7/6

```
def display(**n):
        print(type(n))

display(dharma=99, rajesh=34)

o/p:-<class 'dict'>


def display(**n):

        for k,v in n.items():
                print(k,"=",v)

display(dharma=99, rajesh=34)

o/p:-

  def add(a,b):
        [illegible]
        return a+b
  print(add(10,20))

  o/p:- 30

  add=lambda a,b: a+b
  print(add(10,20))

  o/p:- 30
```

Squaring a number :-

```
def sqr(a):
        return a*a
print (sqr(10))
```

o/p:- 100
(or using lambda)

```
sqr = lambda a: a*a
print (sqr(10))
```

o/p:- 100

Advanced functions :

i, filter   ii, map   iii, reduce

→ filter:

```
list1 = [1,2,3,4,5,6,7,8,9]
```

Let filter Even no's for above list

Ex:
```
def even(a):
        if a%2==0:
            return True
        else:
            return False
list1 = [1,2,3,4,5,6,7,8,9]
print(list(filter(even, list1)))
```

o/p : [2,4,6,8]

Using lambda function

```
list1= [1,2,3,4,5,6]
print(list(filter(lambda a:a%2==0 ,list1)))
o/pt [2,4,6]
```

~~listel~~
Only Single line we can also print

```
print(list(filter(lambda a: a%2==0 ,range(10))))
o/pt [2,4,6,8]
```

→ Map :-

Let double the Every Element in list

```
def double(x):
        x=x*2
list1=[1,2,3,4]
print(list(map(double,list1)))
o/pt [2,4,6,8]
```

(or)

Using lambda

```
list1=[1,2,3,4]
Print(list(map(lambda a:a*2, list1)))
```

→ Reduce:-

Ex:
```
def red(a,b):
        return a+b

list1=[1,2,3,4]
print(reduce(red, list1))
```
o/p: Error

Ex:
```
from functools import reduce

def red(a,b):
        return a+b

list1=[1,2,3,4,5,6]
print(reduce(red, list1))
```
o/p:- 21

(or)

Using lambda ~~function~~.

```
from functools import reduce
print(reduce(lambda a,b:a+b, range(9)))
```
o/p:- 36