**How Does JavaScript Handle the Promises -** [Datt Panchal](https://medium.com/@dattpanchal2904?source=post_page-----8021482db8e5--------------------------------) Aug 9, 2022

Understanding Promises in Modern JavaScript

**Introduction**

In this article, we will learn about the promises in JavaScript. A promise is one of the most important concepts in Modern JavaScript. It isused to handle Asynchronous operations in JavaScript. We will see the below points:

* What are promises?
* Promises objects
* Benefits to use promises
* Syntax of promise
* One simple example with explanation and output
* References
* Summary

So, let’s move on to our main point without further time delay!

**What are promises in JavaScript?**

In simple words, Promise is the one object of JavaScript that will be resolved or rejected. Now, what is meant by resolve or reject? Promises either resolve means the given condition is fulfilled or promises reject when the given condition is not fulfilled.

If the condition will be fulfilled, the promise will resolve. But if the given condition is not fulfilled then it is rejected due to some reasons. Here we will apply the concept of Asynchronous programming,

**Promise objects in JavaScript**

* Pending: It means undefined. The promise is not resolved and is rejected.
* Resolve: It resolves when the given condition is fulfilled.
* Reject: It rejects when the given condition is not fulfilled.

**Benefits to use promises**

* It makes our code more readable and understandable
* To use promises, we can control the errors easily, especially in large web applications
* Easy to control the flow of the code with Asynchronous programming
* It does not block the flow of the code

**Syntax of the promise**

let myPromise = new Promise(function(resolve, reject){  
 // function code  
});

**Understanding Promises with a example**

Talk is cheap, let’s show you the code : )





**How does this code work?**

* We created one function that returns the promise. Inside the function, we give a new promise that returns one function with two arguments. One is ***resolve***and the second one is ***reject***. And then inside that function, we set time delays ( 2-second ) to work it Asynchronously.
* Inside the ***setTimeout()***function, we set the ***error = true***, you can also try with ***error = false***. After that, we applied the conditions that is.

***Condition:*** If the error does not occur ( Which means**error = false** ), then resolve the promise with the***resolve()***function. And if the error occurs ( Which means **error = true**), then reject the promise with ***reject().***

* Then we called the function. Here***then()*** function works/acts as a resolve and ***catch()***function works/acts as a reject. If the promise is resolved then it executes the code which is inside the **then()**, if the promise is rejected, then it executes the code which is inside the ***catch().*** The program is over here, let’s see the output now.

**The final output of the code**

Text

Description automatically generated

The output of resolved promise!

Graphical user interface, text, application

Description automatically generated

The output of rejected promise!

So, this is how you can handle the promises. I hope you like and understand the concept of promises in this article. Now, I’m sharing the resources/references where you can know more about the promises in JavaScript.

**References**

* [geeksforgeeks](https://www.geeksforgeeks.org/javascript-promises/)
* [w3schools](https://www.w3schools.com/js/js_promise.asp)
* [developer.mozilla.org](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise)
* [freeCodeCamp](https://www.freecodecamp.org/news/javascript-promise-tutorial-how-to-resolve-or-reject-promises-in-js/)
* [programiz](https://www.programiz.com/javascript/promise)

**Quick Summary**

In this article, we learned about the Promises in JavaScript. We covered,

* What are promises?
* Promises objects
* Benefits to use promises
* Syntax of promise
* One simple example with explanation and output
* References

And at last, I’ve shared some references where you can easily learn more about promises. I hope this article will helpful to you.

**Want to connect?**

Add me to your [***LinkedIn profile***](https://www.linkedin.com/in/dattpanchal04/)and [***Twitter***](https://twitter.com/dattpanchal2904)***.***

Dear, Thank you for reading this article. I hope you enjoyed it. Please share this article with your friends or relatives who might find it useful. You may be interested in articles about new technologies or programming or electronics or the digital world or reviews, [***Follow me and share this profile***](https://medium.com/@dattpanchal2904) with your circle and you can [***subscribe to my email***](https://medium.com/subscribe/@dattpanchal2904) to receive the notification immediately whenever I published a new article.