

# FAP 2024 – Softex/UFRN

## Prof. José Alfredo F. Costa

16 de agosto de 2024

### Exercício prático:

Das 13 às 14:45 h – em duplas, desenvolvam um programa simples de gerenciamento de tarefas em Python.

O programa deve permitir ao usuário inserir, editar, listar e apagar tarefas. Cada tarefa deve conter os seguintes campos: descrição, data de início, data de fim e observação.

Use uma **lista de dicionários** para armazenar as tarefas em memória.

O programa deve ter um menu interativo que permite ao usuário escolher entre as seguintes opções:

1. Inserir nova tarefa
2. Editar tarefa existente
3. Listar todas as tarefas
4. Apagar tarefa
5. Sair do programa

A seguir, temos possíveis exemplos da simulação de entrada de dados, listagem, edição, etc. - das opções - do programa.

A entrega do projeto deve ser no GitHub.

### Simulação de uso do programa:

1. Inserir nova tarefa:

```
=== Gerenciador de Tarefas ===
1. Inserir nova tarefa
2. Editar tarefa existente
3. Listar todas as tarefas
4. Apagar tarefa
5. Sair do programa
Escolha uma opção: 1

Digite a descrição da tarefa: Desenvolver sistema de login
Digite a data de início (DD/MM/AAAA): 20/08/2024
Digite a data de fim (DD/MM/AAAA): 30/08/2024
Digite uma observação (opcional): Utilizar autenticação JWT
Tarefa adicionada com sucesso!
```

2. Listar todas as tarefas:

```
=== Gerenciador de Tarefas ===
1. Inserir nova tarefa
2. Editar tarefa existente
3. Listar todas as tarefas
4. Apagar tarefa
5. Sair do programa
Escolha uma opção: 3

Tarefa 1:
Descrição: Desenvolver sistema de login
Data de início: 20/08/2024
Data de fim: 30/08/2024
Observação: Utilizar autenticação JWT
```

### 3. Editar tarefa existente:

```
=== Gerenciador de Tarefas ===
1. Inserir nova tarefa
2. Editar tarefa existente
3. Listar todas as tarefas
4. Apagar tarefa
5. Sair do programa
Escolha uma opção: 2

Tarefa 1:
Descrição: Desenvolver sistema de login
Data de início: 20/08/2024
Data de fim: 30/08/2024
Observação: Utilizar autenticação JWT

Digite o número da tarefa que deseja editar: 1
Nova descrição (Desenvolver sistema de login): Implementar sistema de login e registro
Nova data de início (20/08/2024):
Nova data de fim (30/08/2024): 05/09/2024
Nova observação (Utilizar autenticação JWT): Utilizar autenticação JWT e implementar recuperação de senha
Tarefa atualizada com sucesso!
```

### 4. Apagar tarefa:

```
=== Gerenciador de Tarefas ===
1. Inserir nova tarefa
2. Editar tarefa existente
3. Listar todas as tarefas
4. Apagar tarefa
5. Sair do programa
Escolha uma opção: 4

Tarefa 1:
Descrição: Implementar sistema de login e registro
Data de início: 20/08/2024
```

Data de fim: 05/09/2024

Observação: Utilizar autenticação JWT e implementar recuperação de senha

Digite o número da tarefa que deseja apagar: 1

Tarefa removida com sucesso!

5. Sair do programa:

```
=== Gerenciador de Tarefas ===
1. Inserir nova tarefa
2. Editar tarefa existente
3. Listar todas as tarefas
4. Apagar tarefa
5. Sair do programa
Escolha uma opção: 5
Saindo do programa...
```

### **Forma de entrega do projeto no GitHub:**

1. Crie um novo repositório no GitHub com um nome descritivo, como "gerenciador-tarefas-python".
2. Clone o repositório para sua máquina local.
3. Adicione o arquivo Python com o código do gerenciador de tarefas ao repositório.
4. Crie um arquivo README.md na raiz do repositório com as seguintes informações:

#### **# Gerenciador de Tarefas em Python**

Este é um simples gerenciador de tarefas implementado em Python, utilizando listas e dicionários para armazenar as informações em memória.

#### **## Funcionalidades**

- Adicionar novas tarefas
- Editar tarefas existentes
- Listar todas as tarefas
- Apagar tarefas

#### **## Como usar**

1. Certifique-se de ter Python instalado em seu sistema.
2. Clone este repositório:

git clone <https://github.com/seu-usuario/gerenciador-tarefas-python.git>

3. Navegue até o diretório do projeto:

```
cd gerenciador-tarefas-python
```

```
4. Execute o programa:
```

```
python gerenciador_tarefas.py
```

```
5. Siga as instruções no menu interativo para gerenciar suas tarefas.
```

```
## Estrutura do projeto
```

```
- `gerenciador_tarefas.py`: Contém todo o código do gerenciador de tarefas.  
- `README.md`: Este arquivo, com instruções e informações sobre o projeto.
```

```
## Contribuindo
```

```
Sinta-se à vontade para abrir issues ou enviar pull requests com melhorias para o projeto.
```

```
## Licença
```

```
Este projeto está licenciado sob a MIT License - veja o arquivo [LICENSE](LICENSE) para detalhes.
```

5. Adicione um arquivo LICENSE com a licença MIT ou outra de sua escolha.
6. Faça commit das alterações e push para o repositório remoto:

```
git add .  
git commit -m "Implementação inicial do gerenciador de tarefas"  
git push origin main
```

7. (Opcional) Adicione tags ao seu repositório para facilitar a busca, como "python", "task-management", "cli-application".
8. (Opcional) Configure o GitHub Pages para exibir o README.md como uma página web do projeto.

Ao seguir esses passos, você terá um repositório bem organizado e documentado, que outros desenvolvedores poderão facilmente entender e utilizar. O README.md fornece todas as informações necessárias para que alguém possa clonar o repositório e executar o programa, além de entender sua estrutura e funcionalidades.