Types of JS Functions

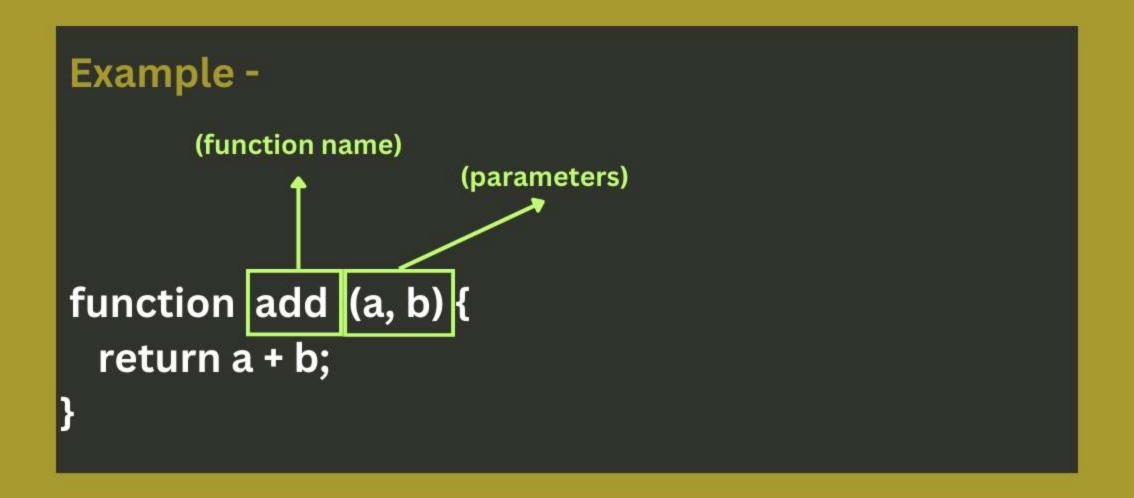
swipe right

curated by Swarnava Dhar

1. Named Functions

Definition -

Functions that have a specific name.





2. Anonymous Functions:

Definition -

Functions without a name, often used as arguments to other functions or assigned to variables.

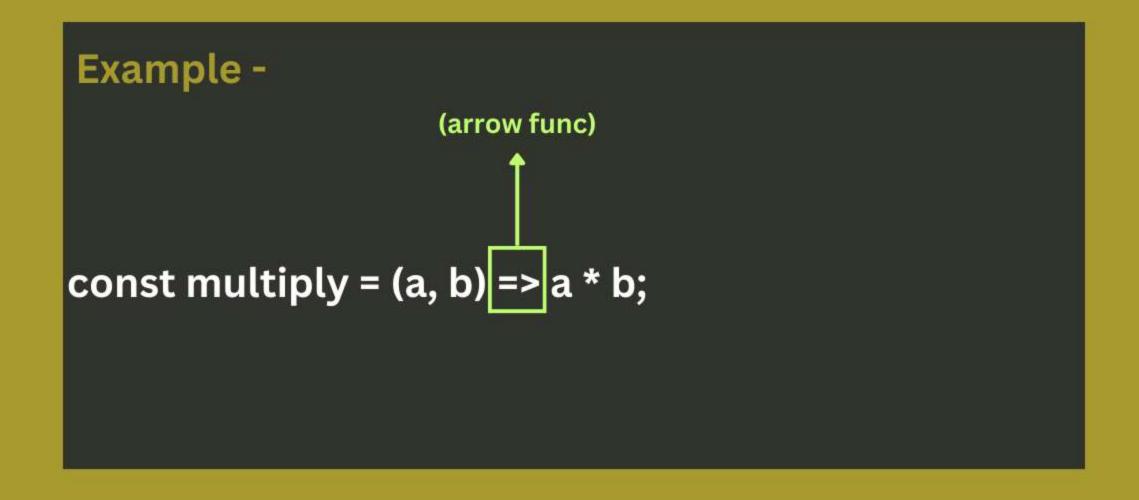
```
Example -
const subtract = function(a, b) {
  return a - b;
};
```



3. Arrow Functions:

Definition -

A more concise syntax for writing anonymous functions, introduced in ES6.



4. Immediately Invoked Function Expressions (IIFE)

Definition -

Functions that are executed immediately after being defined.

```
Example -

(function() {
   console.log("This is an IIFE");
}) ();
```



5. Constructor Functions:

Definition -

Used to create objects and are invoked with the new keyword.

```
Example -
function Person(name, age) {
  this.name = name;
  this.age = age;
}
const person1 = new Person('Alice', 30);
```



6. Generator Functions:

Definition -

Functions that can be paused and resumed, returning an iterator.

```
Example -

function* generatorFunction() {
    yield 1;
    yield 2;
    yield 3;
}

const generator = generatorFunction();
console.log(generator.next().value); // 1
```



7. Async Functions:

Definition -

Functions that handle asynchronous operations more gracefully, introduced in ES8

```
Example -
async function fetchData(url) {
  const response = await fetch(url);
  const data = await response.json();
  return data;
}
```



8. Method Functions:

Definition -

Functions defined within objects.

```
Example -
const obj = {
   greet: function() {
    return 'Hello!';
   }
};
console.log(obj.greet());
```



9. Callback Functions:

Definition -

Functions passed as arguments to other functions and called within the outer function to complete some kind of routine or action.

```
Example -
function fetchData(callback) {
    setTimeout(() => {
        callback('Data received');
    }, 1000);
}
fetchData((message) => {
    console.log(message);
});
```