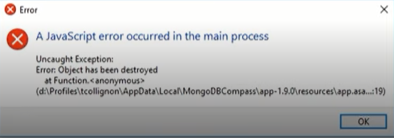
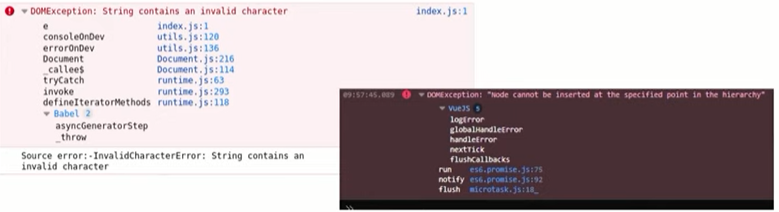
Este curso visa reconhecer erros e fazer erros personalizados.

Existem varios tipos de erros, o **ECMAScript error** é um erro que ocorre no tempo de execução, é o erro causado geralmente por fomatação ou erro de falta de variável, etc.

O **DOMException** é um erro no DOM, no document object model.

o tipo de erro é relacionado ao local em que ele ocorre.

Tratamos erros com os comandos **throw, try/catch** e **finally**, elas são as principais saidas para lidar com erros.

**Throw** é utilizada dentro de uma curta if a fim de identificar uma situação de input invalida para **jogar** uma outra instrução de identificação de erro no lugar da instrução padrão.

*Function exemplo(string){*

*if(!string) throw “string invalida”;(resto das instruções)}*

Basicamente substitui os if returns que temos utilizados, a grande diferença é que o throw **permite que o console identifique e trate esta situação como um erro**.

O **try catch** é uma declaração que coloca um código try que, se acionado por causa de uma input joga a instrução catch, **a intenção do catch é personalizar o erro em diferentes blocos**.

O **finally** é uma instrução extra ao bloco try catch que é chamada idenpendente ao erro, ou seja, seja o erro acionado ou não, o finally sera acionado geralmente para mostrar se foi um sucesso ou não.

O objeto **error** é a principal maneira de personalização do erro, ele aceita três parâmetros, **message**, **fileName** e **lineNumber** (sendo o fileName e lineNumber nem sempre sendo compatíveis, sendo message o usado mais comumente devido a isso), com ele cria-se um nomo objeto filho ao Error que manda uma instrução específica caso a mensagem seja acionada por um catch ou throw.

*New Error(message,fileName,lineNumber);*

*Const meuErro= new Error(‘mensagem invalida’);*

*Throw meuErro;*

O erro também pode ter um **nome**, esse nome sera exposto quando o erro for lançado.

*Const MeuErro=new Error(‘mensagem invalida’);* o melhor jeito de lembrar isso

*MeuErro.name=’invalid message’;* é lembrando do error 404 (nome):

*Throw MeuErro;* page not found. (mensagem)

O subcomando **stack** permite mostrar onde ocorreu o erro, geralmente em x:y, sendo x **linha** e y **coluna**.

