AI ASSISTED CODING

NAME:ERRA SRINIDHI REDDY

HALL TICKET NUMBER:2303A51566

BATCH:09

#Task-01

#write a program to check whether the given number is prime or not without using functions

#take input from user

```
num = int(input("Enter a number: "))
if num > 1:
    for i in range(2, int(num**0.5) + 1):
        if (num % i) == 0:
            print(f"{num} is not a prime number.")
            break
    else:
        print(f"{num} is a prime number.")
else:
    print(f"{num} is not a prime number.")
```

#Task-02

#generate optimized version of above code using function

#take input from user

```
def is_prime(n):
    if n <= 1:
```

```
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True
num = int(input("Enter a number: "))
if is_prime(num):
    print(f"{num} is a prime number.")
else:
    print(f"{num} is not a prime number.")
```

OBSERVATION:

Task 1: The program checks whether a number is prime using a loop and displays the result directly without using any function.

- The entire logic is written in one place.
- If we want to check another number, we should write the same logic again.

Task 2: The program checks whether a number is prime using a function, which makes the code easier to understand and reuse again

- Prime checking logic is placed inside a function `is prime()`
- Code became shorter and clear

In task 01 I observed two else conditions which made code complicated in task two there is only one If and else it is simple and easy to understand.

In task -02 Using functions makes the program more organized, reusable, and easy to understand compared to writing the logic directly without functions.
