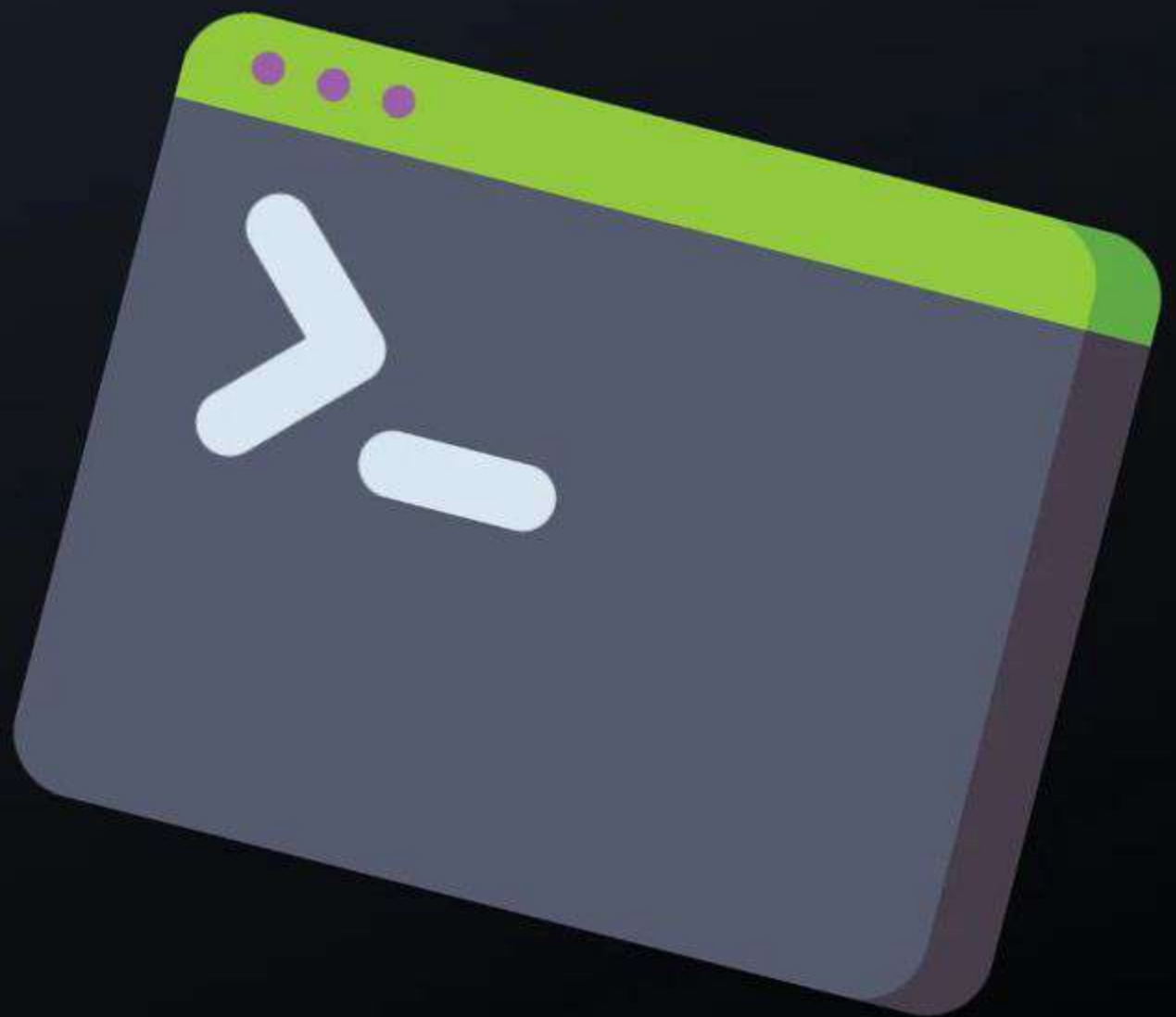


Common JavaScript Console Methods



Ariba M.
@frontendcharm



In this post, we will discuss the different JavaScript console methods. These console methods are very useful when you build a web application.

1. `console.log()`

We are all familiar with this console method. Right?

It gives the output message in the console. The

`console.log()` is a function in JavaScript which is used to print any kind of variables defined before in it or to just print any message that needs to be displayed to the user.



JS script.js

```
1 console.log("Hello World!");
```




2. console.warn()

This console method is used to show the warning to the user in the console.



JS script.js

```
1 console.warn("Warning Message");
```

3. console.error()

This console method is used to show the error message in the console.



JS script.js

```
1 console.error("Error Message");
```



4. console.table()

This console method is used to show the object in the form of a table. If you working with big objects, don't use `console.log()`, use `console.table()`.



JS script.js

```
1 console.table({name:"frontendcharm", age:22});
```

5. console.assert()

This console method is used to show the error if your given condition is false.



JS script.js

```
1 console.assert(10 > 18, "Error Message");
```




6. `console.time()` with `console.timeEnd()`

The `console.time()` method starts a timer you can use to track how long an operation takes. The `console.timeEnd()` stops a timer that was previously started by calling `console.time()`.



JS script.js

```
1 console.time();  
2 for (let i = 0; i < 1000000; i++) {  
3     // some code  
4 }  
5 console.timeEnd();
```



7. console.info()

This console method is used to show the informational message to the Web console.

 JS script.js

```
1 const myObj = {firstname:"John", lastname:"Doe"};  
2 console.info(myObj);
```

8. console.trace()

This console method is used to show trace that shows how the program ended up at a certain point.

 JS script.js

```
1 function myFunction1() {  
2   myFunction2();  
3 }  
4  
5 function myFunction2() {  
6   console.trace();  
7 }
```




9. console.count()

The `console.count()` method logs the number of times that this particular call to `count()` has been called.

 JS script.js

```
1 for (let i = 0; i < 5; i++) {  
2   console.count("myLabel");  
3 }
```

10. console.clear()

The `console.clear()` method clears the console if the environment allows it.

 JS script.js

```
1 console.clear();
```