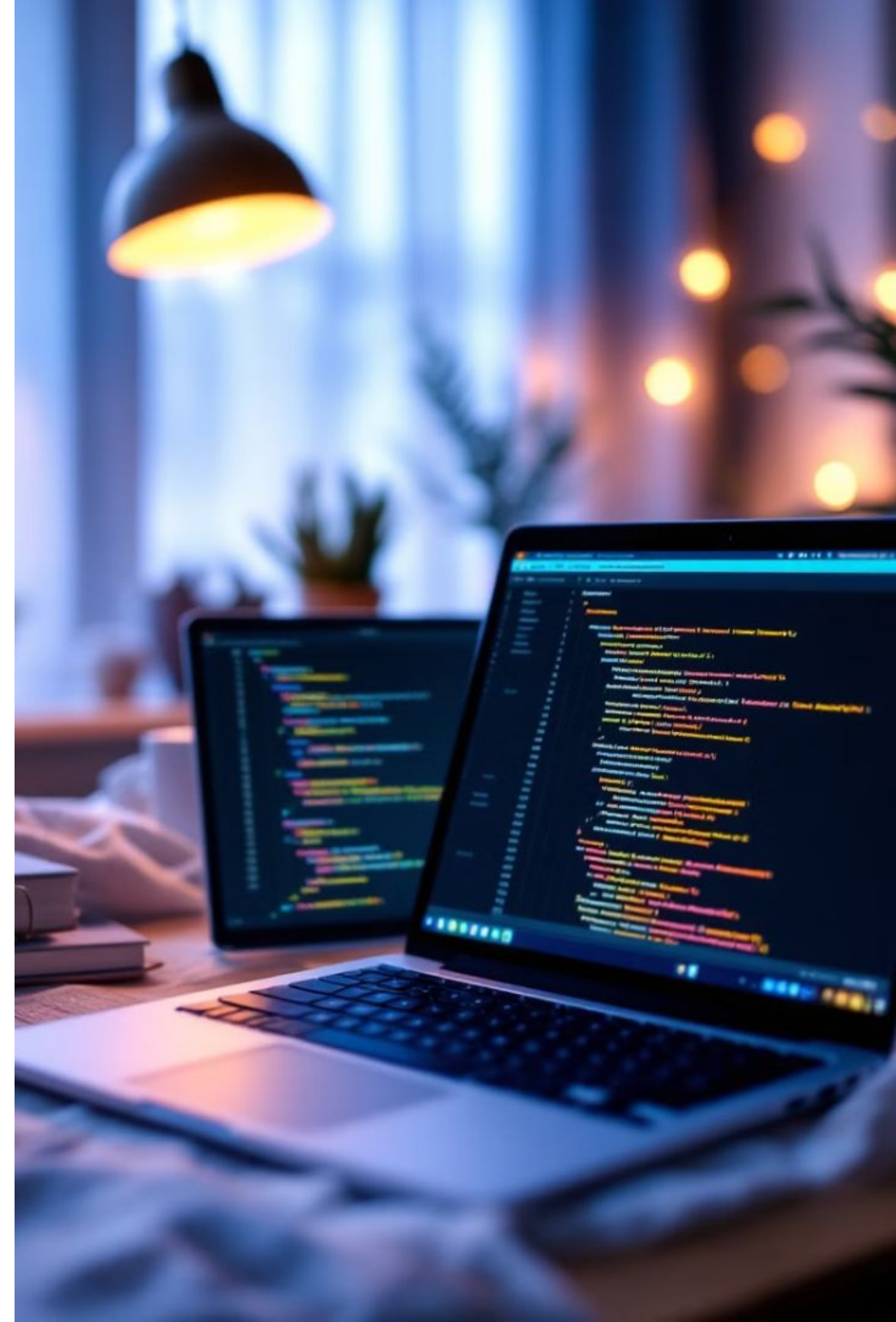


Programação em Javascript, HTML e CSS

Bem-vindos, futuros desenvolvedores! Depois de brincar com a lógica no Portugol, chegou a hora de colocar a mão na massa de verdade! Hoje vamos usar HTML, CSS e JavaScript para transformar código em coisa que a gente vê, clica e interage.



by **ALEX MUNIZ**



Sintaxe Básica do JavaScript

A sintaxe é o alicerce de qualquer linguagem de programação. Em JavaScript, ela define como o código deve ser escrito para ser compreendido pelo computador. Vamos conhecer os elementos básicos que formam a sintaxe do JavaScript e compará-los com o Portugol.

