# Program to print prime numbers from 1 to N.

Difficulty Level : Easy   •   Last Updated : 14 Sep, 2022

Given a number N, the task is to print the prime numbers from 1 to N.
**Examples:**

```
Input: N = 10
Output: 2, 3, 5, 7


Input: N = 5
Output: 2, 3, 5
```

Recommended: Please try your approach on *{IDE}* first, before moving on to the solution.

**Algorithm:**

- First, take the number N as input.
- Then use a for loop to iterate the numbers from 1 to N
- Then check for each number to be a prime number. If it is a prime number, print it.

**Approach 1:**  *Now, according to formal definition, a number 'n' is prime if it is not divisible by any number other than 1 and n. In other words a number is prime if it is not divisible by any number from 2 to n-1.*

Below is the implementation of the above approach:

C++

```cpp
// C++ program to display Prime numbers till N
#include <bits/stdc++.h>
using namespace std;

// function to check if a given number is prime
bool isPrime(int n)
{
    // since 0 and 1 is not prime return false.
    if (n == 1 || n == 0)
        return false;

    // Run a loop from 2 to n-1
    for (int i = 2; i < n; i++) {
        // if the number is divisible by i, then n is not a
        // prime number.
        if (n % i == 0)
            return false;
    }
    // otherwise, n is prime number.
    return true;
}

// Driver code
int main()
{
    int N = 100;

    // check for every number from 1 to N
    for (int i = 1; i <= N; i++) {
        // check if current number is prime
        if (isPrime(i))
            cout << i << " ";
```

C

```c
// C program to display Prime numbers till N
#include <stdbool.h>
#include <stdio.h>

// function to check if a given number is prime
bool isPrime(int n)
{
    // since 0 and 1 is not prime return false.
    if (n == 1 || n == 0)
        return false;

    // Run a loop from 2 to n-1
    for (int i = 2; i < n; i++) {
        // if the number is divisible by i, then n is not a
        // prime number.
        if (n % i == 0)
            return false;
    }
    // otherwise, n is prime number.
    return true;
}

// Driver code
int main()
{
    int N = 100;

    // check for every number from 1 to N
    for (int i = 1; i <= N; i++) {
        // check if current number is prime
        if (isPrime(i))
            printf("%d ", i);
    }

    return 0;
}

// This code is contributed by Sania Kumari Gupta
```

```java
// Java program to display Prime numbers till N
class GFG
{

    //function to check if a given number is prime
    static boolean isPrime(int n){
        //since 0 and 1 is not prime return false.
        if(n==1||n==0)return false;

        //Run a loop from 2 to n-1
        for(int i=2; i<n; i++){
          // if the number is divisible by i, then n is not a prime number.
            if(n%i==0)return false;
        }
        //otherwise, n is prime number.
        return true;
    }


    // Driver code
    public static void main (String[] args)
    {
        int N = 100;
        //check for every number from 1 to N
        for(int i=1; i<=N; i++){
            //check if current number is prime
            if(isPrime(i)) {
                System.out.print(i + " ");
            }
        }

    }
}
```

## Python3

```python
# Python3 program to display Prime numbers till N

#function to check if a given number is prime
def isPrime(n):
  #since 0 and 1 is not prime return false.
  if(n==1 or n==0):
    return False

  #Run a loop from 2 to n-1
```

```python
        return False

    #otherwise, n is prime number.
    return True



# Driver code
N = 100;
#check for every number from 1 to N
for i in range(1,N+1):
    #check if current number is prime
    if(isPrime(i)):
        print(i,end=" ")
```

## C#

```csharp
// C# program to display Prime numbers till N
using System;

class GFG
{

    //function to check if a given number is prime
    static bool isPrime(int n){
        //since 0 and 1 is not prime return false.
        if(n==1||n==0) return false;

        //Run a loop from 2 to n-1
        for(int i=2; i<n; i++) {
            // if the number is divisible by i, then n is not a prime number.
            if(n%i==0) return false;
        }
        //otherwise, n is prime number.
        return true;
    }

    // Driver code
    public static void Main (String[] args)
    {
        int N = 100;
        //check for every number from 1 to N
        for(int i=1; i<=N; i++) {
            //check if current number is
            if(isPrime(i)) {
```

```
        }
    }

    // This code is contributed by Rajput-Ji
```

## Javascript

```html
<script>


// JavaScript program to display Prime numbers till N

// function to check if a given number is prime
function isPrime( n)
{
    // since 0 and 1 is not prime return false.
    if(n == 1 || n == 0) return false;

    // Run a loop from 2 to n-1
    for(var i = 2; i < n; i++)
    {

      // if the number is divisible by i, then n is not a prime number.
      if(n % i == 0) return false;
    }
    // otherwise, n is prime number.
    return true;
}


// Driver code
var N = 100;

// check for every number from 1 to N
  for(var i = 1; i <= N; i++)
  {
      // check if current number is prime
      if(isPrime(i)) {
        console.log( i );
      }
}

 This code is contributed by ukasp.
</script>
```

**Output**

2  3  5  7  11  13  17  19  23  29  31  37  41  43  47  53  59  61  67  71  73  79  83  89  97

