assignment=3

July 22, 2023

1 Q1. Which keyword is used to create a function ? create a function to return a list of odd numbers in the range of 1 to 25.

Ans.

1. The def keyword is used to create a function
2. Odd numbers in the range of 1 to 25

EX,

[2]:
```
L = [1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25]
```

[6]:
```
L
```

[6]:
```
[1,
 2,
 3,
 4,
 5,
 6,
 7,
 8,
 9,
 10,
 11,
 12,
 13,
 14,
 15,
 16,
 17,
 18,
 19,
 20,
 21,
 22,
 23,
 24,
```

25]

[7]:
```
list(filter(lambda x : x % 2 != 0,L))
```

[7]:
```
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25]
```

2 Q2. Why * args and ** kwargs is used in some functions ? create a function each for * args and ** kwargs to demonstrate their use .

Ans. 1. You can use * args and ** kwargs as argument of a function When you are unsure about The number of argument to pass in the function.

2.

(i) * args Example:

[8]:
```
def test1(*args):
    return args
```

[9]:
```
test1()
```

[9]:
```
()
```

[10]:
```
type(test1())
```

[10]:
```
tuple
```

[11]:
```
test1(1,2,3,)
```

[11]:
```
(1, 2, 3)
```

[13]:
```
test1(1,2,3,"sudh" ,"kumar",[1,2,3,4,5,6,7,8,9,])
```

[13]:
```
(1, 2, 3, 'sudh', 'kumar', [1, 2, 3, 4, 5, 6, 7, 8, 9])
```

(ii) ** kwargs Example:

[15]:
```
def test2(**kwargs):
    return kwargs
```

[16]:
```
test2()
```

[16]:
```
{}
```

[17]:
```
type(test2())
```

[17]:
```
dict
```

[18]:
```
test2(a=[1,2,3,4,] , b="sonal" , c=98.89 )
```

[18]:
```
{'a': [1, 2, 3, 4], 'b': 'sonal', 'c': 98.89}
```

3 Q3. What is an iterator in python ? Name the method used to initialise the iterator object and the method used for iteration . Use these methods to print the first five elements of the given list [2,4,6,8,10,12,14,16,18,20].

Ans: 1. IN python , an iterator is an object that allows you to iterate over collections of data,such as lists,tuples ,dictionaries , and sets. python iterators implement the iterator design pattern , Which allows you to traverse a container and access its elements.

```
#An iterator in Python is an object that contains a countable number of elements that can be
```

2. Iter() method is used to initialize the iterator object so that the instance of this object can be used for iterating.

3. Example :

[35]:
```
s ="sonal"
```

[36]:
```
for i in s :
    print (i)
```

```
s
o
n
a
l
```

[37]:
```
s
```

[37]:
```
'sonal'
```

[38]:
```
next(s1)
```

```
---------------------------------------------------------------------------
StopIteration                             Traceback (most recent call last)
Cell In [38], line 1
----> 1 next(s1)
```

```
StopIteration:
```

[39]:
```
s1 =  iter(s)
```

[40]:
```
next(s1)
```

[40]: 's'

[41]:
```
next(s1)
```

[41]: 'o'

[42]:
```
next(s1)
```

[42]: 'n'

[43]:
```
next(s1)
```

[43]: 'a'

[44]:
```
next(s1)
```

[44]: 'l'

[46]:
```
my_list = [2,4,6,8,10,12,14,16,18,20]
```

[47]:
```
for i in my_list:
    print(i)
```

```
2
4
6
8
10
12
14
16
18
20
```

[48]:
```
my1 = iter(my_list)
```

[49]:
```
next(my1)
```

[49]: 2

[50]:
```
next(my1)
```

[50]: 4

[51]:
```
next(my1)
```

[51]: 6

[52]:
```
next(my1)
```

[52]: 8

[53]:
```
next(my1)
```

[53]: 10

[54]:
```
next(my1)
```

[54]: 12

[55]:
```
next(my1)
```

[55]: 14

[56]:
```
next(my1)
```

[56]: 16

[57]:
```
next(my1)
```

[57]: 18

[58]:
```
next(my1)
```

[58]: 20

4 Q4. What is a generator function in python ? Why yield keyword is used? Give an example of a generator function.

Ans:

1. In Python, a generator is a function that returns an iterator that produces a sequence of

2. Yield keyword is used to create a generator function.

3. Example :

[59]:
```
range(10)
```

[59]:
```
range(0, 10)
```

[61]:
```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

[62]:
```
def test__fib(n):
    a,b = 0,1
    for i in range (n):
        yield a
        a,b = b , a+b
```

[63]:
```
for i in test__fib(10):
    print(i)
```

```
0
1
1
2
3
5
8
13
21
34
```

5 Q5. Create a generator function for prime numbers less than 1000 . use the next() method to print the first 20 prime numbers.

[ ]:
```
Ans:

