Roteiro Didático de Aula
Aula 02 – JavaScript
22/09/2020

Assuntos que serão abordados:

Estrutura Condicional

Usando as instruções if .. else

Laços de Repetição

For

While

Do..While

Referências

Estrutura Condicional

Estruturas Condicionais em JavaScript

Estruturas condicionais são instruções as quais podem mudar o fluxo de execução do script. A partir delas pode ser escolhido o bloco de execução a ser executado.

Para criarmos uma estrutura condicional, podemos usar o if .. else ou o switch.

Usando as instruções if .. else

Com as instruções if .. else, é muito simples criar uma estrutura de condição. Traduzidas para o português significam se .. senao, respectivamente. A partir disto, podemos fazer uma analogia com a lingua portuguesa da seguinte maneira:

```
se (condicao) {
        executar operacao
} senao {
        executa outra operacao
}
```

Traduzindo para JavaScript

```
<script>
if (condicao) {
        executar operacao
} else {
        executa outra operacao
```


Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

```
}
</script>
```

EXERCICIO 1:

Desenvolver uma solução JavaScript que defina que ao chover é necessário tirar as roupas do varal

Obs:

As chaves determinam o início e o fim do bloco de script a ser executado.
O else não é obrigatório, sendo necessário apenas aonde é necessária duas estruturas de condições.
O else SÓ pode ser usado quando NA MESMA INSTRUÇÃO está sendo usado o if.
Operadores lógicos condicionais

Você pode usar os seguintes operadores logicos para as instruções if .. else:

> : Maior

```
<script>
var a = 6;
if (a > 6) {
        document.write('Maior que 6');
} else {
        document.write('Menor que 6');
}
</script>
```

< : Menor

```
<script>
var a = 6;
if (a < 6) {
        document.write('Menor que 6');
} else {
        if (a > 6) {
                document.write('Maior que 6');
        } else {
                document.write('Igual a 6!');
        }
}
</script>
```

EXERCICIO 2:

Desenvolver uma solução JavaScript que defina que pessoas com menos de 18 anos não podem comprar bebida alcoolica

Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

== : Igual

```
<script>
var a = 6;
if (a == 6) {
        document.write('Igual a 6');
}
</script>
```

>= : Maior ou igual

```
<script>
var a = 6;
if (a >= 6) {
        document.write('Maior ou igual a 6');
}
</script>
```

<= : Menor ou igual

```
<script>
var a = 6;
if (a <= 6) {
        document.write('Menor ou igual a 6');
}
</script>
```

!= : Diferente

```
<script>
var a = 6;
if (a != 6) {
        document.write('Diferente de 6');
}
</script>
```

EXERCICIO 3:

Desenvolver uma solução JavaScript que compare a idade e altura mínima para uma pessoa ingressar em determinada montanha russa.

Para o uso de uma instrução if .. else com várias variáveis, você pode usar os seguintes operadores lógicos:

&& : E

```
<script>
var a = 6
if ((a > 1) && (a < 6)) {
        document.write('Maior que 1 E menor que 6');
}
```


Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

```
</script>
```

Para o uso de estruturas de condição com várias variáveis é necessário de um parêntese cobrindo todo o conjunto de estruturas de condição:

```
se ( (condicao1) E (condicao2) ) {
        executar operacao
}
```

|| : OU

```
<script>
var a = 6
if ((a > 1) || (a < 6)) {
        document.write('Maior que 1 OU menor que 6');
}
</script>
```

! : NAO

Para indicar negação nas operações é necessário usar o ponto-de-exclamação antes de toda a operação.

```
<script>
var a = 6
if !((a > 1) || (a < 6)) {
        document.write('Não é maior que 1 OU menor que 6');
}
</script>
```

EXERCICIO 4:

Desenvolver uma solução JavaScript que compare a quantidade de pontos na carteira de motorista de alguém e retorne se ela pode ou não continuar dirigindo

Operadores lógicos condicionais booleanos

Para fazer testes booleanos(testando condições em verdadeiro ou falso), os testes são feitos praticamente da mesma maneira que os operadores logicos acima, entretanto há alguns novos conceitos:

```
se (verdadeiro) {
        executar operacao
}
```

Se você quiser testar se a afirmação é falsa, você deve adicionar um ponto-de-exclamação '!' antes do nome da variável.

```
se (!verdadeiro) {
```


Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

```
        executar operacao
}
```

Exemplo:

```
<script>
var retorno = true;

// Testa se a condição é verdadeira.
if (retorno) {
        document.write('condição verdadeira');
}


// Testa se a condição é falsa.
if (!retorno) {
        document.write('condição falsa!');
}

</script>
```

EXERCICIO 5:

Desenvolver uma solução JavaScript que valide se alguém esta usando blusa de frio apenas em dias frios

Operadores lógicos condicionais usados dentro de funções

Caso uma função retorne um valor booleano, você pode fazer o teste da mesma maneira que é feito uma variável booleana.

```
se ( funcao() ) {
        executar operacao
}
```

Caso seja necessário testar se a função retorna um valor falso:

```
se ( !funcao() ) {
        executar operacao
}
```

exemplo:

```
<script>
// Testa se o retorno da função é verdadeira.
if (funcaoTesta()) {
        document.write('Condição verdadeira');
}
```


Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

```
// Testa se o retorno da função é falsa.
if (!funcaoTesta()) {
        document.write('Condição falsa!');
}
</script>
```

Usando a instrução switch

A instrução switch é uma maneira mais elegante de fazer a escolha de uma opção. Caso a instrução if .. else seja usada para vários testes, acaba sendo gerado uma serie de if´s e dificulta em muito a leitura do script.

