

JSDoc



JSDoc 3 is an API documentation generator for JavaScript, similar to Javadoc or phpDocumentor .

You add documentation comments directly to your source code, right alongside the code itself. The JSDoc tool will scan your source code and generate an HTML documentation website for you.

Why JSDoc:

- JSDoc is primarily used for generating documentation from inline comments in JavaScript code. It allows developers to describe the purpose, functionality, and usage of functions, variables, and classes.
- JSDoc enhances code readability by providing a standardized way to document code elements. Clear and consistent documentation makes it easier for developers to understand the purpose and usage of different parts of the code.

Getting started with JSDoc:

Here is a detailed instruction for generating documentation using JSDoc for javascript files. Go to the link below-

[JSDoc video](#)

JSDoc Documentation Example:

```
JS index.js > PrintDetails
1  /**
2   * This function is about adding a new product and setting the price of the product.
3   * @param {string} id The id of the product
4   * @param {string} name The name of the product
5   * @param {number} price The price of the product in $.
6   * @returns {object} The product description.
7   */
8
9  function createItem(id, name, price) {
10
11     return {
12         id: id,
13         name: name,
14         price: price
15     }
16 }
17 class Product {
18     /**
19     * creates a product.
20     * @param {string} id The id of the product
21     * @param {string} name The name of the product
22     * @param {number} price The price of the product in $.
23     */
24 }
```

Html Documentation:

Class: Product

Product(id, name, price)

new Product(id, name, price)

creates a product.

Parameters:

Name	Type	Description
id	string	The id of the product
name	string	The name of the product
price	number	The price of the product in \$.

Source: [index.js, line 25](#)

Home

Classes

Product

Global

PrintDetails

createItem

Advantages:

- **Improved Code Readability:** JSDoc enhances code readability by providing a standardized way to document code elements. This makes it easier for developers, especially when working on collaborative projects or maintaining code over time.
- **Automated Documentation:** The ability to generate documentation automatically from JSDoc comments saves time and ensures that documentation stays up-to-date. It helps in maintaining consistency between code and documentation.
- **Enhanced IDE Support:** JSDoc comments improve the developer experience by enabling powerful features in IDEs, such as intelligent auto completion, type checking, and quick documentation lookup. This contributes to a more efficient coding process.
- **Facilitates Type Annotations:** JSDoc supports type annotations, allowing developers to specify the expected types of parameters and return values. This helps in catching potential type-related errors during development and provides better insights for tooling.

Disadvantages:

- **Maintenance Overhead:** While JSDoc facilitates automated documentation, there is a risk of documentation becoming outdated or inaccurate if not properly maintained. Developers need to be diligent in updating comments to reflect any changes in the code.
- **Learning Curve:** For developers new to JSDoc or documentation practices in general, there might be a learning curve. Understanding the syntax and conventions of JSDoc comments may require additional effort initially.
- **Redundancy:** In some cases, the information in JSDoc comments may duplicate what is evident from the code itself. This redundancy can lead to extra maintenance work and potential inconsistencies if not managed carefully.