

{reprograma}

Jéssica Osko

JavaScript IV

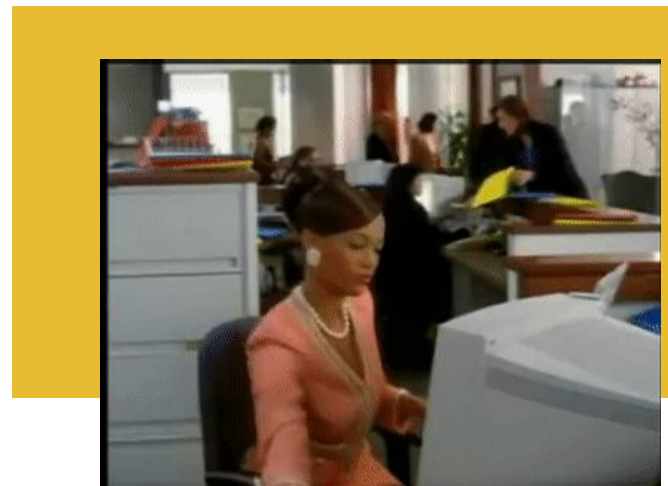
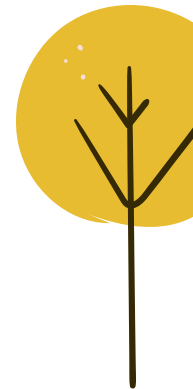


Eu sou a Jéssica,
desenvolvedora e estou aqui
para compartilhar o meu
conhecimento com vocês.

@oskojess



O QUE EU APRENDI



Mentalidade fixa x Mentalidade Esforçada

Inteligência e habilidades são inatas e estáticas.

O erro é indicativo da falta de habilidade, logo, uma fraqueza

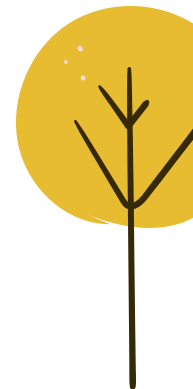
Evita desafios com medo de que revele fraquezas.

É possível desenvolvê-las

O erro faz parte do aprendizado. Importante é o esforço.

Vê o desafio como oportunidade de crescimento.

Fonte: Mindset, Carol Dweck.



Vamos fazer alguns combinados!?

- Não ter vergonha de perguntar
- Errar faz parte da aprendizagem





Revisar

<http://donthpad.com/reprograma/javascript-IV>





Nodejs

**É uma plataforma que utiliza o JavaScript como sintaxe.
Através dele, é possível desenvolver pequenas e grandes
aplicações.**



Objetos

Objetos é uma coleção de dados e/ou funcionalidades relacionadas.



#NaPratica



Praticar

A Empresa **PhoneJá** gostaria de um programa onde eles pudessem cadastrar os clientes que comprassem celulares em suas lojas credenciadas pelo País.

Os campos iniciais são:

- **Nome**
- **CPF**
- **Data da compra**
- **Valor**
- **Modelo do Celular**
- **Ano do celular**
- **Cor do celular**
- **Loja**
- **Nome da atendente**

A partir do **conhecimentos adquiridos** até aqui, qual seria a melhor forma de **criar esses campos iniciais**?

Crie o programa com os itens citados e **cadastre** *três clientes*.



Vamos discutir sobre a
revisão e tirarmos as
dúvidas?

É **sempre bom**
lembrar que **não**
existem dúvidas
“*simples*” ou “*óbvias*”.



Escreva perguntas **sem medo**



Dicas de Leituras:

Você não conhece JS

Eloquente JavaScript

Mulheres na
Tecnologia

Algumas dicas para
aprender a
programar

Dicas de leituras para aprender mais sobre a história da linguagem JavaScript e também entendermos o nosso cenário na tecnologia.



{reprograma}

Jéssica Osko

JavaScript IV



#NaPratica





Arquivar

<http://dontpad.com/reprograma/javascript-IV>



#Motivar

**PROGRAMAÇÃO
ESTRUTURADA**

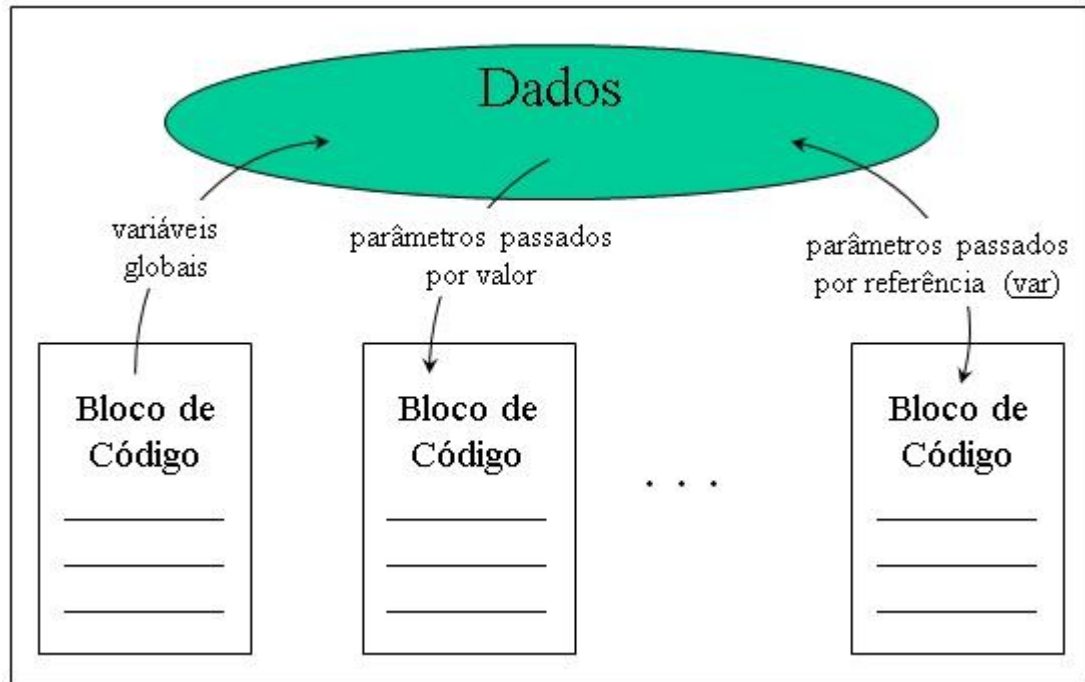
SEQUÊNCIA

DECISÃO

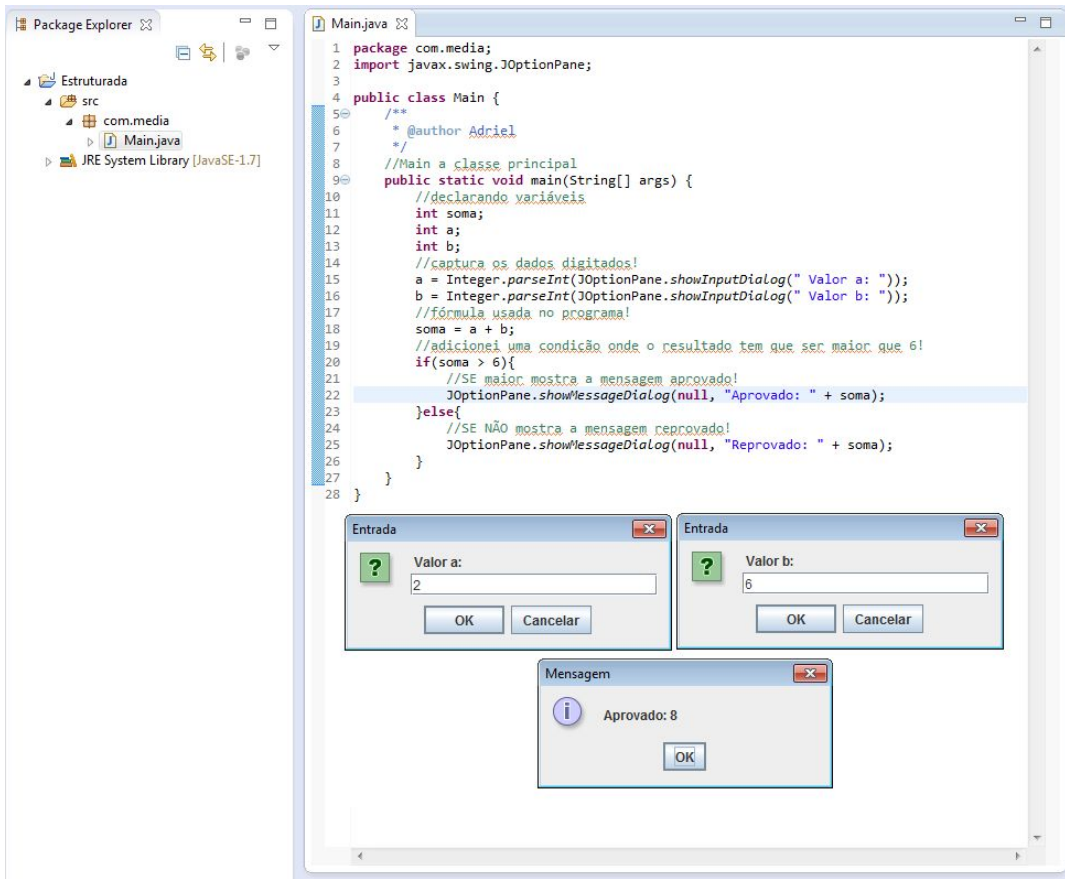
ITERAÇÃO

**PROGRAMAÇÃO
ORIENTADA A OBJETOS**





Fonte: Intro. Linguagem Estruturada



Na imagem da programação estruturada, fica entendível quando falamos em declarar variáveis, métodos e funções em uma mesma página.

#Motivar

**PROGRAMAÇÃO
ESTRUTURADA**

**PROGRAMAÇÃO
ORIENTADA A OBJETOS**

SEQUÊNCIA

DECISÃO

ITERAÇÃO

MODELAR

REUTILIZAR

PADRONIZAR



Programar de forma
modular é vida!
Reaproveito código e
ainda tenho tempo para
um café!

Amada,
o nome disso é
**orientação a
objetos.**





