**¿Qué es try...catch?**

try...catch es una estructura de control de flujo en JavaScript que se utiliza para manejar errores de manera controlada. Esto significa que puedes ejecutar código dentro del bloque try y, si se produce un error, el código dentro del bloque catch se ejecutará.

**¿Cómo se usa try...catch?**

La sintaxis básica de try...catch es la siguiente:

JavaScript

try { // Código que puede generar errores } catch (error) { // Código que se ejecutará si se produce un error console.log(error); }

Use code [with caution.](file:///C:/faq#coding)

**Ejemplo:**

JavaScript

function divideNumbers(a, b) { try { if (b === 0) { throw new Error("No se puede dividir por cero"); } return a / b; } catch (error) { console.log(error.message); return 0; } } console.log(divideNumbers(10, 2)); // Output: 5console.log(divideNumbers(10, 0)); // Output: No se puede dividir por cero

Use code [with caution.](file:///C:/faq#coding)

En este ejemplo, la función divideNumbers intenta dividir dos números. Si el divisor es cero, se lanza una excepción utilizando la palabra clave throw. La excepción es capturada en el bloque catch y se muestra un mensaje de error.

**Puntos importantes a recordar:**

* Puedes usar cualquier tipo de excepción que desees lanzar en el bloque try.
* El bloque catch es opcional, pero es recomendable usarlo para manejar los errores de manera adecuada.
* Puedes usar la palabra clave finally para ejecutar código independientemente de si se produce un error o no.

**Ejemplo con finally:**

JavaScript

function divideNumbers(a, b) { try { if (b === 0) { throw new Error("No se puede dividir por cero"); } return a / b; } catch (error) { console.log(error.message); return 0; } finally { console.log("Operación finalizada"); } } console.log(divideNumbers(10, 2)); // Output: 5console.log(divideNumbers(10, 0)); // Output: No se puede dividir por cero

Use code [with caution.](file:///C:/faq#coding)

En este ejemplo, el bloque finally se ejecuta siempre, independientemente de si se produce un error o no.