**Time Complexity:** O(N^2),

**Auxiliary Space:** O(1)

*Approach 2*: *For checking if a number is prime or not do we really need to iterate through all the number from 2 to n-1? We already know that a number 'n' cannot be divided by any number greater than 'n/2'. So, according to this logic we only need to iterate through 2 to n/2 since number greater than n/2 cannot divide n.*

C++

```cpp
// C++ program to display Prime numbers till N
#include <bits/stdc++.h>
using namespace std;

//function to check if a given number is prime
bool isPrime(int n){
    //since 0 and 1 is not prime return false.
    if(n==1||n==0) return false;

    //Run a loop from 2 to n/2.
    for(int i=2; i<=n/2; i++) {
        // if the number is divisible by i, then n is not a prime number.
        if(n%i==0) return false;
    }
    //otherwise, n is prime number.
    return true;
}


// Driver code
int main()
{
    int N = 100;

    //check for every number from 1 to
```

```
        cout << i << " ";
    }
}

    return 0;
}
```

## Java

```java
// Java program to display
// Prime numbers till N
class GFG
{
    //function to check if a given number is prime
    static boolean isPrime(int n){
        //since 0 and 1 is not prime return false.
        if(n==1||n==0) return false;

        //Run a loop from 2 to n-1
        for(int i=2; i<=n/2; i++){
            // if the number is divisible by i, then n is not a prime number.
            if(n%i==0)return false;
        }
        //otherwise, n is prime number.
        return true;
    }


    // Driver code
    public static void main (String[] args)
    {
        int N = 100;
        //check for every number from 1 to N
        for(int i=1; i<=N; i++){
            //check if current number is prime
            if(isPrime(i)) {
                System.out.print(i + " ");
            }
        }

    }
}
```

```python
# Python3 program to display Prime numbers till N

#function to check if a given number is prime
def isPrime(n):
  #since 0 and 1 is not prime return false.
  if(n==1 or n==0):
    return False

  #Run a loop from 2 to n/2
  for i in range(2,(n//2)+1):
    #if the number is divisible by i, then n is not a prime number.
    if(n%i==0):
      return False

  #otherwise, n is prime number.
  return True



# Driver code
N = 100;
#check for every number from 1 to N
for i in range(1,N+1):
  #check if current number is prime
  if(isPrime(i)):
    print(i,end=" ")
```

# C# ▼

```csharp
// C# program to display
// Prime numbers till N
using System;

class GFG
{

  //function to check if a given number is prime
  static bool isPrime(int n){
      //since 0 and 1 is not prime return false.
    if(n==1||n==0)return false;

      //Run a loop from 2 to n/2.
    for(int i=2; i<=n/2; i++){
        // if the number is divisible b ▲ then n is not a prime number.
```

```csharp
        return true;
}

// Driver code
public static void Main (String[] args)
{
    int N = 100;
    //check for every number from 1 to N
      for(int i=1; i<=N; i++){
      //check if current number is prime
      if(isPrime(i)) {
        Console.Write(i + " ");
      }
    }

}
}

// This code is contributed by Rajput-Ji
```

## Javascript

```javascript
<script>
// Javascript program to display Prime numbers till N

// function to check if a given number is prime
function isPrime(n)
{

    // since 0 and 1 is not prime return false.
    if(n == 1 || n == 0) return false;

    // Run a loop from 2 to n/2.
    for(let i = 2; i <= n / 2; i++)
    {
        // if the number is divisible by i, then n is not a prime number.
        if(n % i == 0) return false;
    }

    // otherwise, n is prime number.
    return true;
}

// Driver code
```

```javascript
for(let i = 1; i <= N; i++)
{
    // check if current number is prime
    if(isPrime(i))
    {
        document.write(i + " ");
    }
}

// This code is contributed by shubham348.
</script>
```

**Output**

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

**Time Complexity:** $O(N^2)$,

**Auxiliary Space:** $O(1)$, since no extra space has been taken.

> **Approach 3:** *If a number 'n' is not divided by any number less than or equals to the square root of n then, it will not be divided by any other number greater than the square root of n. So, we only need to check up to the square root of n.*

---

## C++

```cpp
// C++ program to display Prime numbers till N
#include <bits/stdc++.h>
using namespace std;

//function to check if a given number is prime
bool isPrime(int n){
    //since 0 and 1 is not prime return false.
    if(n==1||n==0)return false;

    //Run a loop from 2 to square root of n.
    for(int i=2; i*i<=n; i++){
        // if the number is divisible by i,      n is not a prime number.
```

```cpp
    return true;
}


// Driver code
int main()
{
    int N = 100;

    //check for every number from 1 to N
      for(int i=1; i<=N; i++){
      //check if current number is prime
      if(isPrime(i)) {
        cout << i << " ";
      }
    }

    return 0;
}
```

## Java

▼

```java
// Java program to display
// Prime numbers till N
class GFG
{
  //function to check if a given number is prime
  static boolean isPrime(int n){
   //since 0 and 1 is not prime return false.
   if(n==1||n==0)return false;

   //Run a loop from 2 to square root of n
   for(int i=2; i*i<=n; i++){
     // if the number is divisible by i, then n is not a prime number.
     if(n%i==0)return false;
   }
   //otherwise, n is prime number.
   return true;
  }


// Driver code
ublic static void main (String[] args)

    int N = 100;
```

