ES6 JS

Javascript Features You need to know

Arrow Functions

The **arrow function** provides a **shorter syntax** for defining the functions.

Syntax

```
let func = (arg1, arg2, ..., argN) => expression;
```

function name

arguments

any valid JS expression

Example

```
let sum = (a, b) => a + b;
```

(a, b) => a + b means a function that accepts two arguments named a and b.

Upon the execution, it evaluates the expression a + b and returns the result.

Important Tip

- If we have only one argument, then parentheses around parameters can be omitted.

```javascript
let double = n => n * 2;
```

- If there are no arguments, parentheses are empty, but they **must be present:**

```javascript
let sayHi = () => alert("Hello!");
```

Multi-line Arrow Functions

- To write multiline functions, we need to use curly braces and explicit return statement.

```javascript
let sum = (a, b) => {
    let result = a + b;
    return result;
};
```

Template Strings #2

Template literals provide an easy way to interpolate variables and expressions into strings. *They are enclosed by backticks (` `) and ${expression} as a placeholder.*

```javascript
const number = 42;
const message = `The number is ${number}`;

console.log(message); // => 'The number is 42'
```

Another Example

```javascript
const person = {
  name: "Zodiac Hasbro",
  age: 56
};

const greeting = `Hello,I am ${person.name}!`
```

The console will display: Hello, I am Zodiac Hasbro!

There are more ES6 features like **Spread Operators and Rest Parameters**, these will be discussed in the next post.

Stay Tuned!