JavaScript Syntax

Summary: in this tutorial, you will learn about JavaScript syntax, including whitespace, statements, identifiers, comments, expressions, and keywords.

Whitespace

Whitespace refers to characters that provide the space between other characters. JavaScript has the following whitespace:

- Carriage return

- Space

- New Line

- tab

JavaScript engine ignores whitespace. However, you can use whitespace to format the code to make it easy to read and maintain.

The following JavaScript code doesn't use whitespace:

```javascript
let formatted = true; if (formatted) {console.log('The code is easy to read');}
```

It is equivalent to the following code that uses whitespace. Hence, this code is much easier to read:

```javascript
let formatted = true;

if (formatted) {
  console.log('The code is easy to read');
}
```

Note that JavaScript bundlers remove all whitespace from JavaScript files and put them into a single file for deployment. By doing this, JavaScript bundlers make the JavaScript code lighter and faster to load in web browsers.

Statements

A statement is a piece of code that either declares a variable or instructs the JavaScript engine to perform a task. A simple statement ends with a semicolon (`;`).

> Note that the semicolon (`;`) is optional, but you should use it to terminate a statement.

For example, the following declares a variable and shows it to the console:

```javascript
let message = "Welcome to JavaScript";
console.log(message);
```

Blocks

A block is a sequence of zero or more simple statements. A block is delimited by a pair of curly brackets `{}` . For example:

```javascript
if (window.localStorage) {
  console.log('The local storage is supported');
}
```

Identifiers

An identifier is a name you choose for variables, parameters, functions, classes, etc.

An identifier name starts with a letter (`a-z` , or `A-Z`), an underscore(`_`), or a dollar sign (`$`) and is followed by a sequence of characters including (`a-z` , `A-Z`), numbers (`0-9`), underscores (`_`), and dollar signs (`$`).

> Note that the letter is not limited to the ASCII character set and may include extended ASCII or Unicode, though it is not recommended.

Identifiers in JavaScript are case-sensitive. For example, the `message` is different from the `Message` .

Comments

Comments allow you to include notes or hints in JavaScript code. When executing the code, the JavaScript engine ignores the comments.

JavaScript supports both single-line and block comments.

Single-line comments

A single-line comment starts with two forward-slashes characters (`//`). It turns all the text following the `//` on the same line into a comment. For example:

```
// this is a single-line comment
```

Block comments

A delimited comment begins with a forward slash and asterisk `/*` and ends with the opposite `*/` as in the following example:

```
/* This is a block comment
that can span multiple lines */
```

Expressions

An expression is a piece of code that evaluates to a value. For example:

```
2 + 1
```

The above expression returns three.

Keywords & Reserved words

JavaScript defines a list of reserved keywords that have specific uses. You cannot use the reserved keywords as identifiers.

The following table displays the JavaScript reserved words as defined in ECMA-262:

break	case	catch
continue	debugger	default
else	export	extends
function	if	import
new	return	super
throw	try	null
void	while	with
class	delete	finally
in	switch	typeof
yield	const	do
for	instanceof	this
var		

In addition to the reserved keywords, ECMA-252 also defines a list of future reserved words that cannot be used as identifiers:

enum	implements	let
protected	private	public
await	interface	package
implements	public	

Summary

- Use whitespace, including carriage return, space, newline, and tab to format the code. The JavaScript engine disregards whitespace.

- Use a semicolon (`;`) to terminate a simple statement.

- Use the curly braces (`{}`) to create a block that groups one or more simple statements.

- A single-line comment starts with `//` followed by text, while a block comment begins with `/*` and ends with `*/` . The JavaScript engine ignores comments.

- Identifiers are names you choose for variables, functions, classes, and so on.

- Avoid using reserved keywords for identifiers.

Quiz