

Regras dos exercícios

1. Utilize um arquivo HTML e uma tag script para resolver esse exercício.
2. Faça do seu jeito, mas lembre-se não esqueça que o objetivo é manipular o DOM, sendo assim, faça seu melhor para fazer todas as operações com a árvore (adição, remoção, busca, etc.)

Exercício 1 – Faça um programa que capture o evento de um input na sua tela e mostre automaticamente a tabuada desse número abaixo.

Exercício 2 – Faça um programa que “pegue” uma lista `` no HTML e coloque negrito dinamicamente a partir de um nome digitado no seu input. Por exemplo
A sua tela possui uma lista de nomes:

- João
- Claudio
- Marcio

Usando um input você digita “Jo”, então o programa no mesmo momento deverá devolver o seguinte resultado:

- João**
- Claudio
- Marcio

Intermediário

Avançado

Regras do exercício

1. Os exercícios Javascript propostos a seguir devem ser resolvidos em arquivos diferentes e recomendo usar também pastas diferentes.
 2. Cada resposta deve estar contida dentro de uma div principal, ou seja, os códigos (inclusive o HTML) deve ser construído usando o JS e inserido na interface usando como referência um div (`<div id="q1"> </div>`).
 3. Lembre-se que o componente deve alertar o usuário em caso de erro.
Nesse exercício vamos usar a terminologia “componente” quando queremos nos referir a um conjunto de elementos HTML, CSS e JS que trabalham juntos para satisfazer uma funcionalidade. Os componentes são na verdade um conceito um pouco mais complexo e são muito usados em frameworks como o [React](#). No entanto, nosso objetivo aqui é apenas entender o “core” dessa ideia e buscar organizar alguns trechos de código que poderiam ser mais facilmente reutilizados em outras aplicações.
-

Exercício 01 – crie um “componente” que possua:

- Duas caixas de texto (definir rótulos (label) com um valor mínimo e um valor máximo)

- Um botão (com o label “contar”)

Nesse componente as caixas de texto tem algumas restrições:

- Só podem aceitar números.
- Sempre que um número for digitado, checar se o valor mínimo é menor que o valor máximo.

Quando clicamos no botão o componente deve contar e imprimir quantos números existem entre valor mínimo e valor máximo que sejam múltiplos de 2 e 3 (simultaneamente).

Ahh, tem mais uma coisa... o programa não deverá calcular nada se algum dos numeros não for inserido.

Exercício 02 – crie um componente possua:

- Uma caixa de texto (que só aceita números e tenha um rótulo (label)).
- Um botão “calcular”.

Use o número para calcular o fatorial dele e mostrá-lo na tela. Além disso, quando o botão é acionado o componente deve também mostrar tempo necessário para a execução.

Exercício 03 – Crie um componente que possua:

- Um botão “criar relatório”

Quando acionado o seu programa deverá imprimir um relatório de resultados da disciplina javascript para 20 alunos, no formato de tabela.

Mas vamos complicar um pouco ne? veja algumas regras adicionais!

- As notas deverão ser números inteiros entre 0 e 100, criadas aleatoriamente; (dica: use a função `math.random()`, dá um google!)
- Cada aluno deve ser representado por um registro composto por numero e nota
- Use uma função construtora chamada Aluno. Se você for bem ousado, implemente usando classes;
- Armazene a nota do aluno como uma variável privada do tipo `symbol`, use getters e setters para recuperar a informação.
- Os registros devem ser armazenados em um array;

A impressão do relatório deve ter:

- Um título;
- Os resultados mostrados linha a linha no formato “Aluno Nº ?? – Nota ?? [(A/RE)PROVADO]”; e
- Um rodapé com estatística final no formato “APROVADOS: ??(??%) | REPROVADOS: YY (??%)”

Exercício 04 – Crie um componente que possui:

- 3 caixas de texto (só aceitam números e possuem rótulos (label) com a quantidade de números, valor mínimo e valor máximo)
- 1 botão calcular

Ao clicar no botão de “calcular”:

- Um array deve ser criado com a quantidade de números aleatórios (informados pelo usuário) no intervalo informado.

- Os números devem estar entre valor mínimo e valor máximo (inclusivos).
 - Calcular quantidade de números deve ser menor que a soma de valor mínimo e valor máximo.
-

Exercício 05 – Crie um componente que possui:

- 3 caixas de texto que só aceitam números e tenha os rótulos (label): lado A, lado B e lado C.
- 1 botão “calcular”

Quando acionado o botão deve dizer se um triângulo é formado por esses 3 lados. Caso afirmativo, informe se o triângulo equilátero, isósceles ou escaleno. (esse é bem simples né =D)

Exercício 06 – Agora vamos criar um último componente mais avançado. Nesse, você precisa criar campos para usuário e login em aplicação(sign up/ sign in).

- No 1º cenário, antes de estar logado, o visitante se depara com o formulário de login ou de criação de usuário. Se for feito o login com sucesso, o componente deve levar ao cenário 2, se falhar ele deve alertar o usuário e voltar ao início do cenário 1. Se o usuário optar por criar um usuário, o componente deve criar o registro de um novo usuário e voltar para início do cenário 1.
- No cenário 2, após logado, mostrar apenas um texto de logado no componente e um botão (ou link) para deslogar, retornando ao início do cenário 1.

Deve ser possível criar múltiplos usuários e, se fechada, a página não pode perder os registros de usuários armazenados.

Desafio!

A ideia principal dos componentes é que você possa “invocá-los” dentro do seu código facilmente, porém da forma que eles foram feitos até agora não torna isso possível, visto que todos os componentes estão armazenados em arquivos diferentes. Sendo assim, o desafio aqui é “juntar” todos esses componentes criados em apenas 1 arquivo principal (HTML) e manter sua organização em diferentes arquivos javascript. É de suma importância também que as variáveis de um componente não sejam visíveis para os demais.

Portanto, quando eu quiser montar uma página com esses componentes eu deveria simplesmente colocar o código HTML:

```
<div id="ex1"> </div>
```