

List of Exercises – Introduction to Python

1. Identification of Types

Enunciado:

Analyze the code below and indicate the type of each variable.

```
idade = 21
altura = 1.65
nome = "Luana"
ativo = True
```

Resposta:

- `idade` → `int` (integer number)
- `altura` → `float` (real number)
- `nome` → `str` (string of characters)
- `ativo` → `bool` (logical value)

2. Case Sensitive

Enunciado:

Explain what happens in the code below and why the result may cause an error.

```
nome = "Ana"
Nome = "Beatriz"
print(nome, Nome, NOME)
```

Resposta:

Python **distinguishes between uppercase and lowercase letters** ("case sensitive").

The variables `nome` and `Nome` exist separately, but `NOME` was not declared — this generates a **type error** `NameError`.

3. Dynamic Typing

Enunciado:

What happens when we run the following code?

```
x = 10
x = "dez"
```

```
print(x)
```

Resposta:

Python permite **alterar o tipo de uma variável durante a execução**, pois usa **tipagem dinâmica**.

Assim, `x` passa de `int` para `str` e o resultado impresso será "dez".

4. Variáveis Semânticas

Enunciado:

Reescreva os nomes das variáveis abaixo seguindo o padrão **snake_case** e boas práticas:

ValorTotal = 250

NomeDoCliente = "Carla"

DataDeNascimento = "2002-04-17"

Resposta:

valor_total = 250

nome_do_cliente = "Carla"

data_de_nascimento = "2002-04-17"

5. Constantes em Python

Enunciado:

Python não possui constantes nativas, mas é possível simulá-las.

Crie uma constante que represente o valor de **PI** e mostre seu uso.

Resposta:

PI = 3.14159

raio = 5

area = PI * raio**2

print(area)

Por convenção, **nomes em maiúsculas** indicam constantes.

Alternativamente, pode-se usar `math.pi` importando a biblioteca `math`.

6. Interpretação e Execução

Enunciado:

Explique a diferença entre uma **linguagem interpretada** e uma **linguagem compilada**.

Por que o Python se enquadra na primeira categoria?

Resposta:

- Uma **linguagem compilada** precisa converter todo o código em linguagem de máquina antes da execução.
- Já uma **linguagem interpretada**, como Python, **executa linha por linha**, traduzindo o código no momento em que roda.
Isso permite testes e desenvolvimento mais rápidos, mas pode ser ligeiramente mais lento na execução final.