

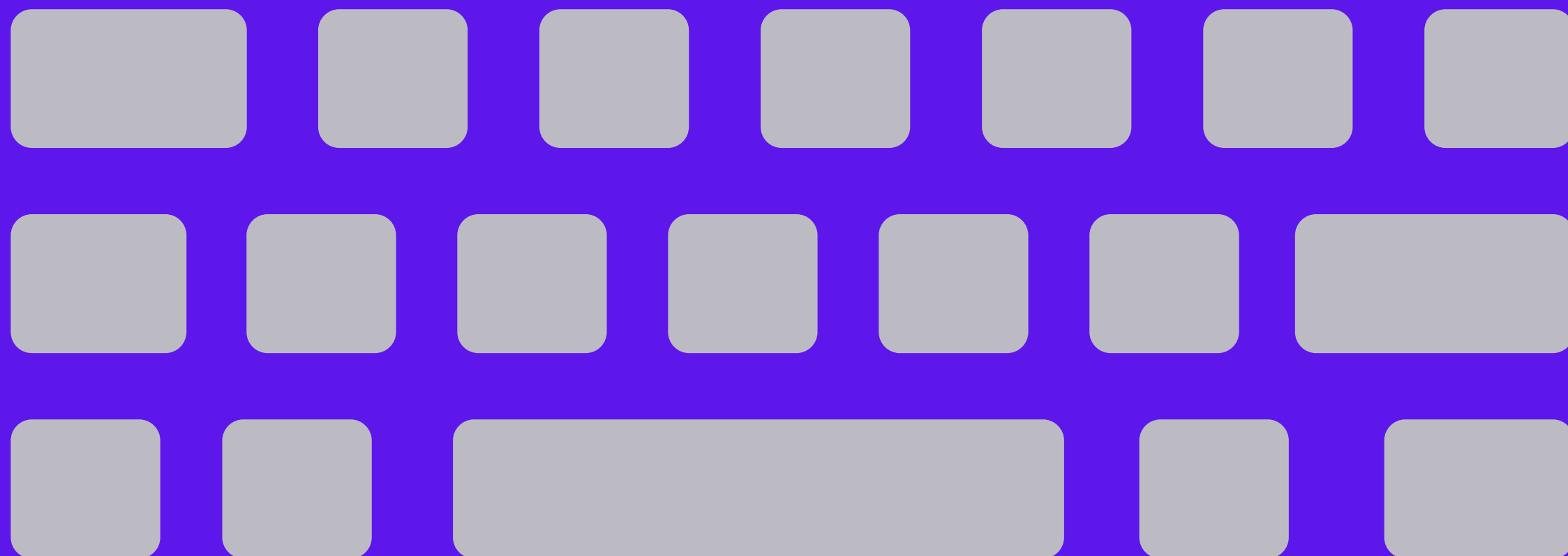


# PROGRAMAÇÃO 101

# Dia 1

- Introdução
- Diferentes Áreas de Especialização
- Linguagem de Baixo e Alto Nível
- Aplicações do Python
- Por que Aprender Python?
- Programação 101

# Programação no Geral



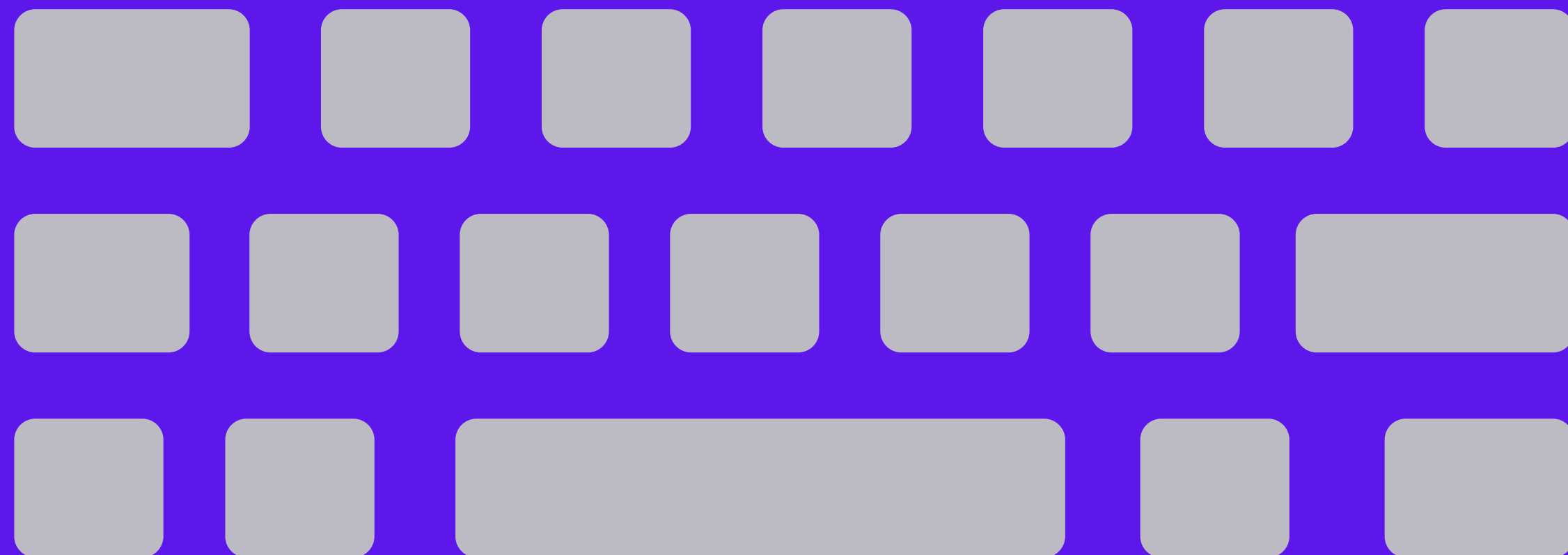
# Definição e Importância

A programação é a arte de criar instruções para um computador realizar tarefas específicas. É a linguagem que permite aos humanos interagirem com máquinas, transformando ideias em software funcional. A importância da programação transcende a simples criação de software, estendendo-se a diversos setores da sociedade.

# Aplicações em Diversas Áreas

- Desenvolvimento Web:
  - Construção de sites interativos e dinâmicos.
  - Tecnologias como HTML, CSS, JavaScript e frameworks como Django, Flask (Python).
- Ciência de Dados e Inteligência Artificial:
  - Análise de dados para tomada de decisões.
  - Machine Learning e Deep Learning para criar modelos preditivos.
- Automação de Tarefas:
  - Automação de processos repetitivos para aumentar eficiência.
  - Uso de scripts para simplificar tarefas diárias.

# Linguagem de Baixo e Alto Nível em Programação



# Tipos de linguagem

## Linguagem de Baixo Nível:

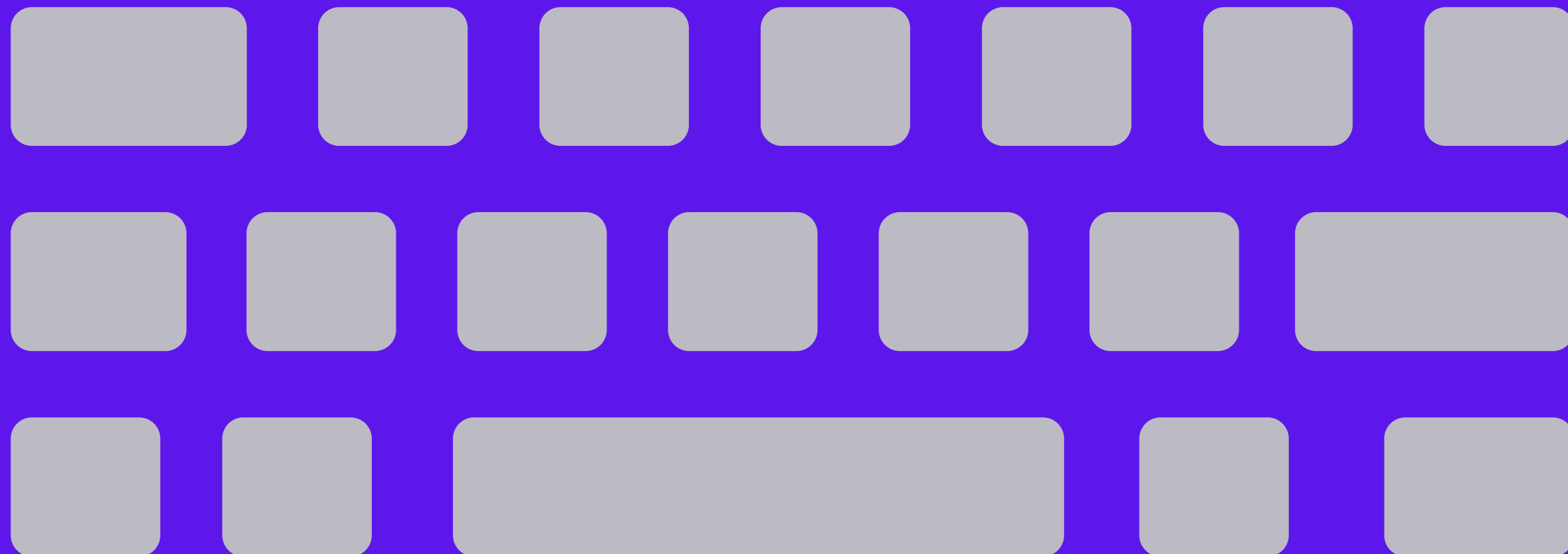
- Próxima ao Hardware
- Controle Fino
- Complexidade e Esforço
- Maior rapidez

## Linguagem de Alto Nível:

- Abstração do Hardware
- Facilidade de Leitura e Escrita
- Automatização de Tarefas
- Portabilidade
- Demoram mais a executar

# Aplicações do Python

```
graph TD; Title[Aplicações do Python] --- Grid[Grid of 21 placeholder boxes];
```

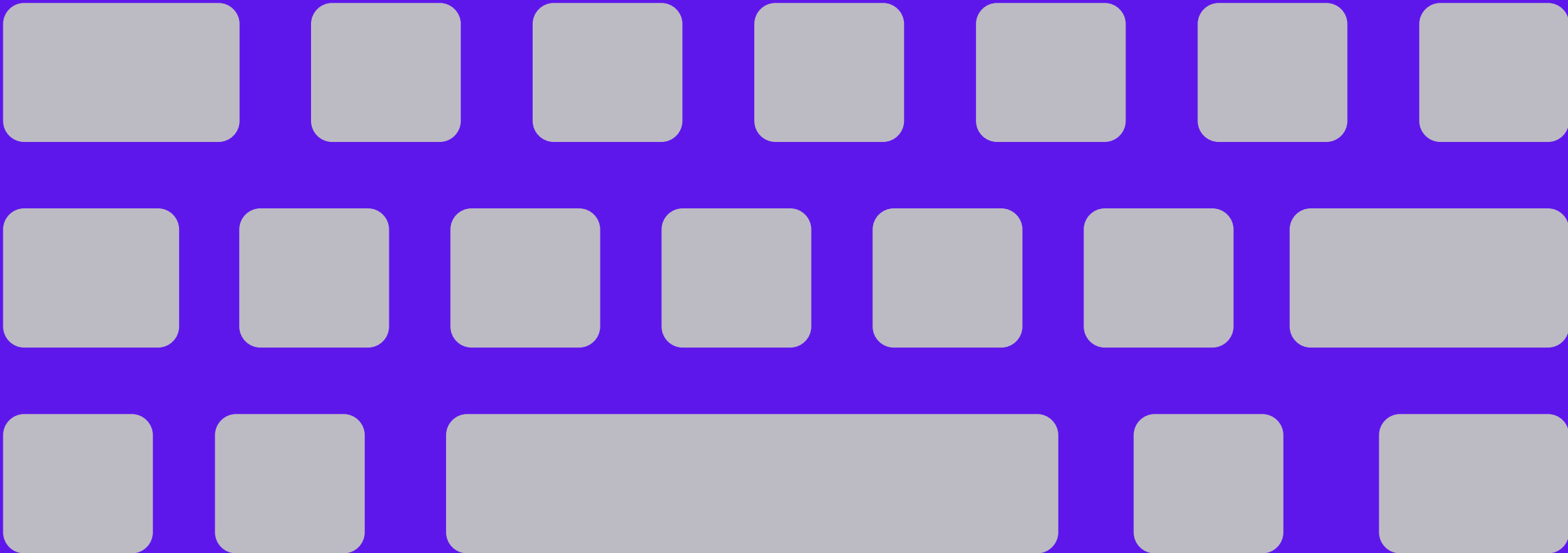




# Aplicações do Python

- Scripting
- Desenvolvimento de Aplicações
- Análise de Dados e Machine Learning
- Automação de Tarefas
- ...