Comentários

```
// comentário de uma linha  
/* comentário de várias linhas */
```

Delimitação de blocos

```
{ ... }
```

Instruções

```
let x = 5;
```

Atribuição x Comparação

Atribuição (=)

Atribuir um valor para uma variável

ex: let idade = 21

Comparação

Igualdade solta (==)

Compara apenas o valor e converte tipos diferentes



```
5 == "5"; // true (conversão implícita de string para número)
```

Igualdade estrita (===)

Compara valor e **tipo**



```
5 === "5"; // false (número !== string)
5 === 5;   // true
```

Declaração de Variáveis e Constantes

Variáveis e constantes são elementos essenciais para armazenar dados em JavaScript. Vamos explorar as palavras-chave utilizadas para declarar variáveis (`let` e `var`) e constantes (`const`), entendendo suas diferenças e como utilizá-las corretamente.



var

Antiga (evitar). Escopo de função.



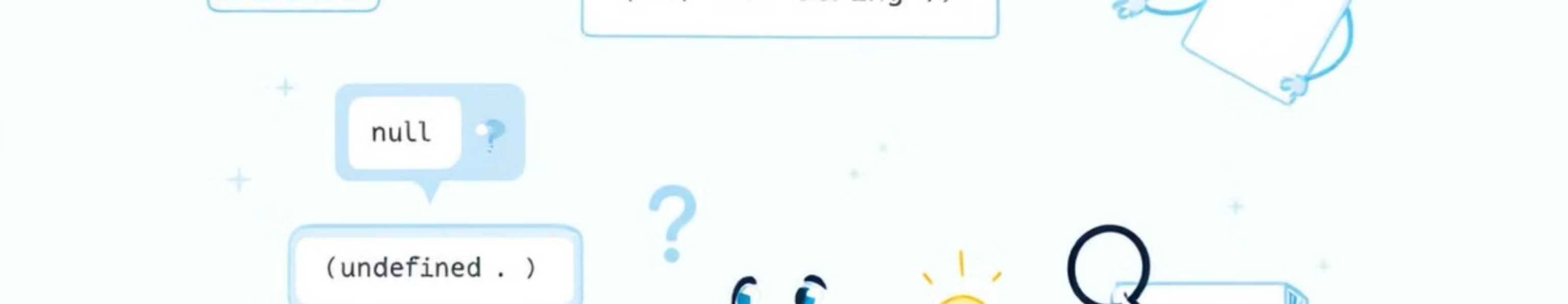
let

Moderna. Escopo de bloco.



const

Moderna. Não pode ser reatribuída. Escopo de bloco.



Tipos de Dados em JavaScript

JavaScript é uma linguagem dinamicamente tipada, o que significa que não precisamos declarar o tipo da variável. O tipo é definido automaticamente com base no valor atribuído. Vamos explorar os tipos primitivos de dados em JavaScript e como utilizá-los.