▲

```
      if(isPrime(i)) {
        System.out.print(i + " ");
      }
    }


  }
}
```

## Python3

```python
# Python3 program to display Prime numbers till N

#function to check if a given number is prime
def isPrime(n):
  #since 0 and 1 is not prime return false.
  if(n==1 or n==0):
    return False

  #Run a loop from 2 to square root of n.
  for i in range(2,int(n**(1/2))+1):
    #if the number is divisible by i, then n is not a prime number.
    if(n%i==0):
      return False

  #otherwise, n is prime number.
  return True



# Driver code
N = 100;
#check for every number from 1 to N
for i in range(1,N+1):
  #check if current number is prime
  if(isPrime(i)):
    print(i,end=" ")
```

## C#

```csharp
// C# program to display
// Prime numbers till N
using System;
```

```csharp
//function to check if a given number is prime
static bool isPrime(int n){
    //since 0 and 1 is not prime return false.
    if(n==1||n==0)return false;

    //Run a loop from 2 to square root of n.
    for(int i=2; i*i<=n; i++){
        // if the number is divisible by i, then n is not a prime number.
        if(n%i==0)return false;
    }
    //otherwise, n is prime number.
    return true;
}

// Driver code
public static void Main (String[] args)
{
    int N = 100;
    //check for every number from 1 to N
    for(int i=1; i<=N; i++){
        //check if current number is prime
        if(isPrime(i)) {
            Console.Write(i + " ");
        }
    }

}
}

// This code is contributed by Rajput-Ji
```

## Javascript

```javascript
<script>
// JavaScript program to display Prime numbers till N

// function to check if a given number is prime
const isPrime = (n) => {

    // since 0 and 1 is not prime return false.
    if(n === 1||n === 0)return false;

    // Run a loop from 2 to square root of n.
    for(let i = 2; i <= Math.floor(Math        (n)); i++)
```

```
        if(n % i == 0)return false;
    }

    // otherwise, n is prime number.
    return true;
}


    // Driver code

    let N = 100;

    // check for every number from 1 to N
    for(let i=1; i<=N; i++)
    {

        // check if current number is prime
        if(isPrime(i)) {
            document.write(i);
        }
    }

    // This code is contributed by shinjanpatra
</script>
```

**Output**

2  3  5  7  11  13  17  19  23  29  31  37  41  43  47  53  59  61  67  71  73  79  83  89  97

**Time Complexity:** O(N^(3/2)),

**Auxiliary Space:** O(1)

You can further optimize the time complexity to **O(n\*log(log(n)))**. Check **Sieve of Eratosthenes**.

▲

Like  45

Previous                                                                    Next

## RECOMMENDED ARTICLES

<span style="float:right">Page : **1** 2 3</span>

**01** Count numbers in a given range having prime and non-prime digits at prime and non-prime positions respectively
15, Jan 21

**02** Print numbers such that no two consecutive numbers are co-prime and every three consecutive numbers are co-prime
22, Jan 19

**03** Print the nearest prime number formed by adding prime numbers to N
26, Jul 18

**04** Print all numbers whose set of prime factors is a subset of the set of the prime factors of X
22, Apr 19

**05** Print prime numbers with prime sum of digits in an array
15, Jan 19

**06** Absolute difference between the Product of Non-Prime numbers and Prime numbers of an Array
03, Nov 18

**07** Absolute difference between the XOR of Non-Prime numbers and Prime numbers of an Array
23, Sep 19

**08** Count prime numbers that can be expressed as sum of consecutive prime numbers
10, Sep 20

▲

**Code_r**
@Code_r

## Vote for difficulty

Current difficulty : _Easy_

| Easy | Normal | Medium | Hard | Expert |
|------|--------|--------|------|--------|

**Improved By :**  ankthon,  Rajput-Ji,  vishnusreddy,  _saurabh_jaiswal,  gabaa406, gsingh6122016,  ukasp,  singhdushyant260,  shinjanpatra,  subham348, krisania804,  souravkumar29

**Article Tags :**  Prime Number,  Mathematical

**Practice Tags :**  Mathematical,  Prime Number

| Improve Article | Report Issue |
|-----------------|--------------|

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

| Load Comments |
|---------------|

▲

# Start Your Coding Journey Now!

## Company

About Us

Careers

In Media

Contact Us

Privacy Policy

Copyright Policy

## Learn

Algorithms

Data Structures

SDE Cheat Sheet

Machine learning

CS Subjects

Video Tutorials

Courses

## News

Top News

Technology

Work & Career

Business

Finance

Lifestyle

Knowledge

## Languages

Python

Java

CPP

Golang

C#

SQL

Kotlin

## Web Development

Web Tutorials

Django Tutorial

HTML

JavaScript

Bootstrap

ReactJS

NodeJS

## Contribute

Write an Article

Improve an Article

Pick Topics to Write

Write Interview Experience

Internships

Video Internship