POO

Programação Orientada a Objetos

Promover maior flexibilidade
e facilidade de manutenção
na aplicação.

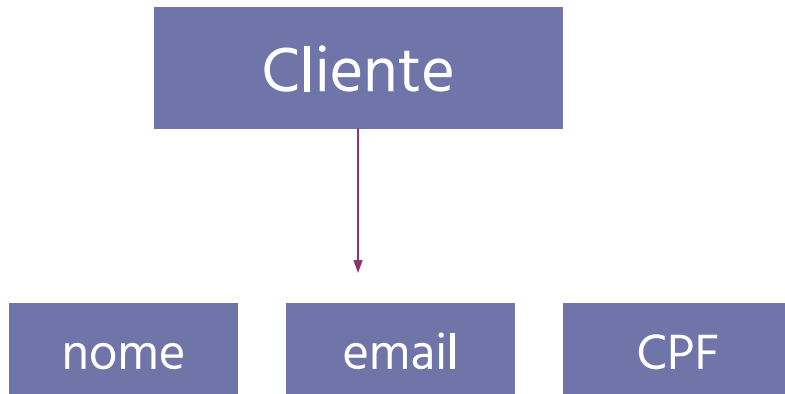
Classes

Classe é um conjunto de características e comportamentos que definem o conjunto de objetos pertencentes à essa classe.



#Classificar

{reprograma}



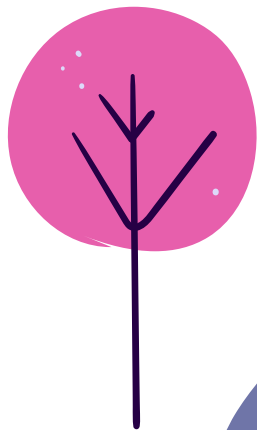
#NaPratica



Métodos

Uma ação do objeto.

Eles determinam o comportamento dos objetos de uma classe. Sendo assim, podem alterar o estado de um objeto em determinadas ações.





Definir

CONSTRUCTOR

Um método chamado assim que um novo exemplar do objeto for criado. Ele geralmente tem o mesmo nome da classe que o contém.

this

Refere-se a um objeto atual no qual está sendo executando no momento.



