assignment2

March 8, 2023

0.0.1 def keyword is used to create a function.

```python
[10]: def test():
          n=[]
          for i in range(1,25):
              if i%2==1:
                  n.append(i)
          return n
```

```python
[11]: test()
```

```
[11]: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23]
```

0.0.2 *args and **kwargs are used when we are unsure about number of arguments to pass in the functions.

0.0.3 *args (Non-keyword argument) **kwargs(keyword argument)

0.0.4 *args example

```python
[12]: def add(*num):
          sum=0
          for i in num:
              sum+=i
          print(sum)
      add(1,2,3,4)
```

```
10
```

0.0.5 **kwargs example

```python
[13]: def fruits(**kwargs):
          print(kwargs)
      fruits(apple=3,banana=4,papaya=5)
```

```
{'apple': 3, 'banana': 4, 'papaya': 5}
```

0.0.6 Iterator in python is an object that is used to iterate over iterable objects like lists,tuples,dicts and sets.

0.0.7 _ _ iter _ _() is used for initializing.

```python
[65]: from itertools import cycle
      def test2():
          l=[2, 4, 6, 8, 10, 12, 14, 16,18, 20]
          nums=cycle(l)
          for i in range(5):
              print(next(nums))
```

```python
[66]: test2()
```

```
2
4
6
8
10
```

0.0.8 A generator is a special type of function which does not return a single value,it returns an iterator object with a sequence of values.

0.0.9 yield is used to pause and resume a generator function.

```python
[68]: def simple():
          for i in range(10):
              if(i%2==0):
                  yield i
      for i in simple():
          print(i)
```

```
0
2
4
6
8
```

```python
[13]: import math
      def Prime(n):
          for i in range(2,n//2+1):
              if(n%i==0):
                  return 0
          return 1
      n=20
      i=2
      list=[]
      while(1):
          if(Prime(i)):
```

```
        list.append(i)
        if(len(list)==n):
            break
    i+=1
print(list)
```

[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71]

[ ]: