

Básico do JS

segunda-feira, 4 de maio de 2020 10:27

- **Variáveis:** não é necessário especificar tipo de variável;
 - Tipagem dinâmica A mesma variável pode assumir outros tipos;
 - Variáveis de mesmo tipo: coloca-se na mesma linha;
 - Operações com variáveis:
 - Para operações com ela mesma ($X = X + 3$), usa-se ($X += 3$);
- Criar objetos em variáveis:
 - ```
var aluno = {
 nome: 'Laura';
 idade: 19;
}
```
- O ponto: utiliza-se para acessar propriedades do elemento:
  - `return(aluno.nome)`
- **Funções:**
  - Para criar:

```
function nome(parâmetros){
 conteúdo da function/conteúdo para execução
}
```
  - Ao criar uma variável na function, ela só será visível lá, caso seja utilizada fora, é preciso defini-la novamente.
  - Return: retorna a variável.
- **Condicionais:**
  - If: if (condição) {  
 execução  
} else {  
 execução  
}
  - Condições de igualdade:
    - (`var = 'X'`): define a variável como 'x';
    - (`var == 'X'`): verifica se o conteúdo de var é igual ao 'X';
    - (`1 === '1'`): verifica igualdade entre o valor e o tipo da variável.
- **Operadores lógicos:**
  - And: `&&`
  - Or: `||`
  - Not: `!=` ou `!==`, verifica desigualdade.
  - O javascript sempre retorna boolean, então pode-se fazer:
    - ```
var sexo = 'M';  
var masc = (sexo === 'M');
```

 isso retornará boolean.
- **Condição ternária:**
 - Condição que possui apenas duas verificações dentro do if.
 - ```
var retorno = (sexo === 'M') ? 'M' : 'F';
```
    - `?` : o resultado entre as variáveis será X ou Y;
    - Dois pontos: condição que satisfaz a sentença e a que não satisfaz.
- **Laços de repetição:**
  - For: usar quando saber o tamanho do intervalo;
    - ```
for (var i=0 ; i<=100 ; i++)  
  i++: contador (somar i a ele mesmo);
```
 - while: usar quando não temos o controle de quantas interações a variável fará;
 - ```
while (j <= 100);
j++;
```