Caltech | Center for Technology & Management Education

Full Stack Java Developer

simplilearn

JavaScript

Async Programming

Learning Objectives

By the end of this lesson, you will be able to:

- Define promises in JavaScript
- Discuss producing code and consuming code
- List the stages in promises
- Define async functions and future

A Day in the Life of a Full Stack Developer

You are working in an organization and have been assigned a web development project. You are facing an issue in the application where long operations in programs like uploading files or fetching records cause the server to freeze.

You decide to make use of promises to solve this issue. To do so, explore more about promises, stages of promises, and async functions.

TECHNOLOGY

Promises and Async Function

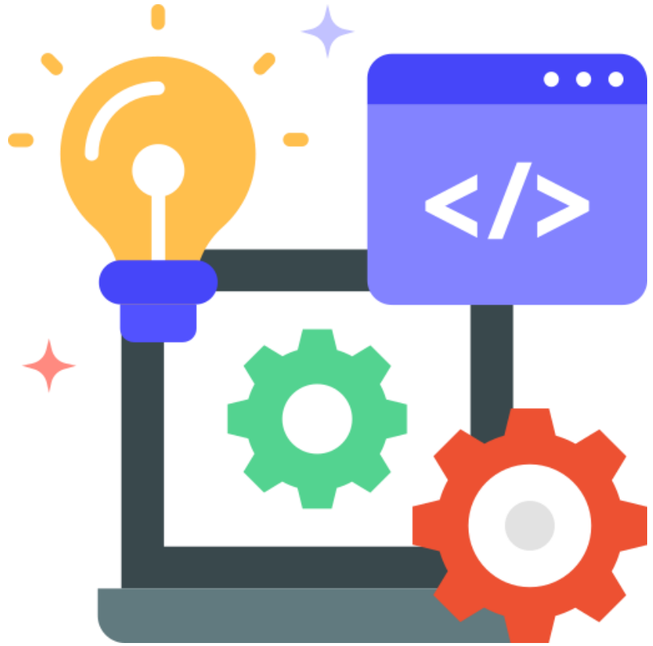
Asynchronous Programming

In asynchronous programming, functions are performed independently of the main program flow.

In consumer computers, programs run for a limited time before giving way to other programs.

Asynchronous Programming

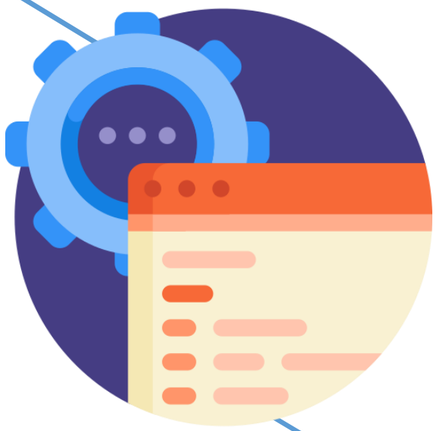
Long operations in programs like uploading files or fetching records can cause freezing.



Promises

Producing code and consuming code are two important terms in promises.

Producing code refers to the piece of code that takes a long time to fetch results.

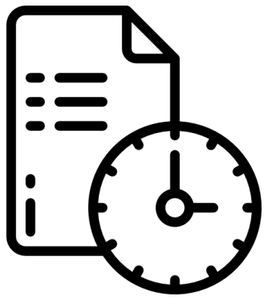


Consuming code refers to the code that waits for the Producing Code to return the result.

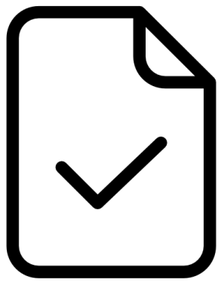
Promises

A promise is a JavaScript object that helps link producing code and consuming code.

It has three stages:



Pending



Fulfilled

Rejected

Promises

```
let myPromise = new promise (function (myResolve,
myReject) {
// producing code
myResolve ();  // successful
myReject();  // error
});
// consuming code
myPromise.then (
    function(value) {   },
    Function (error) {   }
);
```

Promises

When the producing code receives the result, one of the two call-backs should be invoked.

Success

myResolve(result value)

Error

myReject(error object)

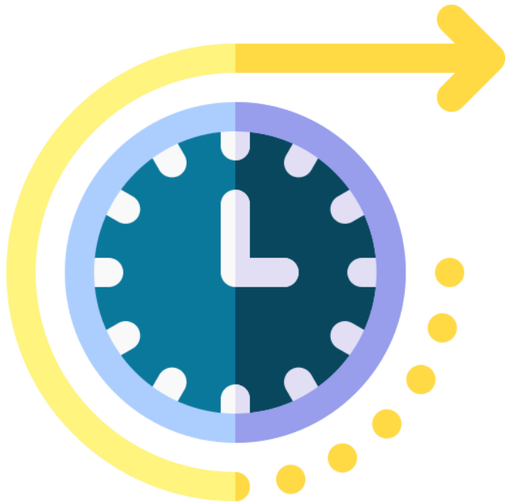
Async Function

The Async keyword can be used with a function to make it asynchronous and return a promise object.

The async or wait keyword makes the function asynchronous.

Future

Future is a read-only container for an undefined consequence, whereas a promise can be written.



Future

Promise

Key Takeaways

- In asynchronous programming, functions are performed independently of the main program flow.
- A promise is a JavaScript object that helps link producing code and consuming code.
- The Async keyword can be used with a function to make it asynchronous and return a promise object.
- Future is a read-only container for an undefined consequence, whereas a promise can be written.

TECHNOLOGY

Thank You

simplilearn