

# 1. JavaScript const

The **const** keyword allows you to declare a constant

Use **const** when you know that the value should not be changed.

للإعلان عن ثابت

# 2. JavaScript let

allows you to declare a variable with block scope

للاستخدام داخل نطاق محدد

# 3. Arrow Functions

You don't need the function keyword, the return keyword, and the curly brackets.

`const x = (prams) => .....`

# 4. The Spread (...) Operator

The ... operator expands an iterable (like an array) into more elements:

# 5. Array findIndex()

The `findIndex()` method returns the index of the first array element that passes a test function

# 6. The Math.trunc()

`Math.trunc(x)` returns the integer part of number

# 7. Modules

Allows you to isolate parts of the code for better organization and reuse  
`import {...} from 'module'`

## 8. JavaScript Maps

Being able to use an Object as a key is an important Map feature

## 9. Default Parameter Values

ES6 allows function parameters to have default values.

## 10. String.includes()

The includes() method returns true if a string contains a specified value, otherwise false

## 11. Array.from()

The Array.from() method returns an Array object from any object with a length property or any iterable object

---

### // fetch 1

```
fetch('https://jsonplaceholder.typicode.com/todos/1')
  .then(response => response.json())
  .then(data => {
    console.log('fetch 1:', data);
  })
```

## // fetch 2

```
async function fetchData() {  
  try {  
    const response = await fetch('https://jsonplaceholder.typicode.com/todos/1');  
    const data = await response.json();  
    console.log('fetch 2:', data);  
  } catch (error) {  
  }  
}  
fetchData();
```

## // fetch 3

```
// axios  
useEffect(() => {  
  axios.get("https://jsonplaceholder.typicode.com/todos/1")  
    .then((response) => console.log('fetch 3 : axios', response.data));  
}, []);
```

## // fetch 4

```
useEffect(() => {  
  fetch('https://jsonplaceholder.typicode.com/todos/1')  
    .then(response => response.json())  
    .then(data => console.log('fetch 4:', data));  
}, []);
```

