

Técnico em Desenvolvimento de Sistemas



**Disciplina de
Fundamentos de Programação Orientada a Objetos**

INSTRUTORES

FUNDAMENTOS DE PROGRAMAÇÃO ORIENTADA A OBJETO



Rodrigo Alves Nunes

rodrigo.alves17@senaisp.edu.org.br

Matheus Souza de Oliveira

matheus.oliveira238@senaisp.edu.br

FUND. DE PROG. ORIENTADA A OBJETO



O QUE APRENDEMOS ATÉ AGORA NA LINGUAGEM PYTHON

- Variáveis
- Constantes
- Operadores
 - Lógicos
 - Aritméticos
 - Relacionais
- Estrutura de decisão
- Funções

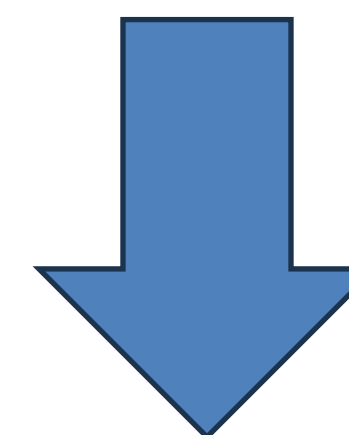


FUND. DE PROG. ORIENTADA A OBJETO



O QUE APRENDEMOS ATÉ AGORA NA LINGUAGEM PYTHON

- Variáveis
- Constantes
- Operadores
 - Lógicos
 - Aritméticos
 - Relacionais
- Estrutura de decisão
- Funções



O QUE O JAVASCRIPT

- É uma linguagem de programação criada em 1995 por Brendan Eich para ser usada pelo Netscape;
- Seu nome oficial é ECMAScript;
- De alto nível e interpretada;
- Multiparadigma
 - Imperativo
 - Orientado a objetos (por meio de protótipos)
 - Orientada a eventos;
- É a principal linguagem de programação usada pelos navegadores;
- Junto com HTML e CSS é possível tornar as páginas dinâmicas.
- Também pode ser usada fora dos navegadores com por meio de runtimes como o **Node.js**