#NaPratica

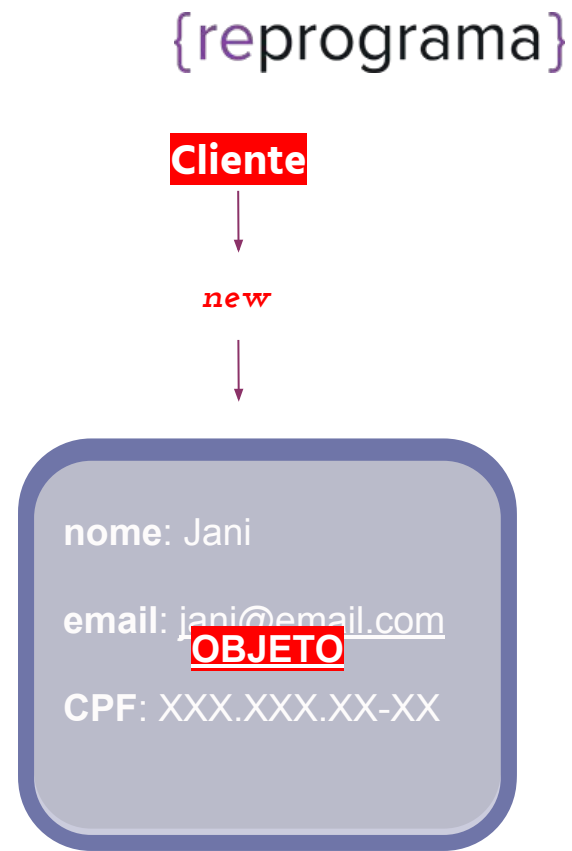
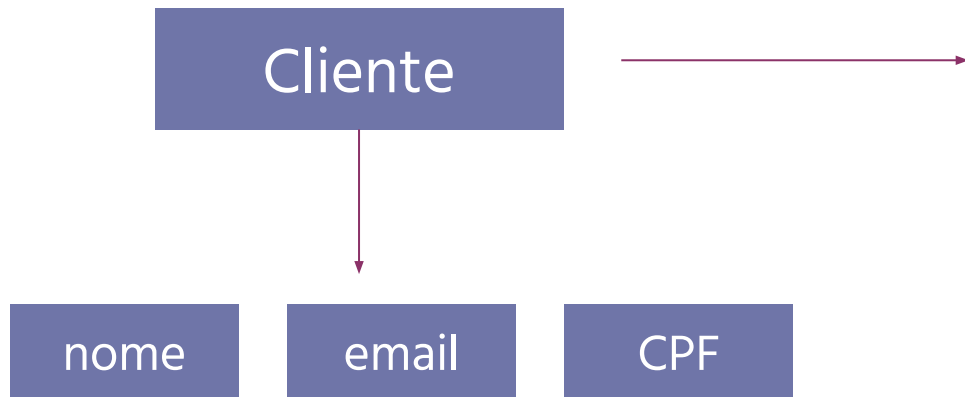


Objetos

Objetos são instâncias de uma classe.
Eles podem ser considerados os principais
atores de uma aplicação ou simplesmente
blocos de construção.



#Objetivar



#NaPratica



Praticar

Agora que você conhece POO, qual seria a melhor forma de **modelar** esses campos iniciais?

Crie o **programa** com os itens citados e **cadastre** *três clientes*.

A Empresa **FoneJá** gostaria de um programa onde eles pudessem cadastrar os clientes que comprassem celulares em suas lojas credenciadas pelo País.

Os campos iniciais são:

- **Nome**
- **CPF**
- **Data da compra**
- **Valor**
- **Modelo do Celular**
- **Ano do celular**
- **Cor do celular**
- **Loja**
- **Nome da atendente**



Praticar

O programa será uma Lista de Afazeres.

Primeiro passo:

- **Classe** chamada **AfazeresLista**
- **Array** de **afazeres** que **receberá os afazeres**.
- **Método** de **adicionar** afazeres.
- **Instancie** a **classe** em uma **const**.

