

Try & Catch

O try e catch é utilizado quando há possibilidade de ocorrer um erro, o try captura esse erro e o catch realiza o tratamento, impedindo assim que o programa pare.

Podemos também utilizar o `throw` para definir um novo tipo de erro conforme exemplo a baixo:

```
throw new Error ('Jarvis falhou');
```

Para capturar um erro e fazer seu tratamento utiliza-se a seguinte estrutura de código:

```
try{
  //Linha de código
  //Linha de código
  //Linha de código
} catch(error){
  alert('ocorreu o seguinte erro ' + error);
}
```

O bloco try tenta executar as linhas de código, caso qualquer erro aconteça, esse erro vai ser armazenado no parâmetro passado no catch e pode ser utilizado no tratamento do erro. O bloco catch por sua vez só é executado caso o erro ocorra e nele que será feito o tratamento do erro.

Problema

Pedir para o usuário digitar dois números que representem os lados de um retângulo. Mostrar na tela a área total. Esse código deve ser executado quando a página é aberta e também ao clicar em um botão. O que acontece se o usuário passar um dado inválido ?

Resolução:

```
iniciar();

function iniciar(){
  var lado1 = parseFloat(prompt('Informe a altura do retangulo:'));
  var lado2 = parseFloat(prompt('Digite a largura do retangulo:'));

  try{
    var area = calcularArea(lado1, lado2);
  } catch(erro){
    alert(erro);
  }

  escrever(area);
}

function calcularArea(l1, l2){

  if(isNaN(l1) || isNaN(l2)){
    throw new Error ('O parâmetro passado é inválido.');
```

```
  }
```

```
  return l1 * l2;
```

```
}
```

```
function escrever(text){
```

```
  document.querySelector('.mostrador').innerHTML = text;
```

```
}
```