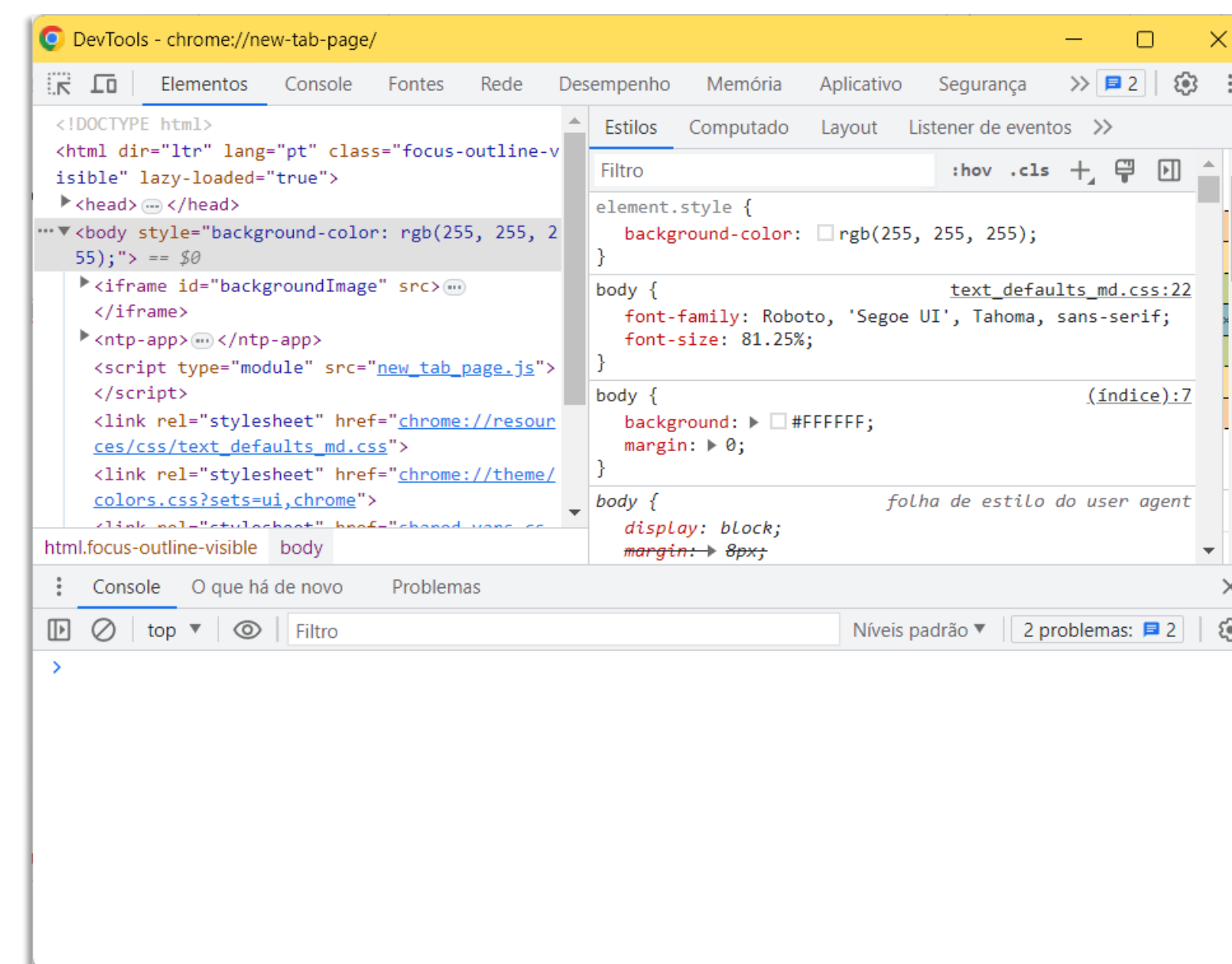
FUND. DE PROG. ORIENTADA A OBJETO



Tipos de dados no Javascript

- Number -> 1, 2, 3, 4, 5.5, 6
- String -> "texto" ou 'texto'
- Boolean -> true ou false
- Null -> valor vazio (**null** em minúsculo)
- Undefined -> valor não definido

Console do navegador (F12)



Como de escobir o tipo de uma variavel

typeof <variavel>

Como descobrir o tipo de uma variavel

O operador **typeof** retorna uma string indicando o tipo do valor do operando

```
let a
typeof a

a = 10;
typeof a

a = 5.3
typeof a

a = "Batata"
typeof a

a = 'Rúcula'
typeof a
```

```
a = true
typeof a

a = false
typeof a

a = null
typeof a

typeof "10";
typeof '10';
typeof 10 / 3;
typeof (10 / 3);
```

OPERADORES	
Operadores aritméticos	+ - * / %
Operadores de agrupamento	()
Operadores de atribuição	= += -= /=
Incrementar	++
Decrementar	--
Concatenação de textos (Strings)	+

OPERADORES RELACIONAIS	
Menor que	<
Maior que	>
Menor ou igual	<=
Maior ou igual	>=
Igualdade	==
Desigualdade	!=
Igualdade restrita (tipo)	===
Desigualdade restrita (tipo)	!==

FUND. DE PROG. ORIENTADA A OBJETO



OPERADORES LÓGICOS	
OU lógico	
E lógico	&&
NÃO lógico	!