# Por que Aprender Python?



# Por que Aprender Python?

- 1. Sintaxe Clara e Legível:
  - Definição:
    - Python é conhecido por sua sintaxe clara e intuitiva, que se assemelha à linguagem natural.
  - Vantagens:
    - Facilita a leitura e escrita de código.
    - Reduz a probabilidade de erros.



# Por que Aprender Python?

- 2. Grande Comunidade de Desenvolvedores:
  - Definição:
    - Python possui uma vasta comunidade global de desenvolvedores ativos.
  - Vantagens:
    - Suporte e colaboração com outros programadores.
    - Acesso a recursos e soluções prontas.

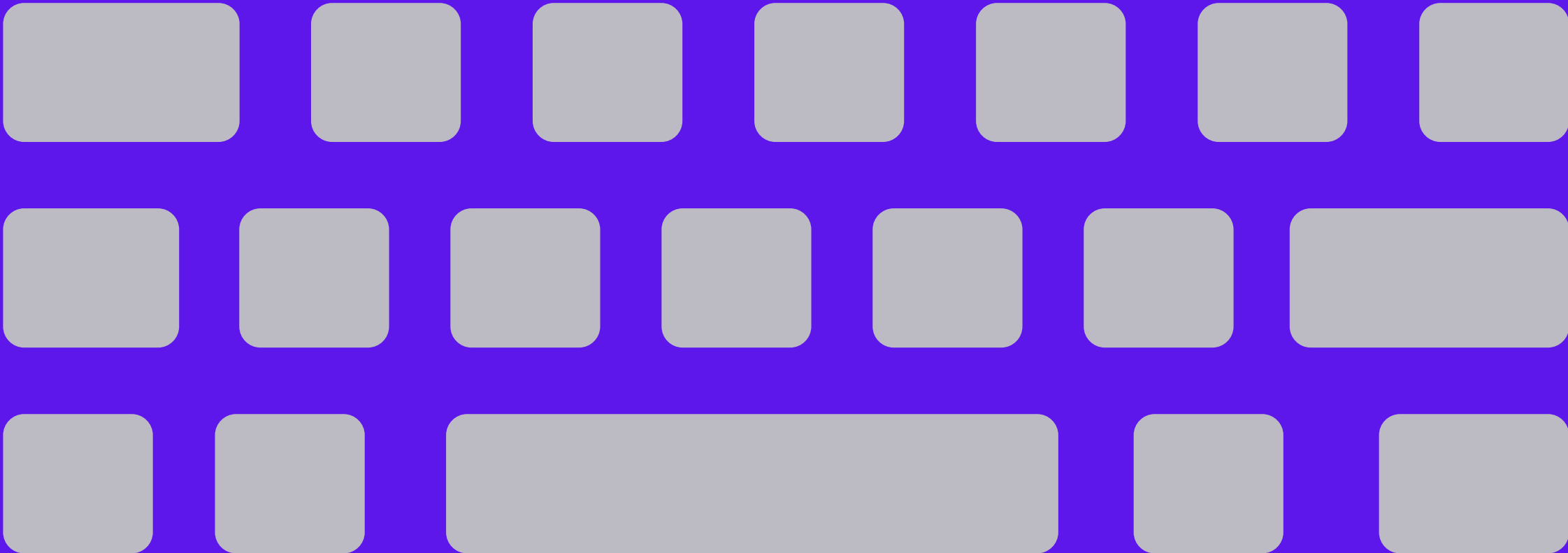
# Por que Aprender Python?

- 3. Ampla Gama de Bibliotecas e Frameworks:
  - Definição:
    - Python oferece uma extensa variedade de bibliotecas e frameworks prontos para uso.
  - Vantagens:
    - Acelera o desenvolvimento de aplicações.
    - Cobertura de diversas áreas, como web, machine learning, ciência de dados.

# Por que Aprender Python?

- 4. Versatilidade e Facilidade de Aprendizado:
  - Definição:
    - Python é versátil, adequado para uma ampla gama de aplicações, e é conhecido por ser amigável para iniciantes.
  - Vantagens:
    - Pode ser utilizado em diferentes domínios (web, automação, ciência de dados).
    - Facilidade de aprendizado, tornando-o ideal para iniciantes.

