***Prime Number Generator and Checker***

***Report***

***Name - Himanshu Malik***

***Branch- CSEAI-B***

***Uni. No.-202401100300123***

***Submitted to - Shivansh***

***Course name - AI***

***Institution: Kiet group of Institutions***

***Date -11 March 2025***

***Prime Number Generator and Checker***

***Introduction***

*This Python code provides two functions: a prime number checker and a prime number generator. The is\_prime(n) function checks if a number n is prime by testing divisibility up to its square root. The prime\_generator(limit) function generates prime numbers up to a given limit using the is\_prime() function. This approach ensures efficient prime checking and generation. The code can be used for mathematical applications or any program requiring prime number operations.*

***METHODOLOGY***

The code consists of two main components: a prime number checker and a prime number generator. The **prime number checker** (is\_prime(n)) works by testing whether a number n is divisible by any integer between 2 and the square root of n. If any divisor is found, the number is not prime; otherwise, it is considered prime. This method improves efficiency by reducing the number of checks needed.

The **prime number generator** (prime\_generator(limit)) generates all prime numbers up to a specified limit. It uses the is\_prime() function to check each number from 2 to the limit, yielding primes one at a time. This approach ensures that prime numbers are produced in a memory-efficient manner, without storing all primes at once, making it suitable for handling large ranges.

***CODE TYPED***

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

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

if is\_prime(number):

    print(f"{number} is a prime number.")

else:

    print(f"{number} is not a prime number.")

***ScreenShots Output photo pasted***

