Task 1: Without using Function

Prompt:

```python
#without using function factorial of a number

n = 5

fact = 1
for i in range(1,n+1):
    fact = fact * i

print(fact)
```

```
C:\Users\aetur\OneDrive\Desktop\DSA training>C:/Users/aetur/AppData/Local/Programs/Python/Python313/python.
120
```

Explanation:

- This code calculates the factorial of 5.
- It multiplies numbers from 1 to 5 using a loop and stores the result in `fact`.
- Finally, it prints the factorial value, which is **120**

Task 2 : Optimize code

Prompt :

```python
#optimize code

n = 5
fact = 1
i = 1
while i <= n:
    fact = fact * i
    i = i + 1
print(fact)
```

```
C:\Users\aetur\OneDrive\Desktop\DSA training>C:/Users/aetur/AppData/Local/Programs/
120
```

Explanation :

- This code calculates the factorial of 5 using a `while` loop.
- It multiplies numbers from 1 to 5 and stores the result in `fact`.
- Finally, it prints the factorial value, which is **120**.

Task 3 : with using function

Prompt:

```python
#with using function factorial of a number
def factorial(n):
    fact = 1
    for i in range(1,n+1):
        fact = fact * i
    return fact
n = 5
print(factorial(n))
```

```
C:\Users\aetur\OneDrive\Desktop\DSA training>C:/Users/aetur/AppData/Local/Programs/Python/
120
```

Explanation :

- This code defines a function `factorial()` that calculates the factorial of a number using a loop.
- The function multiplies numbers from 1 to n and returns the result.
- When called with n = 5, it prints the factorial value **120**.

Task 4: with and without using function factorial of a number

Prompt :

```python
#with and without using function and without recursion
def factorial(n):
    if n == 0 or n == 1:
        return 1
    else:
        fact = 1
        i = 1
        while i <= n:
            fact = fact * i
            i = i + 1
        return fact
n = 5
print(factorial(n))

print("Without function :",end = ' ')
fact = 1
for i in range(1,n+1):
    fact = fact * i
print(fact)
```

```
C:\Users\aetur\OneDrive\Desktop\DSA training>C:/Users/aetur/AppData/Local/Programs/Python/Python313/python.exe
120
Without function : 120
```

Explanation:

- This program calculates the factorial of a number in three ways: using a recursive function, a `for` loop, and an optimized `while` loop.
- The recursive function calls itself to compute the factorial, while the loops compute it iteratively.
- The `while` loop version is considered optimized because it avoids recursion and uses minimal memory and operations.

Task 5: By using Recursion

Prompt :

```python
#by using recursion
def factorial(n):
    if n == 0 or n == 1:
        return 1
    else:
        return n * factorial(n-1)
n = 5
print(factorial(n))
```

```
C:\Users\aetur\OneDrive\Desktop\DSA training>C:/Users/aetur/AppData/Local/Programs/Python/Python313/python.exe "c
120
```

Explanation :

- This code calculates the factorial of a number using recursion.
- The function calls itself with `n-1` until it reaches the base case (`n = 0` or 1).
- Then it multiplies the values while returning back, giving the factorial 120 for `n = 5`.