1. What is the result of the code, and why?

```
def func(a, b=6, c=8):
print(a, b, c)
func(1, 2)
```

```
def func(a,b=6,c=8):
  print(a,b,c)
func(1,2)
```

```
1 2 8
```

Answer: The result of the code is 1,2,8 because it is func() used the default value c-8 ,which is provided at time of decleration.

2. What is the result of this code, and why?

```
def func(a, b, c=5):
print(a, b, c)
func(1, c=3, b=2)
```

```
def func(a,b,c=5):
  print(a,b,c)
func(1,c=3,b=2)
```

```
1 2 3
```

Answer: The result of this code is 1 2 3.because the function used default values only when a value for a argument is not provided and if the argument name is mentioned while doing a function call, the order of arguments is also ignored by the python interpreter

3. How about this code: what is its result, and why?

```
def func(a, *pargs):
print(a, pargs)
func(1, 2, 3)
```

```
def func(a,*pargs):
  print(a,pargs)
func(1,2,3)
```

```
1 (2, 3)
```

Answer: The result of this code is 1,(2,3). because it is function used *pargs stands for variable length arguments. This format is used when we are not sure about the number of arguments to be passed to a function. All the values under this argument will be stored in a tuple.

4. What does this code print, and why?

```
def func(a, **kargs):
print(a, kargs)
func(a=1, c=3, b=2)
```

```
def func(a,**kargs):
  print(a,kargs)
func(a=1,c=3,b=2)
```

```
1 {'c': 3, 'b': 2}
```

Answer: The result of this code is 1{'c':3,'b':2}. ** karges stande for variable keywords arguments.all key value are stored in dicctionary

5. What gets printed by this, and explain?

```
def func(a, b, c=8, d=5): print(a, b, c, d)
func(1, *(5, 6))
```

```
def func(a,b,c=8,d=5):
  print(a,b,c,d)
func(1,*(5,6))
```

```
1 5 6 5
```

Answer: The result of this code is 1 5 6 5. function not throwing an error because, this function expects 4 arguments. The value for a is provided explicitly whereas for arguments b and c, the function will expand *(5,6) and consider the value of b as 5 and value of c as 6. Since the default value of d is provided in function declaration, d value will be 5.

6. what is the result of this, and explain?

```
def func(a, b, c): a = 2; b[0] = 'x'; c['a'] = 'y'
l=1; m=[1]; n={'a':0}
func(l, m, n)
```

```
def func(a,b,c):a=2;b[0]='x';c['a']='y'
l=1;m=[1];n={'a':0}
func(l,m,n)
l,m,n
```

```
(1, ['x'], {'a': 'y'})
```

Answer: The result of this code is (1,['x'],{'a':'y'}. because l,m,n are provided as inputs to the function, it modifies the values of l,m,n and sets the value of l=2 ,m=['x'] and n={'a':'y'}

Colab paid products - Cancel contracts here