# Programação 101



# Por que Aprender Python?

- 5 dias de teoria e prática
- Uma visão geral sobre programação
- Aprender o básico de programação
- E muito mais!

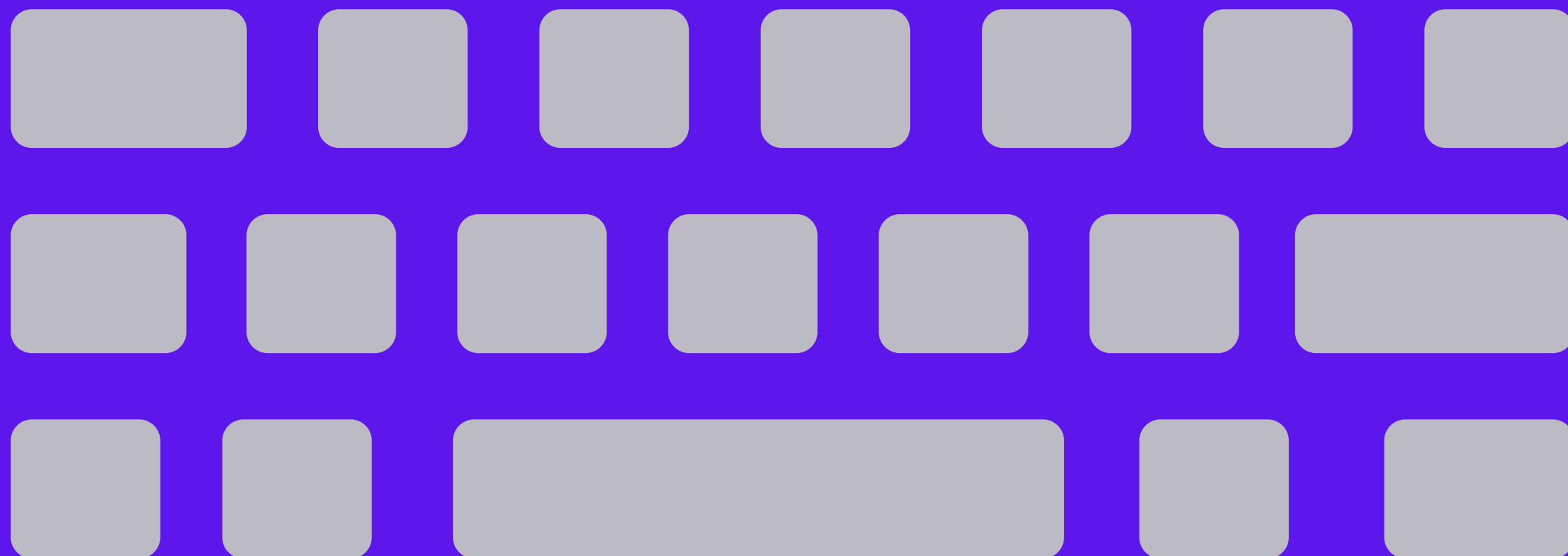




# Dia 1

- Código
- Variáveis
- Convertendo tipos
- Testes

# O que é um Código?

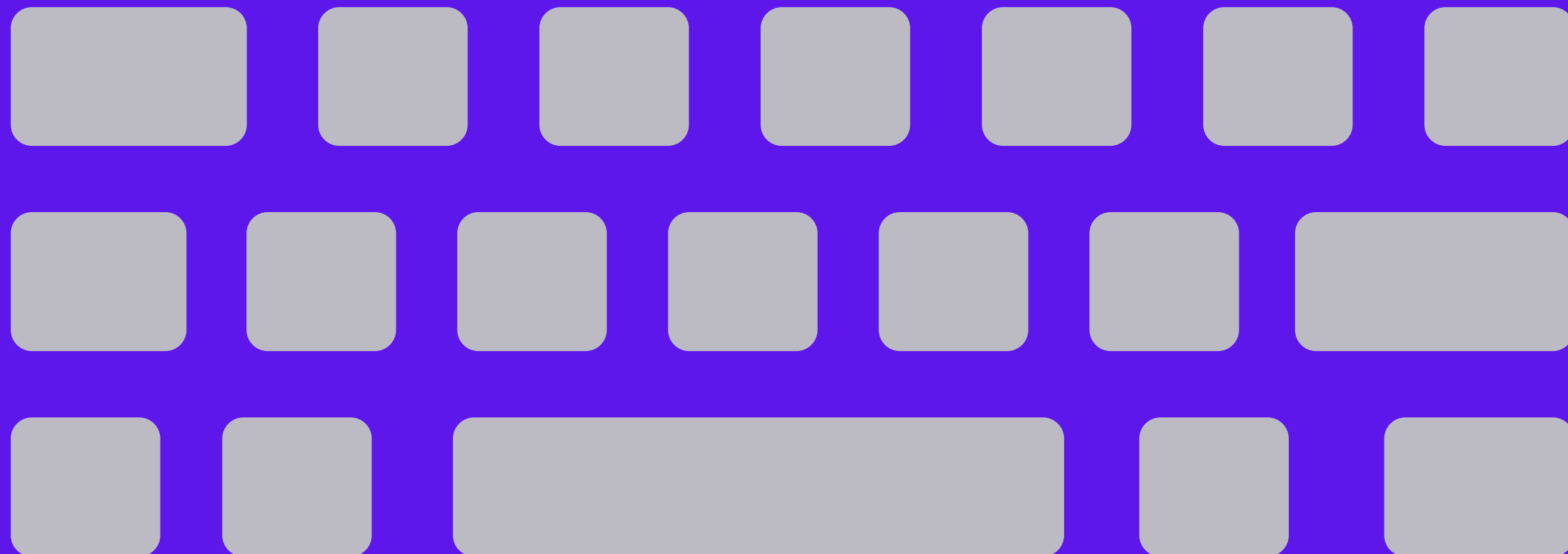


# Definição de Código

Um código é um conjunto de instruções que são dadas ao