Operador OU 		
A	B	A B
F	F	F
F	V	V
V	F	V
V	V	V

Operador E &&		
A	B	A && B
F	F	F
F	V	F
V	F	F
V	V	V

Operador NÃO !	
A	!A
F	V
V	F

Ex1: Somente irei viajar se meu **carro está na garagem** e com o **tanque cheio**

Ex2: Somente irei ao cinema somente se eu **possuir dinheiro** ou **ganhar o um ingresso**

Operador E &&		
A	B	A&&B
F	F	F
F	V	F
V	F	F
V	V	V

Ex1: Somente irei viajar se meu **carro está na garagem** e com o **tanque cheio**

```
ireiViajar = (carroNaGaragem == true) && (tanqueEstaCheio == true)
```

Operador E &&		
A	B	A&&B
F	F	F
F	V	F
V	F	F
V	V	V

Ex2: Somente irei ao cinema se eu **possuir dinheiro** ou **ganhar o um ingresso**

```
ireiAoCinema = (possoDinheiro == true) || (tenhoIngresso == true)
```


Como criar comentários

```
// Comentário de uma linha
```

```
/* Comentário de  
   múltiplas linhas */
```

FUND. DE PROG. ORIENTADA A OBJETO



Funções de apoio para entrada e saída de dados nos navegadores

- `prompt()`

<https://developer.mozilla.org/pt-BR/docs/Web/API/window/prompt>

- `alert()`

<https://developer.mozilla.org/pt-BR/docs/Web/API/Window/alert>

- `confirm()`

<https://developer.mozilla.org/pt-BR/docs/Web/API/Window/confirm>

Funções de apoio para entrada e saída de dados nos navegadores

- `console.log()`

<https://developer.mozilla.org/pt-BR/docs/Web/API/console/log>

- `console.warn()`

<https://developer.mozilla.org/pt-BR/docs/Web/API/console/warn>

- `console.error()`

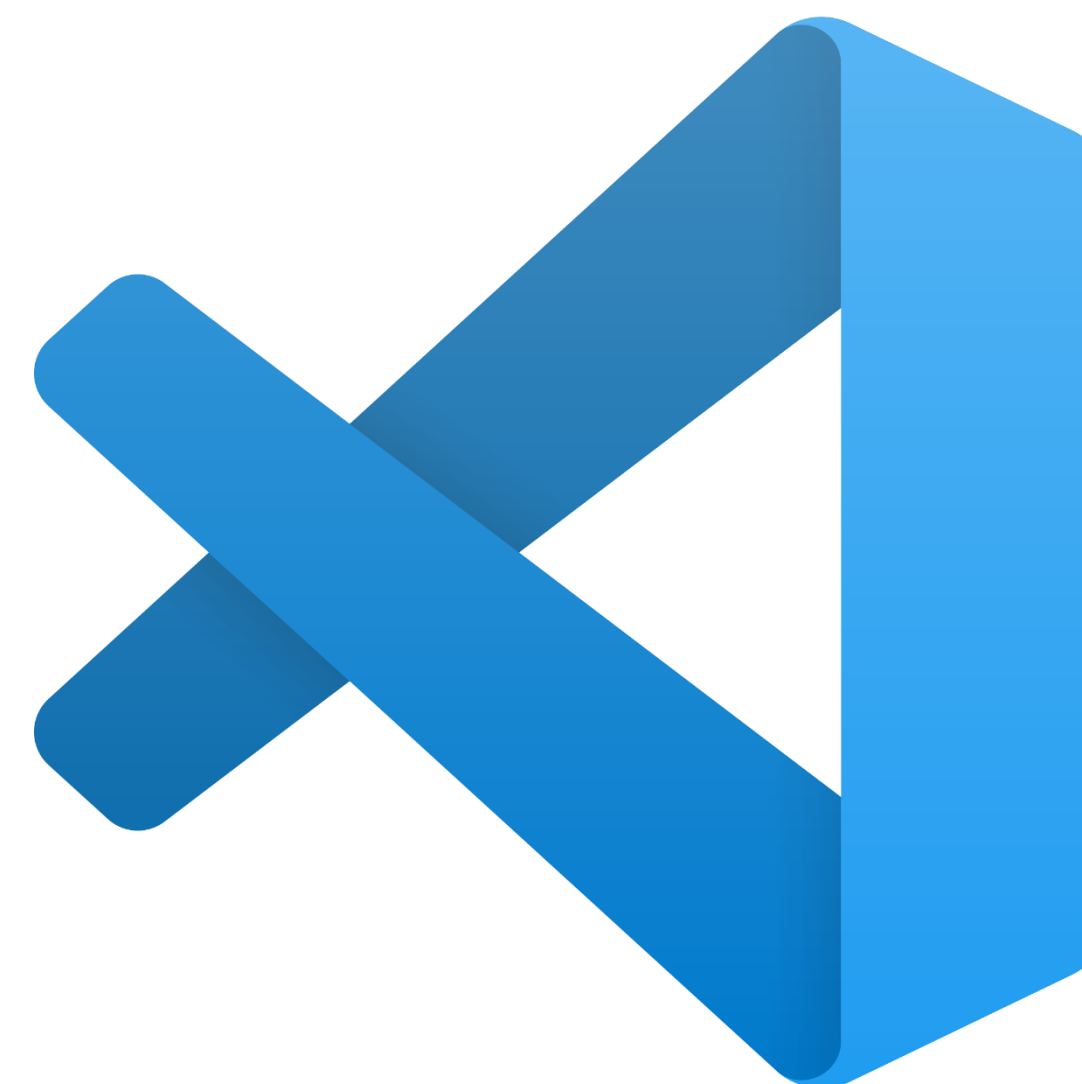
<https://developer.mozilla.org/pt-BR/docs/Web/API/console/error>

