#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.

```
def get_odd_numbers():
    odd_numbers = []
    for number in range(1, 26):
        if number % 2 != 0:
            odd_numbers.append(number)
    return odd_numbers

# Test the function
odd_numbers_list = get_odd_numbers()
print(odd_numbers_list)
```

#Why *args and **kwargs is used in some functions? Create a function each for *args and **kwargs to demonstrate their use
#ans - *args and **kwargs are used in Python functions when you want to accept a variable number of arguments or keyword arguments, respectively.

#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]

```
# Initialize the list
my_list = [2, 4, 6, 8, 10, 12, 14, 16, 18, 20]

# Initialize the iterator object
my_iterator = iter(my_list)

# Iterate and print the first five elements
for _ in range(5):
    print(next(my_iterator))
```

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

```
def fibonacci_sequence(n):
    a, b = 0, 1
    count = 0
    while count < n:
```

```
        yield a
        a, b = b, a + b
        count += 1

# Using the generator function to generate Fibonacci numbers
for num in fibonacci_sequence(10):
    print(num)
```

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

```
def primes():
    # Start generating primes from 2
    num = 2
    while num < 1000:
        if all(num % i != 0 for i in range(2, int(num**0.5) + 1)):
            yield num
        num += 1

# Using the generator function to print the first 20 prime numbers
prime_generator = primes()
for _ in range(20):
    print(next(prime_generator))
```

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

```
# Function to print the first 10 Fibonacci numbers
def fibonacci():
    count = 0
    a, b = 0, 1
    while count < 10:
        print(a, end=" ")
        a, b = b, a + b
        count += 1

# Call the function to print the first 10 Fibonacci numbers
fibonacci()
```

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

```
input_string = 'pwskills'
output_list = [char for char in input_string if char not in 'w']
print(output_list)


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

def is_palindrome(num):
    original_num = num
    reversed_num = 0

    while num > 0:
        remainder = num % 10
        reversed_num = reversed_num * 10 + remainder
        num = num // 10

    return original_num == reversed_num

# Example usage
number = int(input("Enter a number: "))
if is_palindrome(number):
    print(number, "is a palindrome.")
else:
    print(number, "is not a palindrome.")


#Write a code to print odd numbers from 1 to 100 using list comprehension.
#Note: Use a list comprehension to create a list from 1 to 100 and use another List comprehension to filter  out odd numbers.


# Creating a list of numbers from 1 to 100
numbers = [i for i in range(1, 101)]

# Filtering out odd numbers
odd_numbers = [num for num in numbers if num % 2 != 0]

# Printing the odd numbers
print(odd_numbers)
```

#Write a python program using reduce function to compute the product of a list containing numbers from 1 to 25.