computador. Essas instruções são escritas por meio de uma linguagem de programação, no caso o Python.

Quando executamos esse código, as instruções serão realizadas de maneira sequencial.

# O que são Variáveis?

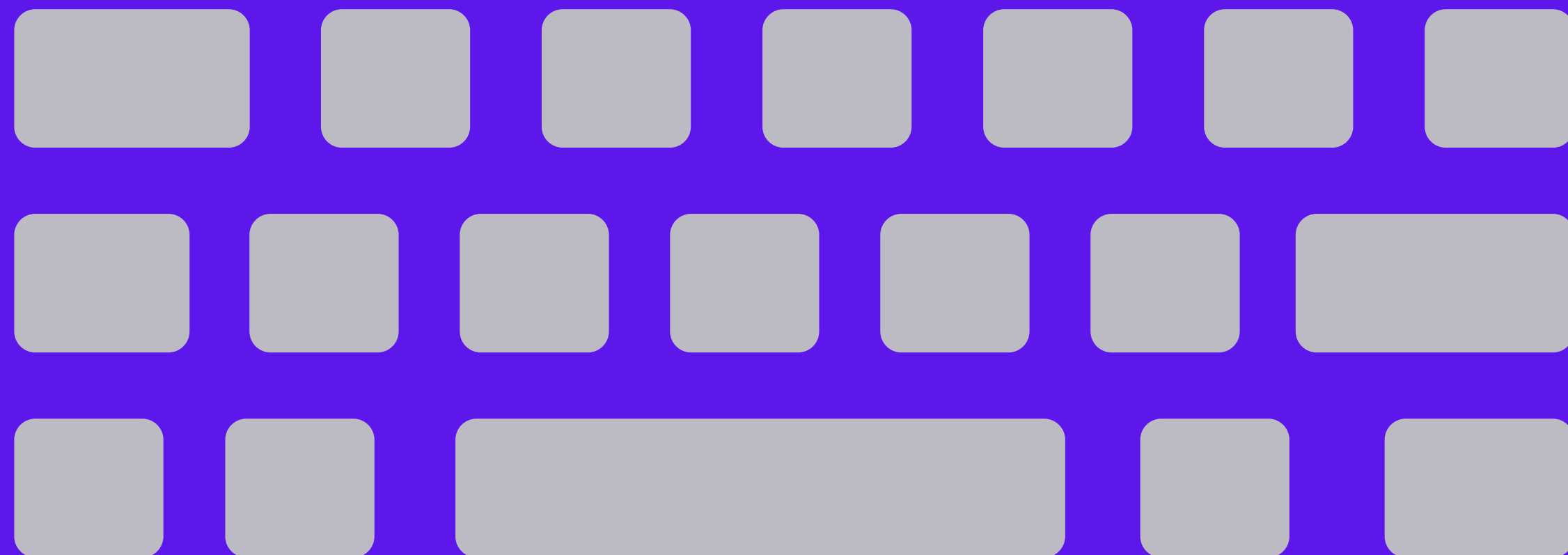


# Definição e Importância

Variáveis são espaços de armazenamento que contêm dados referenciados por um nome.

```
nome = "John"  
idade = 25  
altura = 1.75
```

