



Slides - Encontro 1

Algoritmos + Hello, World!



Objetivos do Encontro

1

Entender Algoritmos

2

Conhecer Python

3

Usar `print()` e `input()`

4

Identificar Tipos

5

Criar Primeiro Programa



🧠 O que é um Algoritmo?

Definição

- Sequência de passos
- Lógica organizada e finita
- Independe da linguagem

Exemplo: Sanduíche

1. Pegar pão
2. Passar manteiga
3. Colocar recheio
4. Fechar sanduíche
5. Servir



Por que Python?

Vantagens

- Sintaxe simples
- Muito usada
- Comunidade gigante
- Versatilidade

Filosofia Python

"Código deve ser fácil de ler e escrever"



Executando Python

Opções de Editor

- VS Code (recomendado)
- Thonny (simples)
- Terminal/cmd

Estrutura Básica

meu_programa.py

Execute: python
meu_programa.py



Hello, World!

```
# Meu primeiro programa Python!  
print("Hello, World!")
```

Pontos Importantes:

- **print()** exibe na tela
- Aspas delimitam texto (strings)
- **#** marca comentários
- Não esqueça os parênteses!



Entrada de Dados – input()

```
nome = input("Digite seu nome: ")  
print("Olá,", nome)
```

Como Funciona:

- **input()** pausa o programa
- Espera o usuário digitar
- Retorna o texto digitado
- Sempre é **string** (texto)



Saída de Dados – print()

Formas de Usar:

Texto simples

```
print("Oi!")
```

Múltiplos valores

```
print("Nome:", nome, "Idade:", idade)
```

Misturando texto e variáveis

```
print("Olá", nome, "!")
```



Dica: Use vírgulas para separar valores



Descobrimos Tipos - type()

```
nome = "João"  
idade = input("Idade: ")  
print("Nome:", type(nome)) # <class 'str'>  
print("Idade:", type(idade)) # <class 'str'>
```

Tipos Básicos:

- **str**: texto ("João")
- **int**: número inteiro (20)
- **float**: número decimal (1.75)

Pegadinhas Comuns

1. Esquecer Aspas

✗ `print(Hello)`

✓ `print("Hello")`

2. Esquecer Parênteses

✗ `print "Hello"`

✓ `print("Hello")`

3. Não Salvar input()

✗ `input("Nome: ")`
`print("Oi", nome) # erro!`

✓ `nome = input("Nome: ")`
`print("Oi", nome)`