

ESMAD | TSIW | POO
Ficha de Exercícios nº1 – Introdução ao JavaScript

Use um playground (ex.: CodePen, JsFiddle, etc.) ou use o Chrome DevTools para resolver aos seguintes exercícios:

1. Crie uma instrução que exiba a mensagem "Olá JavaScript!".

2. Variáveis

- a. Trabalhar com variáveis
 - i. Declare duas variáveis: admin e name.
 - ii. Atribua o valor "John" à variável name.
 - iii. Copie o valor do name para admin.
 - iv. Mostre o valor de admin usando uma caixa de alerta (deve ser exibido "John").
- b. Nomeação de variáveis:
 - i. Crie uma variável com o nome do nosso planeta. Como chamaria a essa variável?
 - ii. Crie uma variável para armazenar o nome de um visitante atual de um site. Como nomearia essa variável?

3. Constantes:

- a. Examine o seguinte código que inclui uma data de aniversário constante e a idade é calculada a partir do aniversário com a ajuda de algum código (não é fornecido por questões de foco):
const birthday = '18 .04.1982 ';
const age = someCode (birthday);
- b. Seria correto usar letras maiúsculas para a variável **birthday**? E para **age**? Ou até para as duas?
const BIRTHDAY = '18 .04.1982 '; // maiúsculas?
const AGE = someCode (BIRTHDAY); // maiúsculas?

4. Tipos de dados:

- a. Qual o output da seguinte script:
let name = "Ilya";
alert(`hello \${1}`); // ?
alert(`hello \${"name"}`); // ?
alert(`hello \${name}`); // ?

5. Conversões de tipos:

- a. Quais são os resultados dessas expressões:
"" + 1 + 0
"" - 1 + 0
true + false
6 / "3"
"2" * "3"

```

4 + 5 + "px"
"$" + 4 + 5
"4" - 2
"4px" - 2
7 / 0
" -9 " + 5
" -9 " - 5
null + 1
undefined + 1

```

6. Operadores:

- a. Quais são os valores finais de todas as variáveis a, b, c e d após o código abaixo?

```
let a = 1, b = 1;
```

```
let c = ++a; // ?
```

```
let d = b++; // ?
```

- b. Quais são os valores de a e x depois do código abaixo?

```
let a = 2;
```

```
let x = 1 + (a *= 2);
```

7. Interação:

- a. Crie uma instrução que solicite um nome e imprima o nome juntando asteriscos antes e depois do nome.

8. Comparações:

- a. Qual será o resultado dessas expressões?

```
5 > 4
```

```
"apple" > "pineapple"
```

```
"2" > "12"
```

```
undefined == null
```

```
undefined === null
```

```
null == "\n0\n"
```

```
null === +"\n0\n"
```

9. Condicionais:

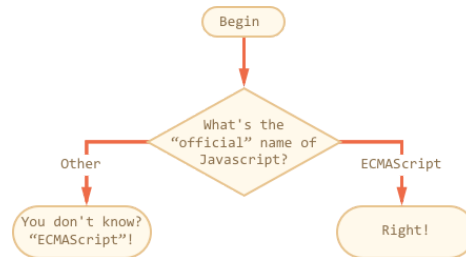
- a. O alerta será mostrado?

```

if ("0") {
  alert( 'Hello' );
}

```

- b. Usando a construção if..else, escreva o código que pergunta: "What's the "official" name of Javascript?". Se o visitante digitar "ECMAScript", envie "Right!", Caso contrário, a saída: "You don't know? "ECMAScript!"



- c. Usando if..else, escreva o código que obtém um número via prompt e, em seguida, mostra em alerta:

- i. 1, se o valor for maior que zero,
- ii. -1, se menor que zero,
- iii. 0, se igual a zero.

Nesta tarefa, assumimos que a entrada é sempre um número.

- d. Reescreva este if com o operador ternário '?':

```

if (a + b < 4) {
  result = 'Below';
} else {
  result = 'Over';
}
  
```

- e. Reescreva o próximo if...else usando múltiplos operadores ternários '?'. Para facilitar a leitura, é recomendável dividir o código em várias linhas.

let message;

```

if (login == 'Employee') {
  message = 'Hello';
} else if (login == 'Director') {
  message = 'Greetings';
} else if (login == "") {
  message = 'No login';
} else {
  message = "";
}
  
```

10. Operadores lógicos:

- a. Qual o output para cada linha de código?

```

alert( null || 2 || undefined );
alert( alert(1) || 2 || alert(3) );
alert( 1 && null && 2 );
alert( alert(1) && alert(2) );
alert( null || 2 && 3 || 4 );
  
