Prime Number

↳ A number only divisible by 1 and itself.

2, 3, 5, 7, 11, 13, ...

⑥ →

```
boolean checkPrime(int n){
    if(n<2) {
    return false;
     }
    if(n==2){
     return true;
       }
   for(int i=2; i<=(int)Math.sqrt(n);i++)
     if(n%i==0){
     return false;
      }
   }
   return true;
```

0, 1, 2, 3 — ..

3, 4, 5, 6, 7, 8 ..

1, 3 ② 2, 3

}

Why are we checking upto Square root

40 → 2

40 % 2 == 0 → false.

41 → 2 to 40.

42 → 2 to 6

42 → 21.

1, 78

78 → 6, 2

13

39

Factors of a number

20 → 1, 2, 4, 5, 10, 20

i = 1 to 20

20 % i == 0

↓ i is factor.

Array → It is collection of elements of similar data type.

20, 5, 6, 7, 100, 204, 380, 14, 11, 75, ..

```
int a=20;
int b=5;
int c=6;
int d=7;
int e=100;
```

int a—
int a[] → array

data-type name_of_array[] = new data-type[size];

```
int arr[] = new int[6];
arr[0]=100;
arr[1]=200;
arr[2]=40;
arr[3]=5;
arr[4]=60;
arr[5]=72;
```

[100, 200, 40, 5, 60, 72]
0 1 2 3 4 5

index →

0 to 5.

```
int arr[] = new int[8];
arr[0]=10;
arr[1]=13;
arr[2]=100;
arr[3]=107;
```

```
arr[4] = 1009;
arr[5] = 987;
arr[6] = 464;
arr[7] = 67846;
```

[10, 13, 100, 107, 1009, 987, 464, 67846]
0 1 2 3 4 5 6 7

arr[7] → 67846.

arr[4] → 1009

arr[2] → 100

4, 6, 8, "abc", 10

↳ X

"4", "6", "8", "abc", "10"

↳ array of string