Packing and Unpacking

Wednesday, July 9, 2025 6:58 PM

1. Packing: It is a phenomenun of grouping the values together to provide security

It can be done on all the collection datatypes but most preferly we use tuple because it is the most secured datatype.

Types of Packing:

i. Single/Tuple Packing
ii. Double/Dictionary Packing

i. Single/Tuple Packing: It is the phenomenun of packing the individual data in form of tuple.

Syntax:

```
def fname(*args):
      S B
fname(val1,val2,val3…….val n)
```

'*' is used to store all the values in form of tuple.

Eg:

```
def sam(*args):
   print(type(args))
   print(args)
sam(2,3,4,5,6,7,8,4,5,11,23,54,'hii')
```

ii. Double/Dictionary Packing:

It is a phenomenun of packing the pair of data in the form of dictionary

Syntax: def fname(**kwargs):

```
      S B
fname(k1=v1,k2=v2……..kn=vn)
```

For naming the keys, we have to follow the identifier rules.

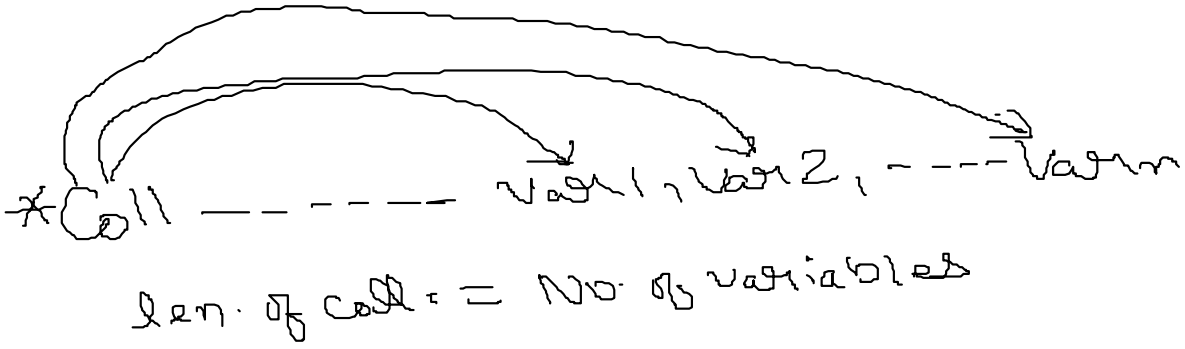
For easy convenience we should use : (*args,**kwargs)
Here first we need to pass positional arguments after that only we can pass keyword arguments.

Eg:

```
def sam(*args,**kwargs):
   print(type(kwargs))
   print(kwargs)
   print(type(args))
   print(args)
sam(5,a=1,b=2,c=3,d=4)
```

Unpacking:

It is the phenomenun of dividing the collection into n number of values.



Syntax:

```
def fname(var1, var2, var3……. Var n):
      S B
fname(*collection)
```

Here * will unpack all the types of data present

Eg:

```
def sam(v1,v2,v3,v4):
   print(v1,v2,v3,v4)
sam(*'abcd')
```