```

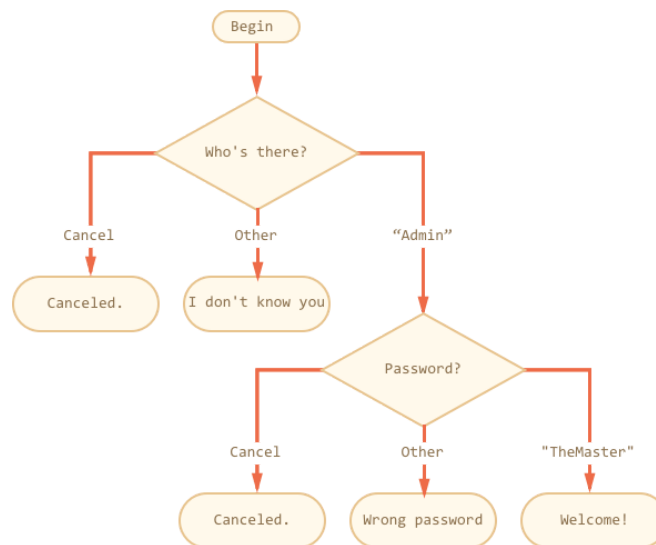
- b. Escreva uma condição "if" para verificar se a idade está entre 14 e 90, inclusive. "Inclusive" significa que a idade pode atingir as extremidades 14 ou 90.
- c. Escreva uma condição if para verificar se a idade NÃO está entre 14 e 90, inclusive. Crie duas variantes: a primeira usando NOT !, a segunda sem ela.
- d. Quais desses alertas serão executados? Quais serão os resultados das expressões dentro if (...)?

```

if (-1 || 0) alert( 'first' );
if (-1 && 0) alert( 'second' );
  
```

```
if (null || -1 && 1) alert( 'third' );
```

- e. Escreva o código que pede um login com prompt.
- Se o visitante digitar "Admin", solicite uma senha, se a entrada for uma linha vazia ou Esc - exibir "Cancelado". Se for outra, mostre "Não conheço você".
 - A senha é verificada da seguinte forma:
 - Se for igual a "TheMaster", então mostre "Welcome!",
 - Outra string - mostra "Senha incorreta",
 - Para uma sequência vazia ou entrada cancelada, mostre "Cancelado".
 - O esquema:



Por favor, use blocos aninhados. Observe a legibilidade geral do código.

11. Ciclos:

- Qual é o último valor alertado por este código? Por quê?

```
let i = 3;

while (i) {
  alert( i-- );
}
```
- Para cada iteração de loop, anote o valor que ele gera e, em seguida, compare-o com a solução. Ambos os loops alertam os mesmos valores ou não?
 - The prefix form ++i:

```
let i = 0;
while (++i < 5) alert( i );
```
 - The postfix form ++i:

```
let i = 0;
while (i++ < 5) alert( i );
```

- c. Para cada loop anote quais valores ele irá mostrar. Em seguida, compare com a resposta. Ambos os loops alertam os mesmos valores ou não?
 - i. The prefix form ++i:
for (let i = 0; i < 5; i++) alert(i);
 - ii. The postfix form ++i:
for (let i = 0; i < 5; ++i) alert(i);
- d. Use o loop for para gerar números pares de 2 a 10.
- e. Reescreva o código alterando o loop for sem alterar seu comportamento (a saída deve permanecer igual).
**for (let i = 0; i < 3; i++) {
 alert(`number \${i}!`);
}**
- f. Escreva um loop que solicite um número maior que 100. Se o visitante inserir outro número, peça-lhes para inserir novamente. O loop deve solicitar um número até que o visitante insira um número maior que 100 ou cancele a entrada / insira uma linha vazia. Aqui podemos supor que o visitante só insere números. Não há necessidade de implementar um tratamento especial para uma entrada não numérica nessa tarefa.
- g. Um número inteiro maior que 1 é chamado de primo se não puder ser dividido sem um resto por qualquer coisa exceto 1 e ele mesmo. Em outras palavras, $n > 1$ é primo se não puder ser dividido por qualquer coisa, exceto 1 e n . Por exemplo, 5 é um primo, porque não pode ser dividido sem um resto por 2, 3 e 4.
Escreva o código que gera números primos no intervalo de 2 a n .
Para $n = 10$, o resultado será 2,3,5,7.
P.S. O código deve funcionar para qualquer n , não deve ser ajustado para qualquer valor fixo.

12. Switch

- a. Escreva o código usando if..else que corresponderia ao seguinte switch:
**switch (browser) {
 case 'Edge':
 alert("You've got the Edge!");
 break;

 case 'Chrome':
 case 'Firefox':
 case 'Safari':
 case 'Opera':
 alert('Okay we support these browsers too');
 break;

 default:
 alert('We hope that this page looks ok!');
}**
- b. Reescreva o código abaixo usando uma única instrução switch:
let a = +prompt('a?', "");

```
if (a == 0) {  
  alert( 0 );  
}  
if (a == 1) {  
  alert( 1 );  
}  
if (a == 2 || a == 3) {  
  alert( '2,3' );  
}
```