"is the greatest number")
elif n2>n3:
    print (n2,"is the greast number")
else:
    print(n3, "is the greatest number")



```

```
enter the first number12
enter the second number23
enter the third number45
45 is the greatest number
```

Type *Markdown* and LaTeX: α^2

In [4]:
```
## Leap year

n=int(input("enter the any of the year"))
if n % 4 == 0 and n % 100 !=0:
    print(n,"leap year")
elif n % 400 == 0:
    print (n,"leap year")
else:
    print("not leap year")



```

```
enter the any of the year2016
2016 leap year
```

In [ ]:
```

```

In [5]:
```
## Check if a given number exits in a given range

n1 = int(input("enter the number"))
lb = 20
ub = 400
if n1 >= lb and n1 <=400:
    print (n1,"the given number is exits")


```

```
enter the number34
34 the given number is exits
```

In [24]:
```
## Calculate if a number of digits in a number

n1 = int(input("enter the number"))
a=len(str(n1))
print(a)

```

```
enter the number12345
5
```

In [6]:
```
## Calculate if a number is a multiple of 10

n1 =int(input("enter number"))
if n1 % 10==0:
    print(n1, "is the multiple of 10")

```

```
enter number20
20 is the multiple of 10
```

In [7]:
```
## Check if a number is a factor of 1000

n1 = int(input("enter the number"))
if 1000%n1 == 0:
    print (n1,"is a factor of 1000")
```

```
enter the number20
20 is a factor of 1000
```

In [12]:
```
## Check if the given string is equal to a number

n1 = int(input("enter number"))
n2 = str(input("enter the string"))
sol=int(str(n2))
if n1 ==sol:
    print("the given string and integer is same")
else:
    print("not same")


```

```
enter number123
enter the string123
the given string and integer is same
```

In [16]:
```
## Calculate the Quadratic equation of a number without function

a = int(input("enter a value"))
b = int(input("enter b value"))
c = int (input("enter c value"))
d = float((b**2)-(4*a*c))
sol1 = (-b+(d)**0.5)/2*a
sol2 = (-b-(d)**0.5)/2*a
print ("the solution of equation is",sol1)
print ("the solution of equation is",sol2)
```

```
enter a value2
enter b value3
enter c value4
the solution of equation is (-2.9999999999999996+4.795831523312719j)
the solution of equation is (-3.0000000000000004-4.795831523312719j)
```

In [18]:
```
## Calculate the square root the given number

n1=int(input("enter number"))
n2=n1**0.5
n2

```

```
enter number25
```

Out[18]:
```
5.0
```

In [19]:
```
## Calculate the number of nano seconds in a given year
## (Considering leap year logic)

year = 2016
if year%400==0 or (year%4==0 and year%100!=0):
    print(366*24*60*60*10**9)
else:
    print(365*24*60*60*10**9)
```

```
31622400000000000
```

In [ ]:
```

```