number

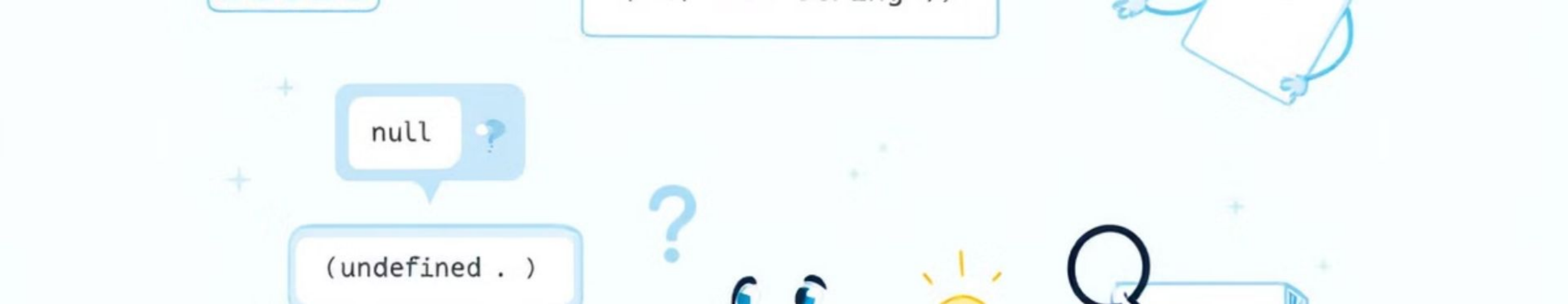
Números inteiros ou reais.

string

Textos, sempre entre aspas simples ou duplas.

boolean

Verdadeiro ou falso (lógico). **true** ou **false**



Null x Undefined

null

Representa a ausência intencional de valor.

```
let idade = null;  
console.log(idade); // null
```

undefined

Representa ausência de valor atribuída automaticamente pelo JavaScript.

```
let nome;  
console.log(nome); // undefined
```



```
5      fervernames in, _putient uoctdân, isOnline: true;
20      fietrutise:sisar:(sinal);
11      pestlnine: fyser(laninutue)
16      usarnaame: fyser dsoNName, and__withninc.isOnline: true)
16      parenants, maylic,lect_&ccMauth,(atervat(ycinis)))
19      pantauate: uyser usabliny,tue)}};
21      userr sctONninee: 30)
```

Uso do Console no Navegador

O console do navegador é uma ferramenta poderosa para depurar e testar código JavaScript. Ele permite exibir saídas, mensagens de erro e interagir diretamente com o código em execução. Vamos aprender como utilizar o `console.log` para exibir informações no console.

1

Abra o navegador

2

Pressione F12

ou clique com o botão direito > Inspecionar > aba "Console".

3

Escreva comandos

diretamente ou veja saídas do código JS em uma página.

Ambiente de Desenvolvimento

Para começar a programar em JavaScript, você precisa de um ambiente de desenvolvimento adequado. Existem diversas opções disponíveis, desde editores de código profissionais até plataformas online que permitem escrever e executar código diretamente no navegador. Vamos explorar algumas sugestões.



VS Code



Navegador



Plataformas online



Resumo Comparativo – Portugol vs JavaScript

Para consolidar o que aprendemos, vamos fazer um resumo comparativo entre o Portugol e o JavaScript, destacando as principais diferenças e semelhanças na sintaxe e nos conceitos básicos. Este resumo será útil para quem está migrando do Portugol para o JavaScript.

Conceito	Portugol	JavaScript
Início do programa	{ }	{ ... } ou direto no script
Impressão	escreva("texto")	console.log("texto")
Variáveis	inteiro, real, cadeia	let, const (tipos dinâmicos)



Operadores JavaScript

Usar operadores JS em expressões lógicas complexas e entender suas aplicações práticas.



Estruturas Condicionais

Compreender e aplicar estruturas de decisão para controle de fluxo em programas.



Conversão de Lógica

Traduzir algoritmos de Portugol para JavaScript mantendo a mesma lógica.

Operadores em JavaScript

Operadores Aritméticos

- Adição (+)
- Subtração (-)
- Multiplicação (*)
- Divisão (/)
- Módulo (%)

