

Theory: Comments

🕒 4 minutes

0 / 5 problems solved

Skip this topic

Start practicing

1761 users solved this topic. Latest completion was about 1 hour ago.

In JS scripts, there is a way to write pieces of text that will be ignored by the JS interpreter. Such texts are called **comments**. Why write them at all if in the end they're simply ignored? Well, the interpreter will ignore comments, but other developers won't. Comments allow you to make your code more readable and comprehensible for developers.

JavaScript language supports two types of comments: one-line and multi-line. Let's look at them more closely.

§1. One-line comments

Comments that occupy only one line are called **one-line comments**. Take a careful look at the syntactic features of these comments:

```
1 | console.log("Nice to see you!"); // This code outputs the message to the console
```

In the example above, all text after `//` is ignored by the interpreter. With the help of such notes, you will quickly remember what a certain code fragment is intended for, even if it's been a while since you last saw it.

§2. Multi-line comments

Comments that take up several lines are called **multi-line comments**. These are their syntactic features:

```
1 | /*
2 |    The following code outputs the message to the console
3 |    The console will display a line with the text "Hello, JS!"
4 | */
5 | console.log("Hello, JS!");
```

The interpreter ignores all text between `/*` and the nearest `*/`. Such comments are useful for explaining particularly complex pieces of code or temporarily commenting out code fragments when you need to find an error. Also, multi-line comments located at the beginning of a file are quite useful when you need to specify copyright information.

Sometimes you may find a slightly modified syntax of multi-line notes: the comment opens with the `/**` tag, and each line starts with an `*`:

```
1 | /**
2 |  * The first program
3 |  * Author: Bob
4 |  */
5 | console.log("Hi, I'm Bob");
```

Such comments often contain information about the programming file including its name, version, and the author of the script.

JavaScript does not support nested comments. This means that you will not be able to write a comment inside another comment.

§3. Conclusion

Comments make your code clearer and help simplify the work of all those programmers who will work with your code in the future. Comments are also used to temporarily comment out code fragments when you need to find an error. In other words, you can really benefit from comments, so don't ignore them, even though your JS interpreter will. Do not overuse comments, use them wisely, do not forget that comment can't replace well written code.

Current topic:

Comments ...

Topic depends on:

✗ Overview of the basic program ...

Topic is required for:

Variables ...

Strings and numbers ...

Table of contents:

[↑ Comments](#)

[§1. One-line comments](#)

[§2. Multi-line comments](#)

[§3. Conclusion](#)

[Feedback & Comments](#)

166 users liked this theory. 1 didn't like it. What about you?



Start practicing

[Comments \(0\)](#)[Hints \(0\)](#)[Useful links \(0\)](#)[Show discussion](#)