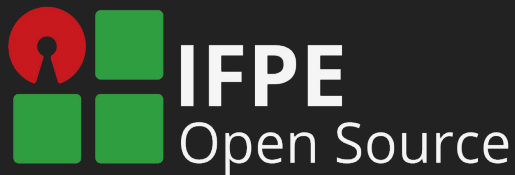




Comunidade de
tecnologia e computação

Objetivos



Unir pessoas **interessadas em tecnologia**,
computação, programação, cultura maker, etc

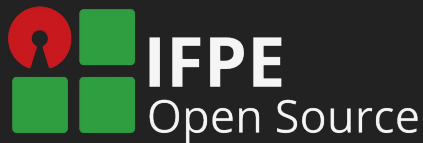


Desenvolver projetos **open source** e colocar em
prática os conhecimentos em tecnologia



Organizar workshops, palestras, eventos, hackathons,
etc e **estimular o interesse** de outros alunos

Criando sites e games na web



CSS



HTML



JavaScript



```

<!DOCTYPE html>
<html lang="pt-BR">
<head>
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Alvinator!</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
<header>
<div>
<h1>Alvinator!</h1>
<p class="paragrafo">
0
<a
href="https://projetos.ifpeopensource.com.br/alvinator"
target="_blank" ref="noopener noreferrer"
>
alvinator
</a>
é uma plataforma onde estudantes podem resolver questões
de trigonometria.
</p>
</div>

</header>

```

Alvinator!

O alvinator é uma plataforma onde estudantes podem resolver questões de trigonometria.



Como funciona?

A aplicação calcula os lados, alturas, projeções, área e perímetro do triângulo baseado em apenas dois valores dados pelo usuário (tudo *client-side*).

Acessar Alvinator!

Próximo Projeto

Um projeto



Feito com
♥ e 🖥️





HTML

(HyperText
Markup Language)

- É uma linguagem de *marcação*
- Tags: `</>`

Fecham: `<tag> </tag>`

Não Fecham: `<tag/>`

p, h1, h2, h3, b,
small, div, a, i



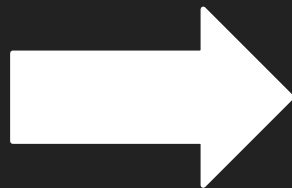
<h1> Olá! </h1>

<h2> Olá! </h2>

<h3> Olá! </h3>

<p> Olá! </p>

<small> Olá! </small>



Olá!

Olá!

Olá!

Olá!

Olá!



`<div>`

`<h1> Olá! </h1>`

`<p> Tudo bem? </p>`

`</div>`

`<div>`

`<h1> Oi! </h1>`

`<p> Como vai? </p>`

`</div>`



Olá!

Tudo bem?

Oi!

Como vai?



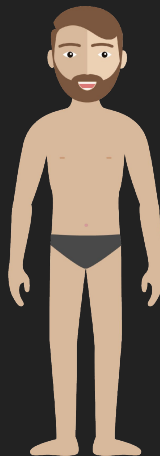


CSS

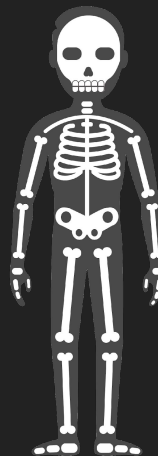
(Cascading
Style Sheets)

- É uma linguagem de *estilo ou estilização*

CSS



HTML



Javascript



Sintaxe

```
seletor {  
  propriedade: valor;  
  propriedade2: valor2;  
}
```



```
h1 {  
  font-size: 10px;  
  color: white;  
}
```

Seletores: elemento que será estilizado

Propriedades: estilo que será modificado



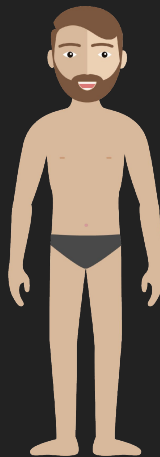


JavaScript

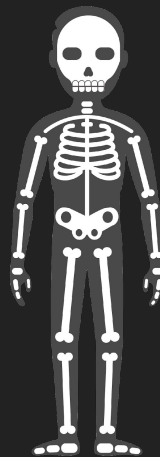
(Não, não é a mesma coisa que Java)