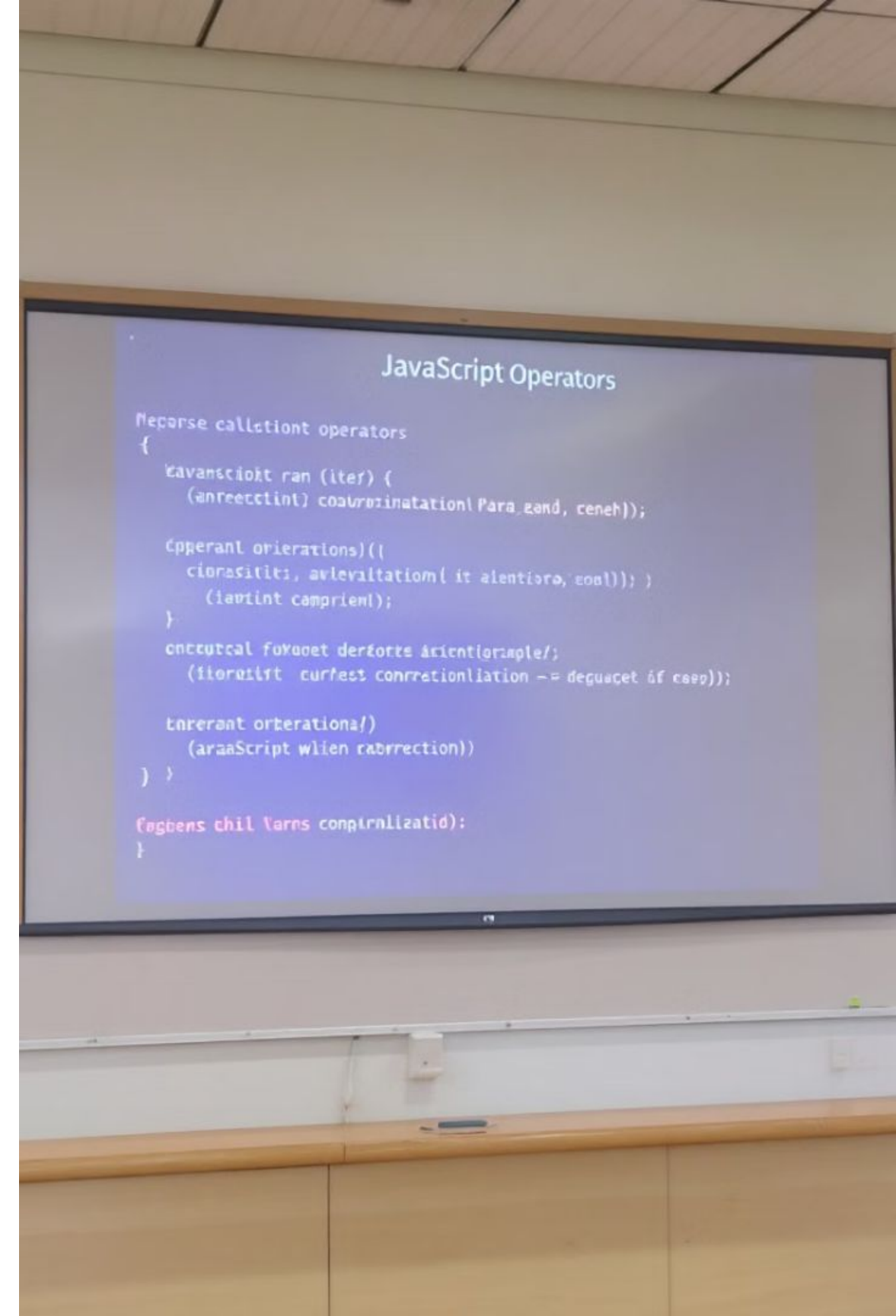
Operadores Relacionais

- Maior que (>)
- Menor que (<)
- Maior ou igual (>=)
- Menor ou igual (<=)
- Igual (==, ===)
- Diferente (!=, !==)

Operadores Lógicos

- E lógico (ampersand)
- OU lógico (pipe)
- NÃO lógico (exclamation mark)

Atividade rápida (individual ou em dupla): Resolver pequenos trechos de código JS e prever o resultado de expressões como: `let x = 10; let y = 5; console.log(x > y && y !== 0);`



Estruturas Condicionais em JS



if/else

Estrutura básica para tomada de decisões baseada em condições booleanas.

Exemplo:

```
if (idade >= 18) { console.log("Maior de idade"); }  
else { console.log("Menor de idade"); }
```



else if

Permite verificar múltiplas condições em sequência.