# Tipos de Variáveis em Python:



# Tipos de Variáveis

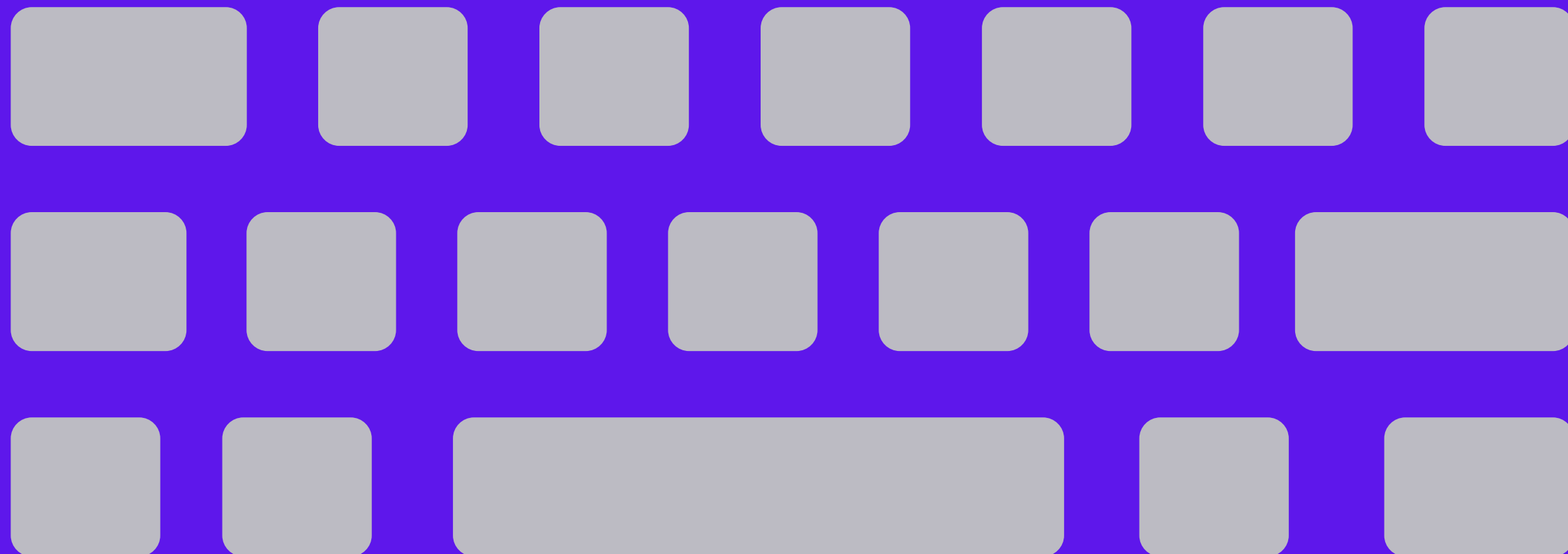
## Tipos Básicos:

- int: Números inteiros.
- float: Números de ponto flutuante.
- str: Sequências de caracteres.
- bool: Valores lógicos (True/False).

```
numero_inteiro = 42  
numero_decimal = 3.14  
texto = "Olá, Python!"  
condicao = True
```

# Métodos para Conferir o Tipo da Variável

```
graph TD; Title[Métodos para Conferir o Tipo da Variável] --- Grid[Grid of 21 boxes];
```





# type()

Função type():

- Retorna o tipo da variável.

```
tipo_numero = type(42)
tipo_texto = type("Olá")
```