FUND. DE PROG. ORIENTADA A OBJETO



Vamos para o VS CODE

- Criar uma pasta e abrir o VS CODE na pasta
- Criar um arquivo HTML
- Criar um arquivo JS
- Anexar o arquivo no HTML

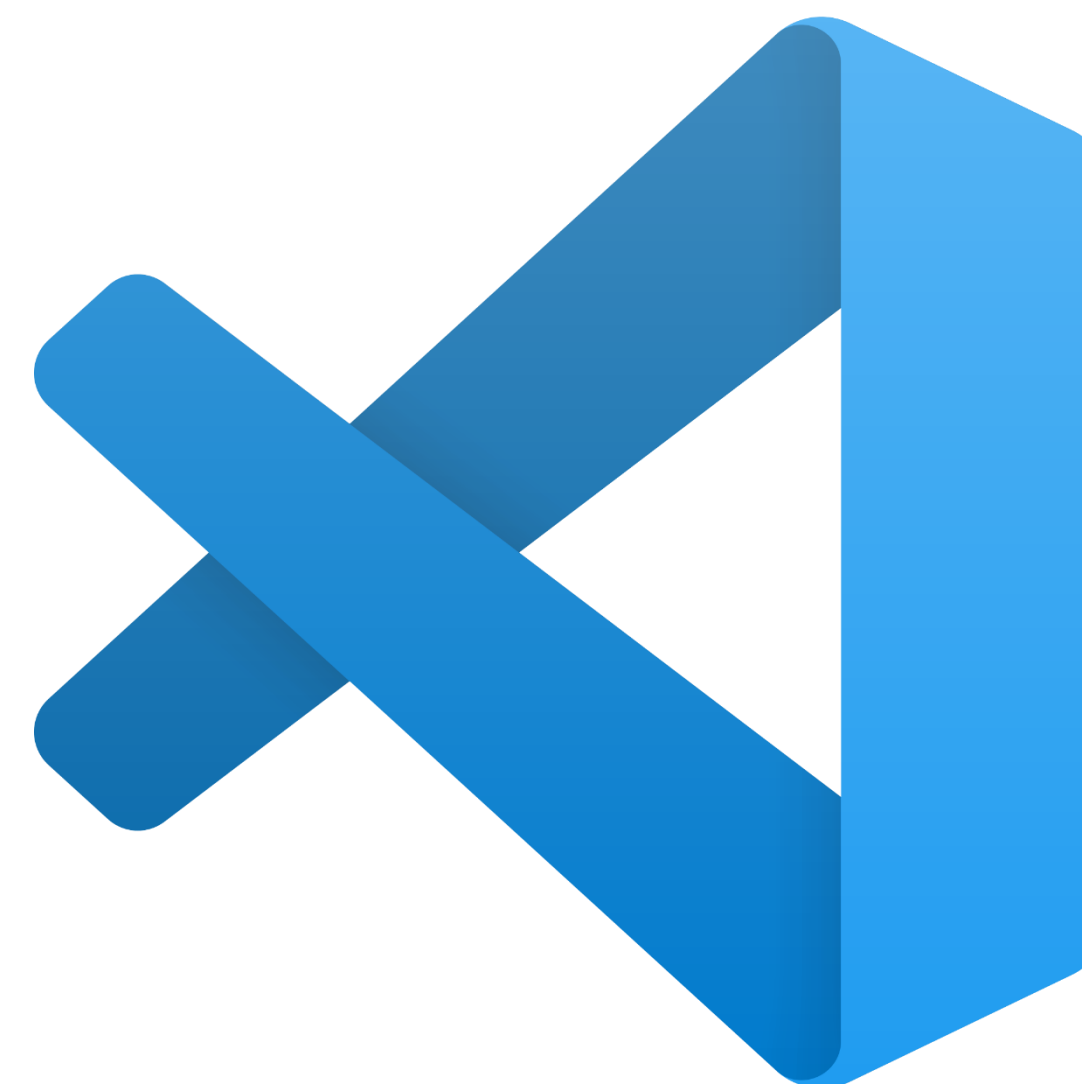


FUND. DE PROG. ORIENTADA A OBJETO



Vamos para o VS CODE

- Vamos montar um programa para capturar nome, sobrenome, idade e outros dados e gerar saídas.
- Use os comandos `prompt()` e `alert()`.



Atenção para as conversões automáticas

```
let a = '10'  
let b = 10  
let c = 10.0
```

```
a == b  
a == c  
b == c
```

?

```
a === b  
a === c  
b === c
```

?

```
a + b  
b + c
```

?

FUND. DE PROG. ORIENTADA A OBJETO



Vamos para o VS CODE



Escreva um programa que peça dois números ao usuário e exiba a soma dos mesmos.

Use os comandos **prompt()** e **alert()**.



CONVERSÕES DE TIPOS

```
let inteiro = parseInt("10")  
let real = parseFloat("1.3")
```


CONDICIONAL

```
if (condicao) {  
    // código para executar caso a condição seja verdadeira  
} else {  
    // senão, executar este código  
}
```

FUND. DE PROG. ORIENTADA A OBJETO



Vamos juntos para o VS CODE



