ES6 Template Literals

Template literals are a powerful feature introduced in ES6 that provide a more flexible and readable way to create strings in JavaScript.

Key Features:

- **Multi-line Strings:**
    - Easily create multi-line strings using backticks (``) instead of single or double quotes.
    - No need for escape characters for newlines.

JavaScript

```
const message = `This is a
multi-line
string.`;
```

- **Embedded Expressions:**
    - Embed JavaScript expressions directly within the string using ${}.
    - This allows for dynamic string creation.

JavaScript

```
const name = "John";
const age = 30;
const greeting = `Hello, my name is ${name} and I am ${age} years old.`;
```

- **Tagged Templates:**
    - Create custom string processing functions (tag functions) that can manipulate the string before it's created.

JavaScript

```
function highlight(strings, ...values) {
  return strings.reduce((acc, curr, i) =>
   acc + `<span class="highlight">${values[i]}</span>` + curr
 );
}

const message = highlight`This is a ${'highlighted'} message.`;
```

Benefits of Template Literals:

- **Improved Readability:**
  - Make strings more readable and easier to maintain.
  - Eliminate the need for string concatenation.
- **Enhanced Flexibility:**
  - Allow for dynamic string creation with embedded expressions.
  - Enable custom string processing with tagged templates.
- **Reduced Errors:**
  - Reduce the risk of errors caused by incorrect string concatenation or escaping.

In Summary:

Template literals are a valuable addition to the JavaScript language. They offer a more concise, readable, and flexible way to work with strings, making your code cleaner and easier to understand.