    1.    Example:
```

[3]:
```
for x in range (1,1000):
    for y in range(2,x):
        if  x%y==0:break
    else:
        print(x,sep='', end='')
```

```
123571113171923293137414347535961677173798389971011031071091131271311371391491511571631671731791811911931971992112232272292332392412512572632692712772812832933073113133173313373473493533593673733793833893974014094194214314334394434494574614634674794874914995035095215235415475575635695715775875935996016076136176196316416436476536596616736776836917017097197277337397437517577617697737877978098118218238278298398538578598638778818838879079119199299379419479539679719779839919971235711131719232931374143475359616771737983899710110310710911312713113713913915115716316717317918119119319719921122322722923323924125125726326927127728128329330731131331733133734734935335936737337938338939740140941942143143343944344945746146346747948749149950350952152354154755756356957157758759359960160761361761963164164364765365966167367768369170170971972773373974375175776176977378779780981182182382782983985385785986387788188388790791191992993794194795396797197798399919971235711131719232931374143475359616771737983899710110310710911312713113713913915115716316717317918119119319719921122322722923323924125125726326927127728128329330731131331733133734734935335936737337938338939740140941942143143343944344945746146346747948749149950350952152354154755756356957157758759359960160761361761963164164364765365966167367768369170170971972773373974375175776176977378779780981182182382782983985385785986387788188388790791191992993794194795396797197798399919971
```

[7]:
```
for i in range(1,1000):
    for e in range(2,i):
        if i%e == 0:break
    else:
        print (i,sep ='',end='')
```

```
123571113171923293137414347535961677173798389971011031071091131271311371391491511571631671731791811911931971992112232272292332392412512572632692712772812832933073113133173313373473493533593673733793833893974014094194214314334394434494574614634674794874914995035095215235415475575635695715775875935996016076136176196316416436476536596616736776836917017097197277337397437517577617697737877978098118218238278298398538578598638778818838879079119199299379419479539679719779839919971235711131719232931374143475359616771737983899710110310710911312713113713913915115716316717317918119119319719921122322722923323924125125726326927127728128329330731131331733133734734935335936737337938338939740140941942143143343944344945746146346747948749149950350952152354154755756356957157758759359960160761361761963164164364765365966167367768369170170971972773373974375175776176977378779780981182182382782983985385785986387788188388790791191992993794194795396797197798399919971
```

2. Exampkle

[50]:
```
for s in range (1,20):
    for j in range(2,s):

        if s%j == 0:break
    else:
        print(s,sep ='', end ='')
```

```
1235711131719
```

[51]:
```
for s in range (20):
    print(s)
```

```
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
```

6 Q6.Write a python program to print the first 10 fibonacci number using a while loop.

[52]:
```
range(10)
```

[52]:
```
range(0, 10)
```

[53]:
```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

[54]:
```
def test__fib(n):
    a,b = 0,1
    for i in range(n):
        yield a
        a,b = b , a+b
```

[55]:
```
for i in test__fib(10):
    print(i)
```

```
0
1
1
2
3
5
8
13
21
34
```

[61]:
```
def test__fib():
    a,b = 0,1
    while True:
        yield a
    a,b = b , a+b
```

[62]:
```
fib = test__fib1()
```

7 Q7. Write a list comprehension to iterate through the given string :'pwskills'.

[64]:
```
s = "pwskills"
```

[65]:
```
list(s)
```

[65]:
```
['p', 'w', 's', 'k', 'i', 'l', 'l', 's']
```

8 Q8. Write a python program to check whether a given number is palindrome or not using a while loop.

Ans:

[66]:
```
class data_science:

    def syllabus(self):
```

```
        print("this is my assignment for data science masters")
```

[67]:
```
class web_dev:

    def syllabus(self):
        print("this is my  syllabus for data science")
```

[68]:
```
def class_parcer(class_obj):
    for i in class_obj:
        i.syllabus()
```

[70]:
```
data_science = data_science()
```

[71]:
```
web_dev = web_dev()
```

[72]:
```
class_obj = [data_science,web_dev]
```

[73]:
```
class_parcer(class_obj)
```

```
this is my assignment for data science masters
this is my  syllabus for data science
```

9 Q9. Write a code to print odd number from 1 to 100 using list comprehension.

Ans:

[82]:
```
#a code to print  odd number from 1 to 100 using list comprehension
[odd for odd in range(1,100) if odd%2!=0]
```

[82]:
```
[1,
 3,
 5,
 7,
 9,
 11,
 13,
 15,
 17,
 19,
 21,
 23,
 25,
 27,
 29,
 31,
 33,
```

```
 35,
 37,
 39,
 41,
 43,
 45,
 47,
 49,
 51,
 53,
 55,
 57,
 59,
 61,
 63,
 65,
 67,
 69,
 71,
 73,
 75,
 77,
 79,
 81,
 83,
 85,
 87,
 89,
 91,
 93,
 95,
 97,
 99]
```

[ ]: