IsPrime()in C# By Firdous Samreen(1950360)

A number ‘n’, is considered a prime number if apart from 1 and itself, it has zero factors.

To achieve this, we can loop through numbers from 1 to n, checking if it is a factor of n.

Initial Solution:

```
for (int possible_factor = 2; possible_factor < n; possible_factor++)

    if (n % possible_factor == 0)

        return false;

return true;
```

To optimize this, we can reduce the limit of loop till square root of n (Suppose axb =n, we know that one factor is less than sqrt(n) whereas other is greater, so we do not need to check twice)

We can further optimize this, by reducing the number of iterations

Consider the numbers:

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24
25	26	27	28	29	30

If we consider the divisibility of 2, 3 we rule out the following columns

1	~~2~~ ~~3~~ ~~4~~	5	~~6~~
7	~~8~~ ~~9~~ ~~10~~	11	~~12~~
13	~~14~~ ~~15~~ ~~16~~	17	~~18~~
19	~~20~~ ~~21~~ ~~22~~	23	~~24~~
25	~~26~~ ~~27~~ ~~28~~	29	~~30~~

Suppose a number from columns 2,3,4,6 is a factor, it means that 2 or 3 must also be a factor. Hence, if we check 2 or 3 is a factor, we can skip these numbers.

Also, If we observe we can see that the remaining columns are of the forms 6n+1 of 6n-1 (except 1)

	5(6x1-1)
7(6x1+1)	11(6x2-1
13(6x2+1)	17(6x3-1)
19(6x3+1)	23(6x4-1) and so on…

Therefore, we can check 2, 3, numbers of the form 6*n-1, 6*n+1 for the possible factors instead of checking every number

My end solution:

```
public static bool isPrime(int n)
{
   if(n == 2 || n == 3)
     return true;
   //if n =1 or 2,3,5 is a factor of n then return false
   if(n % 2 == 0 || n % 3 ==0 || n%5 == 0 || n == 1)
     return false;



//set limit to sqaure root of n
   int limit = (int)System.Math.Ceiling(  System.Math.Sqrt(n));


   //check if number of form 6n+1, 6n-1 is a factor
```

```
for(int i = 6; i < limit; i+=6)
    if(n % (i + 1) == 0|| n%(i - 1) == 0)
      return false;
  return true;
}
```

Some numbers (e.g:25), skips the loop because sqrt(25)<6(loop condition not satisfied).So we have considered divisibility of 5 to deal with such cases and for a faster algorithm.

Resources:

https://stackoverflow.com/questions/15743192/check-if-number-is-prime-number

https://www.youtube.com/watch?v=5liC14kBrQ4

https://mae.ufl.edu/~uhk/sixnplusone.pdf