

```
In [27]: a ="Hello, This is python class"
```

```
In [2]: print(a.upper())
```

```
HELLO, THIS IS PYTHON CLASS
```

```
In [3]: print(a.lower())
```

```
hello, this is python class
```

```
In [4]: print(a.title())
```

```
Hello, This Is Python Class
```

```
In [5]: print(a.capitalize())
```

```
Hello, this is python class
```

```
In [6]: print(a.swapcase())
```

```
hELLO, tHIS IS PYTHON CLASS
```

```
In [7]: print(a.replace('ll','mno'))
```

```
Hemnoo, This is python class
```

## splitting with split function without delimiter

```
In [8]: print(a.split("i"))
```

```
['Hello, Th', 's ', 's python class']
```

```
In [9]: print(a.split("s"))
```

```
['Hello, Thi', ' i', ' python cla', '', '']
```

## splitting with delimiter using "RE" Package

```
In [23]: import re
```

```
In [12]: print(re.split(r'(',')',a))
```

```
['Hello', ',', ' This is python class']
```

```
In [28]: print(re.split(r'(t)(h)',a))
```

```
['Hello, This is py', 't', 'h', 'on class']
```

## Built-in Functions

```
In [13]: print("This is a code")
```

This is a code

### round off function with single argument

```
In [14]: print(round(8.2762))
```

8

### round off function with two argument

```
In [15]: print(round(8.9456,3))
```

8.946

```
In [16]: a = 8.97654  
print(round(a,2))
```

8.98

```
In [17]: help(round)
```

Help on built-in function round in module builtins:

```
round(number, ndigits=None)  
    Round a number to a given precision in decimal digits.
```

The return value is an integer if ndigits is omitted or None. 0 otherwise

the return value has the same type as the number. ndigits may be negative.

### divmod will do division and output will be(quotient,remainder)

```
In [18]: x=divmod(30,7)
```

```
In [19]: print(x)
```

(4, 2)

```
In [20]: type(x)
```

Out[20]: tuple

```
In [21]: print(x[1])
```

2

## Output will be in bool

```
In [22]: a = 5.7  
print(isinstance(a,int))
```

False

```
In [23]: print(isinstance('abc',(str,float,int)))
```

True

```
In [24]: a= 5
```

```
In [25]: type(a)
```

```
Out[25]: int
```

## returns the power

```
In [26]: print(pow(2,8))
```

256

## $\text{pow}(x,y,z) = ((x**y) \% z)$

```
In [27]: print(pow(2,4,7))
```

2

```
In [29]: print(pow(5,2,7))
```

4

## taking user input

```
In [4]: name = input("Enter your name: ")
```

Enter your name: Joseph

```
In [5]: print(name)
```

Joseph

```
In [6]: a = input("enter a value: ")
```

enter a value: 32

```
In [7]: print(a)
```

32

```
In [8]: b = input("Enter your name:")
print("hello,"+ b +" for todays lecture ")
```

Enter your name:Joseph  
hello,Joseph for todays lecture

## User defined functions

```
In [9]: def printsomething():
        print("Good Job")
        print("We are studying functions")
```

```
In [10]: printsomething()
```

Good Job  
We are studying functions

## function with 1 argument

```
In [11]: def printmsg(msg):
        if isinstance(msg,str):
            print(msg)
        else:
            print("Not a string")
```

```
In [12]: printmsg("Vijal")
```

Vijal

```
In [17]: printmsg("Hithere@Joseph")
```

Hithere@Joseph

```
In [18]: printmsg(34)
```

Not a string

```
In [20]: def mypower(a,b):
        c=a**b
        print(c)
```

In [21]: mypower(3,4)

81

In [ ]: