

QUES 1: 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.

```
In [13]: # def keyword is used to create a function .
```

```
def test():
    list_of_oddnumbers = []
    for i in range(25):
        if i%2!=0:
            list_of_oddnumbers.append(i)
    return list_of_oddnumbers
```

```
In [14]: test()
```

```
Out[14]: [1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23]
```

QUES2 : Why args and **kwargs is used in some functions? Create a function each for args and **kwargs to demonstrate their use.

```
In [31]: '''ANS: *args and **kwargs is used to take multiple input of arguments from user.The difference is *args is used to give output in the
form of tuple and **kwargs is used to take key value pair input and return it in the type of key value pair like dictionary.'''
```

```
#example of *args
```

```
def test2(*args):
    return args
```

```
In [32]: test2(2,3,4)
```

```
Out[32]: (2, 3, 4)
```

```
In [33]: # example of **kwargs
```

```
def test3(**kwargs):
    return kwargs
```

```
In [34]: test3(a = [1,2,3], b = "Arjun", c = 23.98)
```

```
Out[34]: {'a': [1, 2, 3], 'b': 'Arjun', 'c': 23.98}
```

QUES3: 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: an iterator is an object that allows traversal through a collection of elements, such as a list, tuple, or dictionary. It provides a way to access the elements one by one without the need for storing the entire collection in memory.

To initialize an iterator object in Python, you can use the iter() function. It takes an iterable as an argument and returns an iterator object.

To iterate over the elements of an iterator, you can use the next() function. It retrieves the next element from the iterator. Each time you call next()

```
In [66]: my_list = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
```

```
a = iter(my_list)
```

```
for i in range(5):
    print(next(a))
```

```
2
4
6
8
10
```

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

```
In [77]: ''' In Python, a generator function is a special type of function that returns an iterator called a generator.
It allows you to define an iterative algorithm by writing a function that generates the values on the fly,
instead of returning them all at once in a list or another data structure.
Generator functions are defined using the yield keyword instead of return.'''
```

```
''' The yield keyword is used within a generator function to indicate where the generator should pause its execution and
yield a value to the caller.'''
```

```
# example of generator function
```

```
def even_numbers(limit):
    num = 0
    while num < limit:
        yield num
        num += 2
```

```
In [79]: my_generator = even_numbers(10) # Create a generator object
```

```
for i in my_generator:
    print(i)
```

```
0
2
4
6
8
```

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

```
In [103... def prime_generator():
    primes = []
    num = 2
    while True:
        is_prime = True
        for prime in primes:
            if num % prime == 0:
                is_prime = False
                break
        if is_prime:
            primes.append(num)
            yield num
            num += 1

# Create the generator object
prime_gen = prime_generator()

# Print the first 20 prime numbers
for _ in range(20):
    print(next(prime_gen))
```

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

QUES 6: Write a python program to print the first 10 Fibonacci numbers using a while loop.

```
In [104... a, b = 0, 1
count = 0

while count < 10:
    print(a)
    a, b = b, a + b
    count += 1
```

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

QUES 7: Write a List Comprehension to iterate through the given string: 'pwskills'. Expected output: ['p', 'w', 's', 'k', 'i', 'l', 'l', 's']

```
In [105... string = 'pwskills'
output = [char for char in string if char in 'pwski']

print(output)
```

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

QUES 8: Write a python program to check whether a given number is Palindrome or not using a while loop.

```
In [106... def is_palindrome(number):
    original_number = number
    reverse_number = 0

    while number > 0:
        digit = number % 10
        reverse_number = reverse_number * 10 + digit
        number = number // 10

    return original_number == reverse_number

num = int(input("Enter a number: "))

if is_palindrome(num):
    print(f"{num} is a palindrome number.")
else:
    print(f"{num} is not a palindrome number.")
```

```
Enter a number: 54
54 is not a palindrome number.
```

QUES 9: Write a code to print odd numbers from 1 to 100 using list comprehension.

```
In [107... odd_numbers = [num for num in range(1, 101) if num % 2 != 0]
print(odd_numbers)
```

```
[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]
```

```
In [ ]:
```