1. Java script is single thread , it means it is non blocking in nature and will work on asynchronous action
2. Promises are ideal choice for handling asynchronous operation , they are used to provide better error handling than call back and events
3. Await function is used to wait for the promise. It could be used within the async block only. It makes the code wait until the promise returns a result. It only makes the async block wait.
4. Using error handling method
5. Returns a promise
6. ,then() consumes a value in promise return
7. Return promise ((resolve, reject)=>set Timeout(( )=>resolve( ),100));
8. Pending , resolve and reject
9. Fails to even change the state from pending to anything else.
10. Only the first promise will be executed
11. The finally statement lets you execute code, after try and catch, regardless of the result.
12. Micro-task is said to be a function which is executed after the function or program which created it exits
13. function print(callback) {

Callback (3000 ) ; }

1. new delay(function(resolve, reject){

set timeout(() =>resolve(“should alert after sec),3000);})

.then (function (delay){

alert (result);})

1. async function f(){

let promise = new Promise((res,rej) => {

   set Timeout(() => res("Resolved!"),3000)

} );

let result = await promise;

alert("Response value is: " + res));

}

1. The difference is

promise .then (f1).catch(f2);

If the error happens by f1 then its handled by .catch

But in

promise .then(f1, f2);

even if there is error in any one of the functions the execution stops and the debugging is difficult