



Aula 16

Prof: Henrique Augusto Maltauro

Codificar Back-end de Aplicações Web

Tratamento de Exceções

- C#: `ArgumentNullException`

O `ArgumentNullException` é uma classe filha da classe `ArgumentException` que representa qualquer exceção gerada quando um valor nulo é passado a um método que não o aceita como um parâmetro válido.

Podemos passar uma string como parâmetro, a qual irá definir uma mensagem de erro personalizada.

Tratamento de Exceções

- C#: `ArgumentNullException`

```
try
{
    throw new ArgumentNullException();
}
catch (ArgumentNullException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `ArgumentNullException`

```
try
{
    throw new ArgumentNullException("Mensagem de erro");
}
catch (ArgumentNullException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `ArgumentOutOfRangeException`

O `ArgumentOutOfRangeException` é uma `classe` filha da `classe` `ArgumentException` que representa qualquer exceção gerada quando o valor de um `parâmetro` está fora do intervalo permitido de valores.

Podemos passar uma `string` como `parâmetro`, a qual irá definir uma mensagem de erro personalizada.

Tratamento de Exceções

- C#: `ArgumentOutOfRangeException`

```
try
{
    throw new ArgumentOutOfRangeException();
}
catch (ArgumentOutOfRangeException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `ArgumentOutOfRangeException`

```
try
{
    throw new ArgumentOutOfRangeException("Mensagem de erro");
}
catch (ArgumentOutOfRangeException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `ArithmeticException`

O `ArithmeticException` é uma `classe` filha da `classe` `SystemException` que representa qualquer exceção gerada por erros em uma operação aritmética, projeção ou conversão.

Podemos passar uma `string` como `parâmetro`, a qual irá definir uma mensagem de erro personalizada.

Tratamento de Exceções

- C#: `ArithmeticException`

```
try
{
    throw new ArithmeticException();
}
catch (ArithmeticException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `ArithmeticException`

```
try
{
    throw new ArithmeticException("Mensagem de erro");
}
catch (ArithmeticException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `DivideByZeroException`

O `DivideByZeroException` é uma `classe` filha da classe `ArithmeticException` que representa qualquer exceção gerada quando há uma tentativa de dividir um valor numérico por zero.

Podemos passar uma `string` como `parâmetro`, a qual irá definir uma mensagem de erro personalizada.

Tratamento de Exceções

- C#: `DivideByZeroException`

```
try
{
    throw new DivideByZeroException();
}
catch (DivideByZeroException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `DivideByZeroException`

```
try
{
    throw new DivideByZeroException("Mensagem de erro");
}
catch (DivideByZeroException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `OverflowException`

O `OverflowException` é uma classe filha da classe `ArithmeticException` que representa qualquer exceção gerada quando uma operação aritmética, projeção ou conversão resulta em um estouro do limite de valores.

Podemos passar uma `string` como `parâmetro`, a qual irá definir uma mensagem de erro personalizada.

Tratamento de Exceções

- C#: `OverflowException`

```
try
{
    throw new OverflowException();
}
catch (OverflowException ex)
{
    return ex.Message;
}
```

Tratamento de Exceções

- C#: `OverflowException`

```
try
{
    throw new OverflowException("Mensagem de erro");
}
catch (OverflowException ex)
{
    return ex.Message;
}
```


Exercício

Exercício

gg.gg/BackEnd16

github.com/hmaltaurodev/slides