Útil quando há mais de duas possibilidades de resultado.



switch

Alternativa ao if/else quando há múltiplas comparações com um mesmo valor.

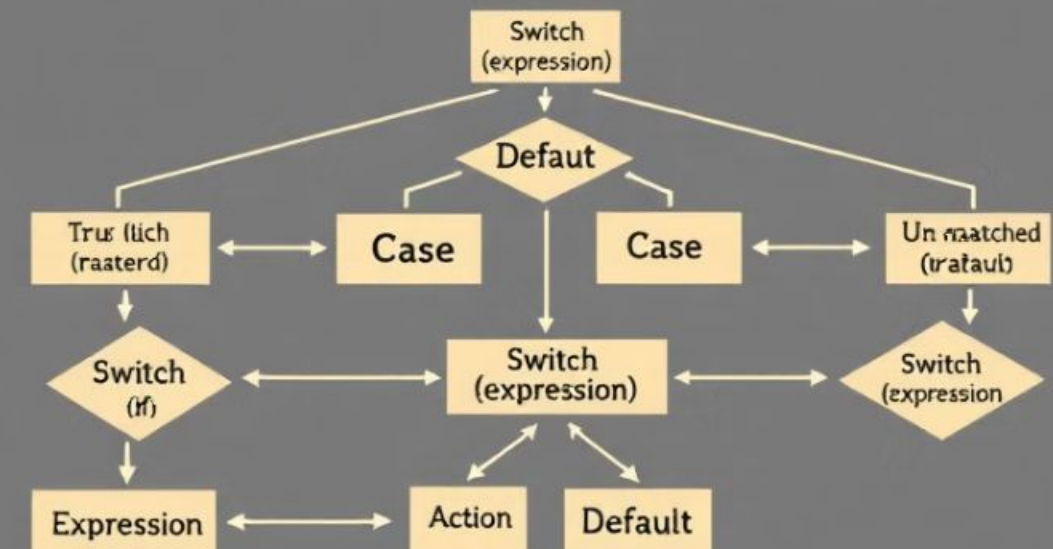
Mais limpo e eficiente para verificar valores específicos de uma variável.

JavaScript Decision Structures

The ogram. limrogrdow, with unnalacher foy hian stuthers : mard is you uolds?



(switch (expression : 2 Detaich (smatch, Default Action))



Comparação



Comparação com **Portugol**:

se (idade >= 18) então escreva("Maior de idade")

senão escreva("Menor de idade")

Javascript

```
if (idade >= 18) { console.log("Maior de idade"); }
```

```
else { console.log("Menor de idade"); }
```

Atividade Prática de Conversão

Algoritmo em Portugol

Receber algoritmos escritos em Portugol com estruturas de decisão

Verificação

Testar se o código mantém a mesma lógica e resultados



Análise da Lógica

Compreender a lógica do algoritmo independente da sintaxe

Conversão para JavaScript

Reescrever utilizando a sintaxe correta de JavaScript

Exemplo: se (nota \geq 6) então escreva("Aprovado") senão escreva("Reprovado") \rightarrow if (nota \geq 6) { console.log("Aprovado"); } else { console.log("Reprovado"); }

Exemplo:

Portugol

```
se (nota >= 6) {  
    escreva("Aprovado")  
} senão {  
    escreva("Reprovado")  
}
```

Javascript

```
if (nota >= 6) {  
    console.log("Aprovado");  
} else { console.log("Reprovado"); }
```



Revisão



Operadores

Aritméticos, relacionais e lógicos para construir expressões



Estruturas

Conditionais

if/else e switch para controle de fluxo



Conversão

De Portugol para JavaScript mantendo a lógica

Mini-desafio: "Escreva um programa JS que avalia a idade e a nota de um aluno e retorna sua situação."

Desafio da aula



Criar algoritmo

Desenvolver código com operadores e decisões. Baseado na lista de exercício disponível no AVA, exercícios que fizeram em portugol



Apresentar para o professor

Compartilhar soluções em portugol e javascript