**Synchronous and asynchronous communication**

**Synchronous communication**

It will execute the set of statement sequentially or one by one

console.log(“1st statement”);

console.log(“2nd statement”);

console.log(“3rd statement”);

synchronous function call

fun1();

fun2()

fun3();

**Asynchronous communication**: it will execute set of statemen independently.

console.log(“1st statement”);

console.log(“2nd statement”); asynchronous

console.log(“3rd statement”);

asynchronous function call

fun1();

fun2()

fun3();

if we want to make asynchronous statement execution on client machine. JavaScript provided 3 functions

setTimeout()

setInterval()

clearInterval()

1st req to search some information from database

2nd req get some dynamic code or random code

3rd req simple static message

1 passing emailid and password if correct open dashboard and display the message.

2 open the dashboard once we get confirmation

Client Server

AJAX :

Asynchronous JavaScript and XML (eXtensible markup of language)

XMLHttpRequest() :

ActiveXObject() : these pre defined object allow to achieve core Ajax features.

**Promise** : Promise is a pre defined object which help to handle asynchronous event of data of any type.

Promise can be resolved or rejected.

To handle the promise we need to use then() and catch(). If promise resolve then will call to load the promise data if promise rejected catch() call to handle the error.

**Web Service :** Giving the service for web application when both the application running using different technologies or languages.

Amazon ----🡪 React or Angular or Python or Java

Payment ----🡪 Paypal

Paytm Java

Gpay Python

Phone pay .net

Credit card php

Debit card

Net banking

XML / JSON

XML and JSON is common format we use in industry to share the data.

JavaScript provided pre defined function is **fetch().** This function is use to consume rest api or web service develop using any language.

**fetch(“url”).** Return type of fetch is promise object.

ES5 style handling asynchronous evet of data

fetch(“url”).then(result=>console.log(result)).catch(error=>console.log(error));

**ES6**

Then replace by await and async keyword.