Segundo passo:

- **index.html**: **botão** com o **id** **identificador "novo-botao"**.
- **index.js**, com o *getElementId*, insira o **id** **"novo-botão"** passando o *onClick* onde vai **receber** uma **função** que **chamará** a *lista* **instanciada** e o *método* que **adiciona os afazeres**.



#NaPratica



Vamos discutir sobre o
que aprendemos e
tirarmos as dúvidas?

É **sempre bom**
lembrar que **não**
existem dúvidas
“*simples*” ou “*óbvias*”.



Escreva perguntas **sem medo**



Dicas de sites para praticar lógica de Programação:

Código Livre Camp

O que o Javascript é capaz de fazer

Rocketseat

Introdução ao Javascript

Dicas de plataformas para reforçar o que você aprendeu e aprender muito mais!

Todos os sites são 100% gratuitos.



JavaScript IV



Vamos discutir sobre o
que aprendemos e
tirarmos as dúvidas?

É sempre bom
lembrar que **não**
existem dúvidas
“simples” ou “óbvias”.



Escreva perguntas **sem medo**



Vamos fazer alguns combinados!?

- Não ter vergonha de perguntar
- Erro faz parte da aprendizagem





Revisar

<http://donthpad.com/reprograma/javascript-IV>



Praticar

A empresa PhoneJá percebeu que era necessário inserir uma opção de **descontos**, de acordo com a tabela:

- Até R\$ 1.500,00 - 5%
- De R\$ 1.500,00 a R\$ 3.000,00 - 7%
- A acima de R\$ 3.000,00 - 10%



Organizar

```
export class nomeClasse { ...}
```



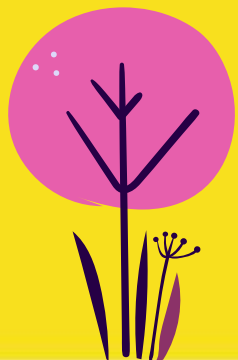
JavaScript IV



Vamos fazer alguns combinados!?

- Não ter vergonha de perguntar
- Erro faz parte da aprendizagem





ES6

Breve História

Versões

Babel





Gerenciador de Pacotes de Node





```
1 {  
2   "name": "npndemo",  
3   "version": "1.0.0",  
4   "description": "npm init demo Package",  
5   "main": "index.js",  
6   "scripts": {  
7     "test": "echo \\Error: no test specified\\ &&  
8   },  
9   "author": "Geeks Read",  
10  "license": "ISC",  
11  "dependencies": {  
12    "express": "^4.16.3",  
13    "knock-knock-jokes": "^1.7.0"  
14  }  
15 }  
16
```

package.json

```
>npm init  
>npm init --save  
>npm install express --save
```




Gerenciador de Pacotes de Node



Transpilador Javascript

#NaPratica





A minha **mãe** disse que
eu tenho o **sobrenome**
da minha **tataravó**!

Amada,
o nome disso na
POO é **herança**!

Praticar

O programa será uma Lista de Afazeres.

Primeiro passo:

- **Classe** chamada **AfazeresLista**
- **Array** de **afazeres** que **receberá os afazeres**.
- **Método** de **adicionar** afazeres.
- **Instancie** a **classe** em uma **const**.

Segundo passo:

- **index.html**: **botão** com o **id** **identificador "novo-botao"**.
- **index.js**, com o *getElementId*, insira o **id** **"novo-botão"** passando o *onClick* onde vai **receber** uma **função** que **chamará** a *lista instanciada* e o *método que adiciona os afazeres*.



Herança

Uma classe pode **herdar**
características de outra classe.



#NaPratica





Projeto Final



Vamos discutir sobre o
que aprendemos e
tirarmos as dúvidas?

É **sempre bom**
lembrar que **não**
existem dúvidas
“*simples*” ou “*óbvias*”.



Escreva perguntas **sem medo**



Credits

This is where you give credit to the ones who are part of this project.

- Presentation template by [Slidesgo](#)
- Icons by [Flaticon](#)
- Infographics by [Freepik](#)
- Images created by [Freepik](#)
- Author introduction slide photo created by [Freepik](#)
- Image slide photo created by [Freepik](#)

