```
# 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.

# Answer :-

# def is used to create a function

def is_odd():
  l = []
  for i in range(1,25):
    if i % 2 != 0:
      l.append(i)
  return l


is_odd()
```

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

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

#Answer :-

# *args >> It stands for arguments >> When we don't know that the number of argument of the function then we use *args
# **kwargs >> IT stands for Key words arguments >> When we take keys value pair i.e a dict in the arguments then we use **kwargs

# Example:-1

def sum(*args):
  add = 0
  for i in args:
    add += i
  return add


sum(1,2,3,4,5,6,47,48)
```

```
116
```

```
# Example:-2

def is_dict(**kwargs):
  return kwargs


is_dict(a = 2, b = 5,c =[1,2,3])
```

```
{'a': 2, 'b': 5, 'c': [1, 2, 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].


#Answer :-

# Iterator >> After converting to the iterable object using iter function the object is called iterator
# Frist we have to conver the iterable object to the iterator using iter function then use next to print the irerator element

lis = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
s = iter(lis)


next(s)
```

```
2
```

```
next(s)
```

```
4
```

```
next(s)
```

```
6
```

```
next(s)
```

```
8
```

```
next(s)
```

```
10
```

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

#Answer:-

# generator function >> Generator function is a special type of function which does not return a single object instead return an iterator obj
# Yield >> Yield calculate the value one by one to reduce the space of the memory.

# Example :- Generate a fibonachi series using generator function.

def fibo(n):
  a = 0
  b = 1
  for i in range(n):
    yield a
    a , b = b , a + b


f = fibo(100)


next(f)
```

```
0
```

```
next(f)
```

```
1
```

```
next(f)
```

```
1
```

```
next(f)
```

```
2
```

```
next(f)
```

```
3
```

```
next(f)
```

```
5
```

```
next(f)
```

```
8
```

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

def is_prime(n):
    if n <= 1:
        return False
    if n <= 3:
        return True
    if n % 2 == 0 or n % 3 == 0:
        return False
    i = 5
    while i * i <= n:
        if n % i == 0 or n % (i + 2) == 0:
            return False
        i += 6
    return True

def prime_generator(limit=1000):
    for num in range(2, 21):
        if is_prime(num):
            yield num

for prime in prime_generator():
    print(prime)
```

```
2
3
5
7
11
13
17
19
```

```
# Q6. Write a python program to print the first 10 Fibonacci numbers using a while loop.

# Answer :-

l = []
a = 0
b = 1
i = 0
while i<=10:
  a , b = b , a +b
  l.append(a)
  i += 1

print(l)
```

```
[1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
```

```
# Q7. Write a List Comprehension to iterate through the given string: ‘pwskills’.
#Expected output: ['p', 'w', 's', 'k', 'i', 'l', 'l', 's']

# Answer :-

string = 'pwskills'
[i for i in string]
```

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

```
# Q8. Write a python program to check whether a given number is Palindrome or not using a while loop.

# Answer :-

num = input("Enter a number : ")

while i
```

```
# Q9. Write a code to print odd numbers from 1 to 100 using list comprehension.

[i for i in range(1,101) if i % 2 != 0]
```

```
[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]
```

Start coding or generate with AI.