# JavaScript Promises - Easy Practice Solutions

Here are the solutions to the easy-level JavaScript Promise questions. These examples demonstrate how to use .then(), .catch(), and async/await for handling asynchronous operations.

* 1. Promise that resolves with 'Hello, World!' after 1 second (using .then):

const promise = new Promise((resolve) => {  
 setTimeout(() => resolve("Hello, World!"), 1000);  
});  
  
promise.then((msg) => console.log(msg));

* 2. Promise that rejects with 'Something went wrong' after 2 seconds (using .catch):

const promise = new Promise((\_, reject) => {  
 setTimeout(() => reject("Something went wrong"), 2000);  
});  
  
promise.catch((err) => console.error(err));

* 3. delayedMessage() that resolves after 3 seconds:

function delayedMessage() {  
 return new Promise((resolve) => {  
 setTimeout(() => resolve("Done!"), 3000);  
 });  
}  
  
delayedMessage().then((msg) => console.log(msg));

* 4. failRandomly() that sometimes resolves or rejects:

function failRandomly() {  
 return new Promise((resolve, reject) => {  
 if (Math.random() > 0.5) {  
 resolve("Success");  
 } else {  
 reject("Failed");  
 }  
 });  
}  
  
failRandomly()  
 .then((msg) => console.log(msg))  
 .catch((err) => console.error(err));

* 5. Convert Question 1 to async/await:

async function helloWorld() {  
 const msg = await new Promise((resolve) => {  
 setTimeout(() => resolve("Hello, World!"), 1000);  
 });  
 console.log(msg);  
}  
  
helloWorld();

* 6. Convert Question 2 to async/await with try/catch:

async function errorExample() {  
 try {  
 const msg = await new Promise((\_, reject) => {  
 setTimeout(() => reject("Something went wrong"), 2000);  
 });  
 console.log(msg);  
 } catch (err) {  
 console.error(err);  
 }  
}  
  
errorExample();

* 7. Async function getNumber():

async function getNumber() {  
 const num = await new Promise((resolve) => {  
 setTimeout(() => resolve(42), 2000);  
 });  
 console.log(num);  
}  
  
getNumber();

* 8. Async function catching rejection:

async function rejectExample() {  
 try {  
 const result = await new Promise((\_, reject) => {  
 setTimeout(() => reject("Invalid Request"), 1000);  
 });  
 console.log(result);  
 } catch (err) {  
 console.error("Caught Error:", err);  
 }  
}  
  
rejectExample();

* 9. Fetch random dog image using .then():

fetch("https://dog.ceo/api/breeds/image/random")  
 .then((res) => res.json())  
 .then((data) => console.log(data.message))  
 .catch((err) => console.error(err));

* 10. Fetch GitHub users with async/await:

async function fetchGitHubUsers() {  
 try {  
 const res = await fetch("https://api.github.com/users");  
 const data = await res.json();  
 console.log(data[0].login);  
 } catch (err) {  
 console.error(err);  
 }  
}  
  
fetchGitHubUsers();