

## Planejamento de Aula

**Disciplina:** Programação Aplicada à Ciência de Dados

**Turma:** 43 alunos (15 a 17 anos)

**Carga horária:** 200 minutos (100 minutos em cada dia, com intervalo de 24 horas)

**Recursos:** Quadro branco, datashow pequeno, notebook individual, Jupyter

Notebook/Anaconda instalado

## Objetivos Gerais

- Compreender e aplicar funções em Python, com ênfase em parâmetros-padrão e a função embutida `enumerate`.
- Desenvolver lógica de programação para manipulação e análise de dados.
- Praticar resolução de problemas reais usando listas, iteração e funções em Python.
- Fixar conceitos por meio de exercícios práticos e atividades colaborativas.

## Dia 1 (100 minutos)

### 1. Acolhida e Contextualização (10 min)

- Apresentação dos objetivos da aula.
- Discussão breve sobre a importância da programação para a Ciência de Dados e o papel das funções para automação de tarefas.

### 2. Exposição: Funções e Parâmetros-Padrão (20 min)

- Explicação teórica sobre funções em Python, parâmetros obrigatórios e parâmetros-padrão (opcionais) <sup>[1]</sup>.
- Exemplos práticos no quadro e no Jupyter Notebook:

```
def saudacao(nome, mensagem='Olá'):  
    print(f'{mensagem}, {nome}!')  
saudacao('João')  
saudacao('Maria', 'Oi')
```

- Discussão: Por que parâmetros-padrão tornam funções mais flexíveis?

### 3. Atividade Prática 1: Funções com Parâmetros-Padrão (20 min)

- Proposta: Criar funções com parâmetros-padrão para diferentes contextos (saudação, cálculo de desconto, etc.).
- Orientação individual: cada aluno desenvolve e testa pelo menos uma função com parâmetro-padrão em seu notebook.
- Compartilhamento rápido de exemplos no datashow.

### 4. Exposição: Estruturas de Controle de Fluxo e Enumeração (15 min)

- Revisão de laços `for` e introdução ao conceito de enumeração de iteráveis.
- Apresentação da função embutida `enumerate` e sua sintaxe:

```
for indice, valor in enumerate(lista, start=0):  
    print(indice, valor)
```

- Exemplos com listas, strings e tuplas<sup>[1]</sup>.

### 5. Atividade Prática 2: Aplicando `enumerate` (20 min)

- Exercícios guiados:
  - Imprimir itens de uma lista de conquistas com seus índices.
  - Utilizar `enumerate` para exibir índices e valores de diferentes tipos de iteráveis (listas, strings, tuplas).
- Discussão de erros comuns (ex: tentar enumerar um inteiro).

### 6. Quiz Interativo e Revisão (15 min)

- Perguntas rápidas baseadas no material:
  - O que acontece se não fornecer valor para um parâmetro opcional?
  - Como definir um parâmetro com valor-padrão?
  - Qual a principal finalidade dos parâmetros opcionais?
- Resolução coletiva e esclarecimento de dúvidas<sup>[1]</sup>.

### Encerramento do Dia 1 (5 min)

- Recapitulação dos conceitos vistos.
- Orientação para trazer dúvidas e exemplos próprios para a próxima aula.
- Aviso: próxima aula terá foco em exercícios práticos e aplicação dos conceitos.

## Dia 2 (100 minutos)

### 1. Retomada e Esclarecimento de Dúvidas (10 min)

- Breve revisão dos principais pontos do dia anterior.
- Espaço para dúvidas e compartilhamento de dificuldades.

### 2. Atividade Prática 3: Exercícios de Fixação (30 min)

- Propostas baseadas no material didático<sup>[1]</sup>:
  - Criar função para imprimir produtos com índice.
  - Função para calcular média de notas.
  - Função para aumentar salários em 10%.
  - Utilizar `enumerate` para imprimir altura de alunos com índice.
- Alunos trabalham individualmente ou em duplas, com apoio do professor.

### 3. Atividade Prática 4: Desafios Colaborativos (30 min)

- Divisão em grupos pequenos.
- Cada grupo recebe um desafio mais elaborado, por exemplo:
  - Criar função que recebe lista de temperaturas em Celsius e retorna lista em Fahrenheit.
  - Criar função que imprime tarefas com índice e status (feito/não feito).
  - Criar função que recebe lista de palavras e imprime o número de letras de cada uma, usando `enumerate`.
- Grupos compartilham soluções e discutem abordagens.

### 4. Atividade Final: Integração de Conceitos (20 min)

- Proposta: Desenvolver uma função que receba duas listas (temperaturas em Kelvin e Fahrenheit) e imprima cada temperatura convertida para Celsius, junto com seu índice.
- Dicas no quadro:
  - Conversão Kelvin para Celsius:  $C = K - 273.15$
  - Conversão Fahrenheit para Celsius:  $C = (F - 32) \times \frac{5}{9}$
- Alunos implementam e testam suas soluções.

## **5. Socialização e Discussão (5 min)**

- Voluntários apresentam suas soluções.
- Discussão sobre diferentes formas de resolver o mesmo problema.

## **6. Fechamento e Orientações Finais (5 min)**

- Recapitulação dos principais aprendizados.
- Orientação para envio dos códigos pelo AVA.
- Indicação de materiais complementares para aprofundamento.

## **Estratégias para Engajamento e Retenção**

- Alternância entre exposição, prática individual e atividades em grupo.
- Monitoramento ativo e feedback imediato.
- Uso de exemplos do cotidiano e desafios contextualizados.
- Incentivo à colaboração e compartilhamento de soluções.

## **Referências**

- Material didático oficial: DADOS-ANO1C2B2S11.pdf (Aulas 1 a 4)
- MENEZES, N. N. C. Introdução à programação com Python: algoritmos e lógica de programação para iniciantes. Novatec, 2019.

**Este planejamento garante continuidade entre os dois dias, promove a fixação de conceitos e a aplicação prática, e está alinhado ao material didático da disciplina.**<sup>[1]</sup>