- É uma linguagem de programação que permite a criação de scripts na web do lado do usuário

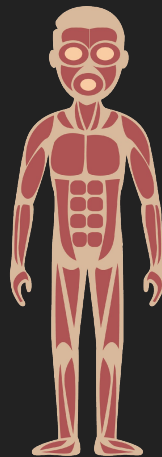
CSS



HTML



Javascript



Vamos programar?

```
console.log('Hello World!')
```

```
nome = 'João'
```

```
console.log('Olá, ' + nome)
```



Vamos programar?

```
const nome = 'João'
```

→ Cria um valor constante, que não pode ser alterado

```
let sobrenome = 'Silva';
```

→ Cria um valor que pode ser alterado

```
console.log(`Olá, ${nome} ${sobrenome}`);
```



Mas e os games?

O que vamos fazer hoje?







KABOOM

- “É uma biblioteca de programação de jogos para JavaScript que lhe ajuda a criar jogos rápidos e divertidos”

`kaboomjs.com`



Vamos desenvolver nosso game?!

Vamos editar o arquivo JavaScript `game.js`.



Vamos desenvolver nosso game?!

Vamos editar o arquivo JavaScript `game.js`.

O código do nosso jogo vai estar em

```
scene ( 'game' ,  ()  =>  { . . . } )
```



Criando o chão e o jogador

```
add([
    rect(width(), 48),
    pos(0, height() - 48),
    outline(2),
    area(),
    solid(),
    color(127, 200, 255),
]);
```

add() -> Adiciona um objeto no game com base em uma lista de componentes desse objeto

rect() -> Retângulo

width() -> Comprimento da tela

outline() -> Borda

area() -> É um objeto que possui área e caixa de colisão

solid() -> É um sólido afetado pela gravidade

color() -> Cor do fundo



Funções em JavaScript

Trechos de código que podemos controlar o momento em que são executados

```
function nomeDaFunção () {  
    ...  
}
```

```
() => {  
    ...  
}
```

Arrow Functions =>
Funções Anônimas

Fazendo o jogador pular

```
objeto.isGrounded()
```

Retorna verdadeiro se o objeto estiver em cima de uma plataforma

```
onKeyPress('tecla', função)
```

```
onClick(função)
```

Executa a função assim que a tecla indicada for pressionada / clicar na tela



Inserindo desafios!

```
wait(segundos, função)
```

Executa a função após os segundos se passarem

```
rand(valor1, valor2)
```

Gera um número aleatório no intervalo informado



Pontuação!! 🕶️ Derrota!! 😐

```
onUpdate (função)
```

Executa a função em cada *frame* do jogo

```
objeto.onCollide('outroObjeto', função)
```

Executa a função sempre que o objeto colidir com outro(s) que tenham a *tag* indicada



Morte 🤪

```
burp()
```

Toca um som de arrote, padrão do Kaboom

```
addKaboom(posição)
```

Adiciona um efeito de explosão

```
shake()
```

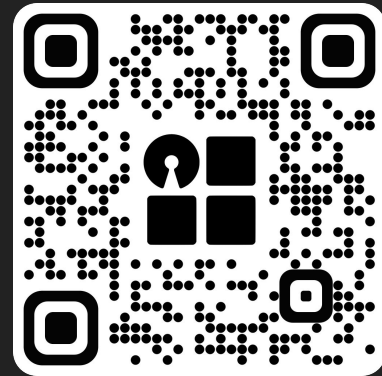
Treme a tela

```
objeto.destroy()
```

Destrói o objeto



Obrigado!



Poste o seu game no insta ou LinkedIn e marque o



@ifpeopensource



IFPE Open Source



IFPE

Open Source

