

PRIME NUMBER GENERATOR

* **Project Name:** PRIME NUMBER GENERATOR
* **Name:** [ARYAN KUMAR SRIVASTAVA]
* **Roll No:** [202401100300069]
* **Date:** [11/3/2025]

|  |
| --- |
| introduction |

**This project focuses on generating and verifying prime numbers using Python. A prime number is a number greater than 1 that has no divisors other than 1 and itself. The program randomly generates prime numbers within a given range, allows users to check if a number is prime, and visualizes prime number distributions.**

|  |
| --- |
| methodology |

* Prime Number Check – The is\_prime(n) function determines whether a number is prime using an optimized approach.
* Random Prime Generator – The generate\_random\_prime(min\_val, max\_val) function randomly selects a number within a range and checks if it is prime.
* User Input Check – Users can input a number to check its primality.

Visualization – The program uses Matplotlib to display prime distributions and their cumulative count up to a given limit.

|  |
| --- |
| code |

import random

def is\_prime(n):

"""Check if a number is prime."""

if n <= 1:

return False

if n <= 3:

return True

if n % 2 == 0 or n % 3 == 0:

return False

i = 5

while i \* i <= n:

if n % i == 0 or n % (i + 2) == 0:

return False

i += 6

return True

def generate\_random\_prime(min\_val=2, max\_val=100):

"""Generate a random prime number within the given range."""

while True:

num = random.randint(min\_val, max\_val)

if is\_prime(num):

return num

# Generate and print a random prime number

min\_range = 2

max\_range = 1000

random\_prime = generate\_random\_prime(min\_range, max\_range)

print(f"Generated prime number: {random\_prime}")

# Check if a user-input number is prime

number\_to\_check = int(input("Enter a number to check if it's prime: "))

result = is\_prime(number\_to\_check)

print(f"{number\_to\_check} is {'prime' if result else 'not prime'}")

# Visualizing prime numbers

import matplotlib.pyplot as plt

import numpy as np

def get\_primes\_up\_to(limit):

"""Get all prime numbers up to the given limit."""

primes = [num for num in range(2, limit + 1) if is\_prime(num)]

return primes

# Set the limit

limit = 100

primes = get\_primes\_up\_to(limit)

# Create data for visualization

all\_numbers = list(range(2, limit + 1))

is\_prime\_list = [1 if is\_prime(n) else 0 for n in all\_numbers]

plt.figure(figsize=(12, 6))

# Plot 1: Prime distribution

plt.subplot(1, 2, 1)

plt.scatter(all\_numbers, is\_prime\_list, color='blue', alpha=0.7)

plt.xlabel('Number')

plt.ylabel('Is Prime (1=Yes, 0=No)')

plt.title(f'Distribution of Prime Numbers up to {limit}')

|  |  |  |
| --- | --- | --- |
| Output/Result |  | References/Credits |
|  | Python Standard Library (random, matplotlib, numpy)  Prime number algorithms used from number theory concepts |

3. GRAPH

