



Instruções Gerais:

- Atividade Deverá ser Entregue Individualmente.
- Entregar a solução completa em formato ZIP pela tarefa do SIGAA.
- **Horário de Entrega definido na Tarefa do SIGAA**
- Formato de Entrega: Coloque os arquivos referentes a tarefa (.html, .css, e .js) em um diretório. Compacte esse diretório em formato ".zip" e faça a entrega no SIGAA.


Objetivos: Familiarizar com a manipulação do DOM com JavaScript:

- Selecionar Elementos
- Tratar Eventos
- Modificar Elementos

Tarefa Única: Jogo Educativo para as operações básicas de soma, subtração e multiplicação.

Nessa tarefa você deverá criar um jogo estilo *quis* para validar as operações básicas de somar, multiplicar e subtrair. A Figura 1 apresenta a visão geral da tela do jogo.

Jogo da Matemática



Gerar Equação Validar Resposta

Insira sua resposta

Resposta

Figura 1

Funcionamento:

- 1 – Ao clicar no botão “Gerar Equação” uma equação será apresentada na área acima desse botão.
 - 2 – Ao clicar no botão “Validar Resposta” o programa fará a leitura do valor inserido no campo “Insira sua Resposta” e fará a verificação se está correto.
 - 3 – Após a verificação o programa irá emitir uma mensagem abaixo do campo “Insira sua Resposta” informando se a resposta está certa ou errada. Se estiver certa a mensagem será da cor verde e com os dizeres “Parabéns”. Se estiver incorreta a mensagem será da cor vermelha com os dizeres “Deu Ruim. A resposta correta é X”, onde X é o valor que seria correto.
 - 4 – Os operadores podem assumir apenas valores inteiros no intervalo [0-9].
- As Figuras 2-4 apresentam um exemplo de funcionamento.

Jogo da Matemática



6 + 5

Gerar Equação

Validar Resposta

Insira sua resposta

Resposta

Figura 2

Jogo da Matemática



6 + 5

Gerar Equação

Validar Resposta

Insira sua resposta

11

Parabéns!

Figura 2

Jogo da Matemática



6 + 5

Gerar Equação

Validar Resposta

Insira sua resposta

15

Deu Ruim! A resposta
correta é 11

Figura 3

Dicas:

Exemplo 1 : Registrando eventos de clique em um botão usando *arrow func*

```
const btn = document.querySelector('button');
```

```
btn.addEventListener('click', () => {  
  console.log("Fui clicado");  
});
```

Nesse exemplo ao invés de passar o *callback*, fizemos a própria função dentro do campo do segundo parâmetro da função *addEventListener*. Foi utilizada a notação funcional, isto é, com “*arrow function*”.

Exemplo 2: Buscando valor de <input> e registrando “click”

Você também tratar o evento diretamente no botão. Nesse caso o tipo do evento é “click” e o botão requer que no HTML o type seja “button”. Para pegar o valor do <input> basta acessar a propriedade “value”

HTML

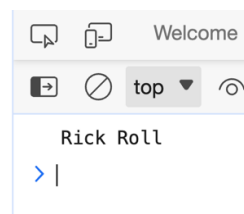
```
<label for="username">Username</label>  
<input type="text" name="username" id="username">  
<button type="button">Clique Aqui</button>
```

JavaScript

```
const btn = document.querySelector('button');  
const input = document.querySelector('#username');
```

```
btn.addEventListener('click', () => {  
  let valorInput = input.value;  
  console.log(valorInput); //Irá imprimir "Rick Roll"  
});
```

Username



OBS: Recomendo não utilizar a função “eval” pois já mostrado que ela pode comprometer a segurança de código JavaScript.

Link: [eval\(\) - JavaScript | MDN \(mozilla.org\)](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/eval)