A palavra switch, traduzida para o português significa escolha. Seu comportamento é muito semelhante ao verbo EVALUATE presente na linguagem COBOL.

```
escolha(opcao) {
        caso 1: opcao 1;
        caso 2: opcao 2;
        caso 3: opcao 3;
        caso outro: outra opção;
}
```

Traduzindo para o javascript.

```
<script>
switch(variavel) {
        case 1:
                document.write('Opção 1');
                break;
        case 2:
                document.write('Opção 2');
                break;
        case 3:
                document.write('Opção 3');
                break;
        default:
                document.write('Padrão');
                break;
}
</script>
```

É necessário usar a instrução break após a ações de cada case da instrução switch, por que na implementação da instrução switch não finaliza a opção selecionada. Caso não seja usado a instrução break, o fluxo do programa irá testar se a variável é igual as demais opções criadas na instrução switch.

Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

EXERCICIO 6:

Desenvolver uma solução JavaScript que tome uma ação baseada no nível de satisfação do cliente

Laços de Repetição

Existem várias formas diferentes de laços, mas eles essencialmente fazem a mesma coisa: repetir uma ação múltiplas vezes (inclusive você poderá repetir 0 vezes). Os vários mecanismos diferentes de laços oferecem diferentes formas de determinar quando este irá começar ou terminar. Há várias situações em que é mais fácil resolver um problema utilizando um determinado tipo de laço do que outros.

For

Um laço for é repetido até que a condição especificada seja falsa. O laço for no JavaScript é similar ao Java e C. Uma declaração for é feita da seguinte maneira:

```
<script>
for (condicaoInicial; condicaoFinal; acaoExecutar) {
        executa bloco de código;
}
</script>
```

Por exemplo:

```
<script>
        for (i=0; i<= 10; i++) {
        document.write('Linha '+i);
}
</script>
```

Constata-se que enquanto i for menor ou igual a 10 a instrução document.write('Linha '+i) será executada.

While

Equivalente a instrução enquanto (verbo PERFORM .. UNTIL na linguagem cobol), executa um bloco de operações até a condição da instrução enquanto ser atendida.

estrutura:

```
<script>
while (condicao) {
    bloco de operação
}
</script>
```


Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

```
<script>
var var1
while (var1 <= 10) {
        document.write('linha '+var1);
        var1++;
}
</script>
```

Do..While

A sintaxe desta estrutura é muito parecida com a estrutura do..while, existente na linguagem C. Ela tem a mesma funcionadalide que o while e o for, só que de uma maneira mais explicativa.

estrutura:

```
<script>
do {
        bloco de operacao
} while (condicao     );
</script>
```

Interrompendo a execução de estruturas de repetição

Para se interromper a execução de uma estrutura de repetição há duas situações:

1. A condição de execução da estrutura é alcançada.

Neste caso, a execução natural da estrutura repetição é suficiente para que isso aconteça.
exemplo:

```
<script>
while (i <= 10) { document.write('Linha ' + i); }
</script>
```

2. É usada a instrução **break** para interromper e finalizar a estrutura.

exemplo:

```
<script>
while(i <= 10) {
        if (i == 3) {
        // O break interrompe a execução na terceira execução da
        // estrutura de repetição.
                document.write('Linha ' + i + ' Fim do laço');
                break;
        }
```


Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br

```
        document.write('Linha ' + i);
}
</script>
```

O Resultado da execução deste script será:
Linha 1
Linha 2
Linha 3 Fim do laço

3. É usada a instrução continue para interromper a execução do bloco de instrução atual, não de todo a estrutura.

exemplo:

```
<script>
while(i <= 10) {
        if (i == 3) {
        // O continue interrompe APENAS a terceira execução da
        // estrutura de repetição.
                continue;
        }
        document.write('Linha ' + i);
}
</script>
```

Referências

MDN. (s.d.). *https://developer.mozilla.org/pt-BR/docs/Web/JavaScript/Guide/Fun%C3%A7%C3%B5es*. Fonte: MDN web docs.

Prof. Caique Zaneti Kirilo & Prof. Priscila Medeiro
ESCOLA SENAI DE INFORMÁTICA - Al. Barão de Limeira, 539 - Santa Cecília - São Paulo/SP - CEP 01202-001 Telefone: (11) 3273-5000 |E-mail:informatica@sp.senai.br