**JavaScript**

- window.alert(‘ ... ’) – para adicionar um alerta à página

- window.confirm(‘ ... ‘) – para adicionar uma caixa de confirmação

- window.prompt(‘ ... ‘) – para adicionar uma caixa de pergunta

- // - para inserir um comentário em uma única linha

- /\* \*/ - para inserir comentários em múltiplas linhas

- o sinal de **=** em JavaScript significa “recebe”

- null significa nulo/sem nada

- var é usado para criar uma variável que seria como uma vaga

- identificadores podem começar com letra, $ ou \_. Não podem começar com números. É possível usar letras com números. É possível usar acentos e símbolos. Não podem conter espaços. Não podem se palavras reservadas como “alert” ou “var”

- Ctrl + l – ele vai limpar

- o + tem efeito de adição e de concatenação. Number + number para adição, string + string para concatenação

- Number.parseInt(n) converte para número inteiro, exemplo 10

- Number.parseFloat(n) converte para número real, exemplo 10.4

- se usar apenas Number também converte para inteiro ou real

- String(n) converte para string

- n.toString( ) – converte para String

Exemplos de concatenação

> nome = 'Luan'

'Luan'

> idade = '41'

'41'

> nota = 5.5

5.5

> nome

'Luan'

> idade

'41'

> nota

5.5

Exemplo 1 > 'O aluno ' + nome + ' com ' + idade + ' anos tirou a nota ' + nota

'O aluno Luan com 41 anos tirou a nota 5.5'

Exemplo 2 > `O aluno ${nome} com ${idade} anos tirou a nota ${nota}`

'O aluno Luan com 41 anos tirou a nota 5.5'

- Adicionando valor monetário no NODE:

> var n1 = 1545.5

undefined

> n1

1545.5

> n1.toFixed(2)

'1545.50'

> n1.toFixed(2).replace('.', ',')

'1545,50'

> n1.toLocaleString('pt-BR', {style: 'currency', currency: 'BRL'})

'R$ 1.545,50'

> n1.toLocaleString('pt-BR', {style: 'currency', currency: 'USD'})

'US$ 1.545,50'

> n1.toLocaleString('pt-BR', {style: 'currency', currency: 'EUR'})

'€ 1.545,50'

Operadores aritméticos

* 5 + 2 = 10 | 5 – 2 = 3 | 5 \* 2 = 10 | 5 / 2 = 2.5 | 5 % 2 = 1 (resto da divisão) | 5 \*\* 2 = 25 (potenciação)
* n = n + 4 > n + = 4
* n = n – 5 > n - = 5
* n = n \* 4 > n \* = 4
* n = n / 2 > n / = 2
* n = n \*\* 2 > n \*\* = 2
* n = n % 5 > n % = 5