

Promise has 3 stages

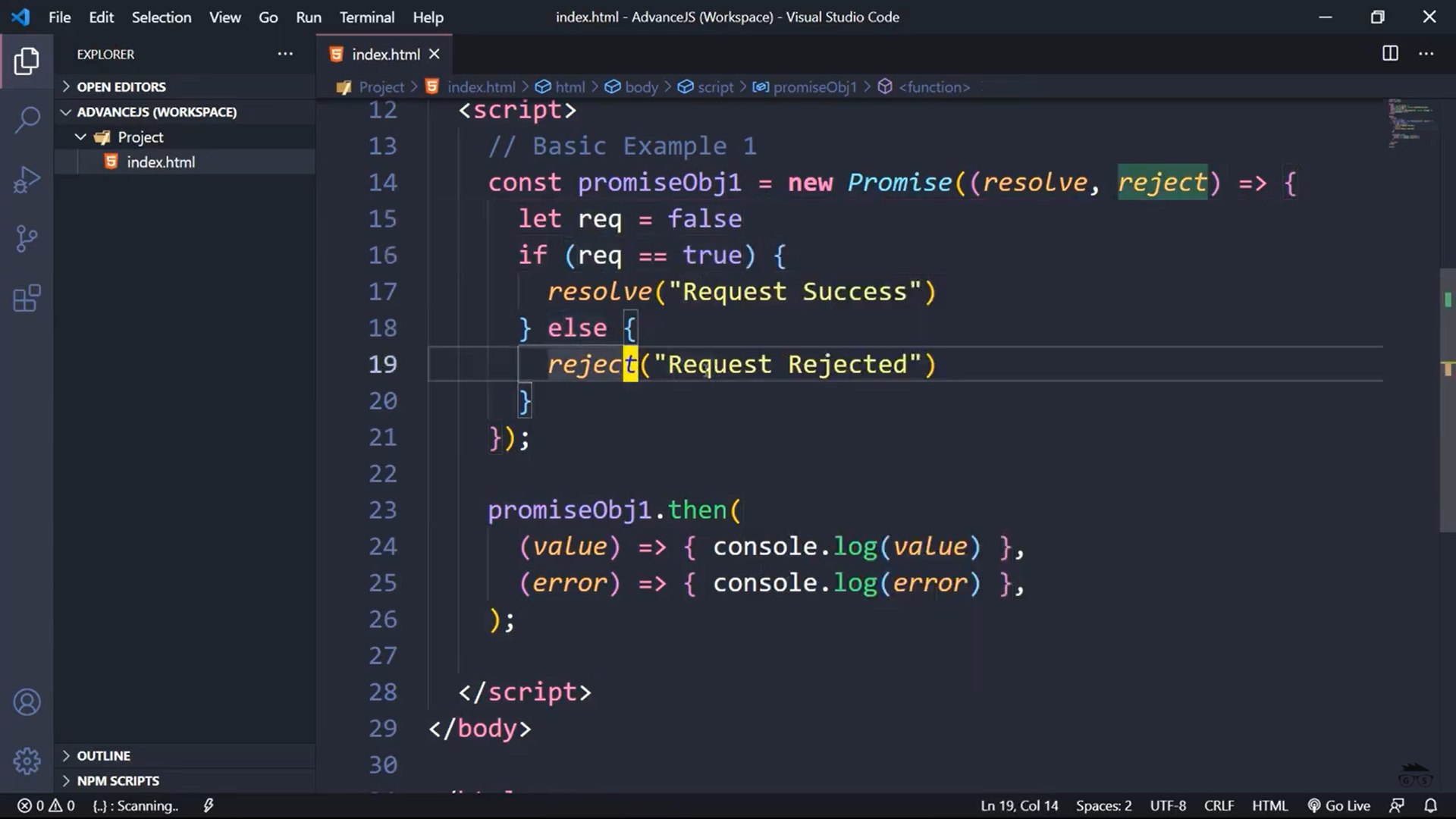
When we start a task. Its in 3 different states

**LIKE THERE IS A TASK : Like you call me at 10 am tomorrow**

So, before 10 am , its is **PENDING STATE (before event)**

**AT 10 a.m. , it might be either in success (if you call or it might be in fail state, if you don’t call)**

**RESOLVE : WHEN IT IS FULLFIILED  
REJECT : WHEN YOU DON’T CALL**



function func1() {

return new Promise(function (resolve, reject) {

setTimeout(() => {

const error = true;

if (!error) {

console.log("Your promise has been fulfilled");

resolve();

}

else {

console.log("Promise not fulfilled");

reject("Sorry not fulfilled");

}

}, 2000);

});

}

func1().then(function () {

console.log("Thanks for resolving");

});

function func1() {

return new Promise(function (resolve, reject) {

setTimeout(() => {

const error = true;

if (!error) {

console.log("Your promise has been fulfilled");

resolve();

}

else {

console.log("Promise not fulfilled");

reject("Sorry not fulfilled");

}

}, 2000);

});

}

func1().then(function () {

console.log("Thanks for resolving");

}).catch(function () {

console.log("Very Bad");

});

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

<title></title>

<script>

const pObjt1 = new Promise((resolve, reject) => {

setTimeout(() => {

let rollNo = [1, 2, 3, 4, 5];

resolve(rollNo);

reject('Error while communication');

},2000);

});

//promise consume

pObjt1.then((rollno) => {

console.log(rollno);

});

</script>

</head>

<body>

Working with Variables

</body>

</html>