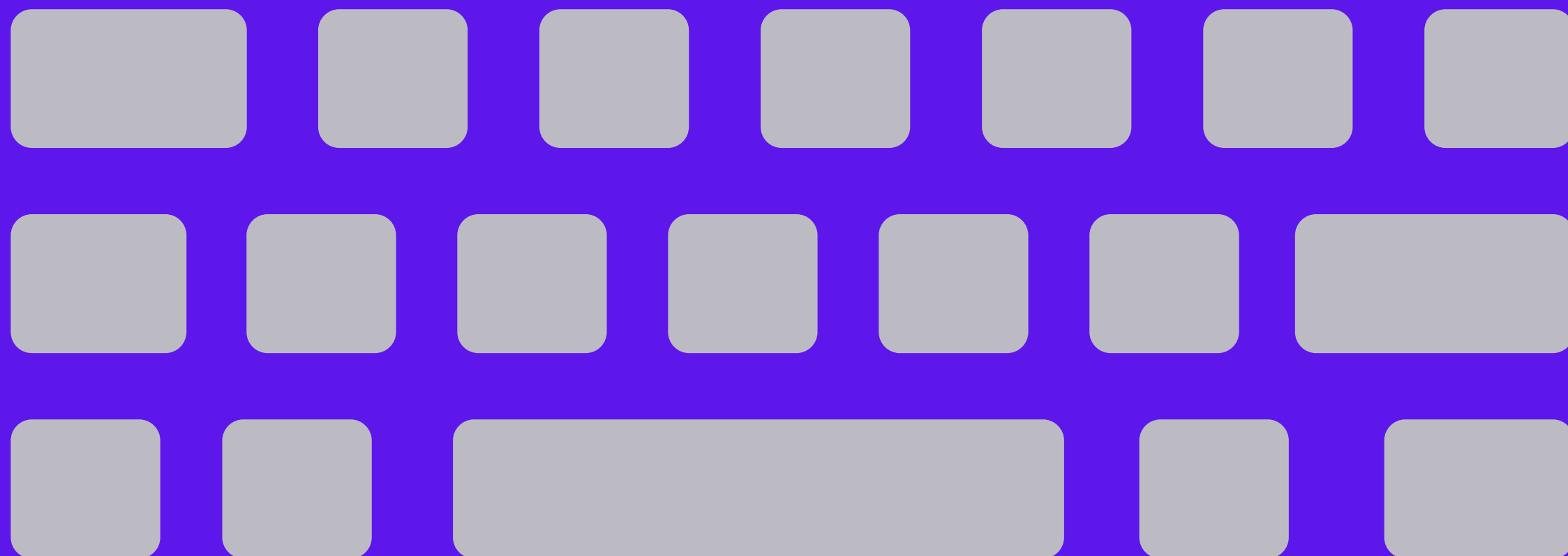
# isinstance()

Método isinstance():

- Verifica se a variável é de um tipo específico.

```
resultado = isinstance(numero_inteiro, int)
```

# Conversão de Variáveis



# Conversão de Variáveis

## Funções de Conversão:

- `int()`: Converte para inteiro.
- `float()`: Converte para ponto flutuante.
- `str()`: Converte para string.

```
numero_como_string = "42"  
numero_convertido = int(numero_como_string)
```

# Dia 1

Operações em Python

# Operadores em Python

Operação	Operador	Exemplo	Resultado
Soma	<b>+</b>	9 + 7	<b>16</b>
Subtração	<b>-</b>	9 - 7	<b>2</b>
Multiplicação	<b>*</b>	9 * 7	<b>63</b>
Divisão	<b>/</b>	5 / 2	<b>2.5</b>
Exponenciação	<b>**</b>	9 ** 2	<b>81</b>
Módulo	<b>%</b>	9 % 2	<b>1</b>

# Realizando operações

- Utilizando os operadores em inteiros e decimais

```
resultado1 = 10 + 2  
resultado2 = 13.5 - 1.5
```

- Utilizando os operadores em variáveis

```
total = resultado1 * resultado2
```

# Dia 1

Interagindo com o  
Usuário em Python



# Impressão de Valores

Função `print()`:

- Exibe valores na saída padrão (console).

```
idade = 25  
print("Idade:", idade)
```

Função `input()`:

- Permite a leitura de mensagens inseridas pelo usuário.

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

# Dia 1

Comentários de código

# Comentários de código

Para criar comentários em seu código, ou seja, partes do código que o usuário vê mas o programa não lê, basta utilizar “#”

```
#Esse é meu código abaixo:  
  
print("Olá Mundo")  
  
# Ao usar a hashtag você comenta a linha inteira <3
```





# PROGRAMAÇÃO 101