Vamos montar um algoritmo que leia um valor e informe “**O número X é maior que 10**” caso o número seja maior que 10 e caso não seja informe “**O número X não é maior que 10**”.

Use os comandos **prompt()** e **alert()**.



FUND. DE PROG. ORIENTADA A OBJETO



Vamos para o VS CODE



Crie um algoritmo que receba três números do usuário e determine o maior entre eles.

Use os comandos **prompt()** e **alert()**.



FUND. DE PROG. ORIENTADA A OBJETO



Vamos para o VS CODE



Crie um algoritmo que receba três números do usuário e determine o maior entre eles.

Use os comandos **prompt()** e **alert()**.



FUNÇÕES

```
function nome(parametro1, parametro2){  
    return valorRetorno;  
}
```

```
function somar(a, b){  
    return a + b;  
}
```

Estrutura de repetição

WHILE ou **ENQUANTO**

```
while (condition) {  
  
}
```

Exemplo

```
let contar = 0;  
  
while (contar < 10) {  
  console.log(contar);  
  contar++;  
}
```


Estrutura de repetição

FOR - PARA

```
for (let indice = 0; indice < 10; indice++) {  
    console.log(indice)  
}
```

Template literals (Template strings)

```
let num1 = 10;  
let num2 = 20;  
  
let resultado = num1 + num2;  
  
let mensagem = `A soma de ${num1} e ${num2} é ${resultado}`;  
  
alert(mensagem);
```