## MouseEvent button Property

- ✓ The button property returns which mouse button is pressed when a mouse event occurs.
- ✓ The button property is mostly used with the onmousedown event.
- ✓ The button property is read-only.

**Syntax**

```
event.button
```

**Return Value:** **A Number.**

Which mouse button that was pressed:

- 0 : Left button
- 1 : Wheel or middle button (if present)
- 2 : Right button

For a left-hand configured mouse, the values are reversed.

## KeyboardEvent key Property

- ✓ The key property returns the key that was pressed when the event occured.
- ✓ The key property is read-only.

**Syntax**

```
event.key
```

**Return Value:** **A String**

The key that was pressed:

- A single character ("A", "a", "4", "+", "$")
- Multiple characters ("F1", "Enter", "HOME", "CAPS LOCK")

## KeyboardEvent keyCode

- ✓ Get the value of the pressed keyboard key:

**The keyCode property is deprecated.**

**Syntax**

```
event.keyCode
```

**Example: Write JS to handle following key events**

**1) Give keycode for the key pressed**

**2) Script should give message “vowel is pressed” on pressing vowel key**

**3) Background color should change to red after releasing the pressed key**

```
<html>
<head>
<script>
function fun1()
{
c= (event.key);

if(c=='a' || c=='e' || c=='i' || c=='o' || c=='u' || c=='A' || c=='E' || c=='I' || c=='O'
|| c=='U')
{
   document.getElementById("p1").innerHTML= "Vowel : " + c + " > " +
event.keyCode
}
else
{
   document.getElementById("p1").innerHTML= "Consonants : " + c + " > " +
event.keyCode;
}
}
</script>
</head>
<body>
<input type="text" id="i1" onkeypress="fun1()" />
<p id="p1"></p>
</body>
</html>
```

## To check an Armstrong number

```
<script>
// In the case of an Armstrong number of 3 digits, the sum of cubes of each digit is equal to the number itself. For example, 153 is an Armstrong number because
// 153 = 1*1*1 + 5*5*5 + 3*3*3
// Similarly, 1634 is an Armstrong number because:
// 1634 = 1*1*1*1 + 6*6*6*6* + 3*3*3*3 + 4*4*4*4


let sum = 0;
const number = prompt('Enter a three-digit positive integer: ');

// create a temporary variable
let temp = number;
while (temp > 0) {
   // finding the one's digit
   let remainder = temp % 10;

   sum += remainder * remainder * remainder;

   // removing last digit from the number
   temp = parseInt(temp / 10); // convert float into integer
}
// check the condition
if (sum == number) {
   console.log(`${number} is an Armstrong number`);
}
else {
   console.log(`${number} is not an Armstrong number.`);
}
</script>
```

## Example: Check Prime Number

A prime number is a positive integer that is only divisible by **1** and itself. For example, **2**, **3**, **5**, **7**, **11** are the first few prime numbers.

```
<script>
// program to check if a number is prime or not
// take input from the user
const number = parseInt(prompt("Enter a positive number: "));
let isPrime = true;
// check if number is equal to 1
if (number === 1) {
   console.log("1 is not a prime number");
}
// check if number is greater than 1
else if (number > 1) {
      if (number % 2 == 0) {
         console.log(number+ " is a not prime number");
      }
      else{
         console.log(number + " is a prime number");
      }
   }
// check if number is less than 1
else {
   console.log("The number is not a prime number.");
}

</script>
```

**Example 1:** **Write HTML form accepting an integer having four digits. Input should not accept characters of letters and special symbols.**

**Example 2:** **Write a JS to validate username and password.**
**Password: Length must be of 6 to 12 characters.**
**Username: Should not start with _, @ and any number.**
**Both must not be blank**

**Example 3:** **Design a login form using JS. Following validation in password field, Minimum length of password must be of 8 letters and it must have some special characters.**

**Example: Show validation using JS on fields like name, phone number and email id**
RE for Username: /^[A-z]+$/;
RE for Phone number: /^\d{10}$/;
RE for email: /^\w+([\.-]?\w+)*@\w+([\.-]?\w+)*(\.\w{2,3})+$/;