



## LISTA 08 – Exercício de Programação com Flask e Jinja2

Você precisa desenvolver um website usando Flask e Jinja2 que lista todos os clubes existentes na escola. O site terá as seguintes páginas:

- Página inicial: mostrar uma lista de todos os clubes.
- Clube: mostrar detalhes de um clube específico.
- Professores: mostrar uma lista de todos os professores.
- Estudantes: mostrar uma lista de todos os estudantes.

No Python:

- Crie uma classe Estudante que tenha os seguintes atributos: nome e turma.
- Crie uma classe Professor que tenha os seguintes atributos: nome e disciplina.
- Crie uma classe Clube que tenha os seguintes atributos: nome, professorResponsavel, horarioReuniao e uma lista de membros.
  - Quando for criar um objeto da classe Clube, lembre-se que o atributo professorResponsavel deve ser um objeto da classe Professor.
  - A lista de membros deve começar vazia.
  - Crie o método adicionarMembro que recebe um objeto do tipo Estudante como parâmetro e o adiciona na lista de membros.
- Crie um arquivo main.py para o seu aplicativo Flask. Crie rotas no aplicativo:
  - A rota "/" deve renderizar um template index.html que mostra uma lista de todos os clubes, o professor responsável e o horário de reunião de cada clube.
  - A rota dinâmica "/clube/<nomeDoClube>" deve renderizar um template clube.html que mostra informações detalhadas sobre o clube selecionado, incluindo todos os membros do clube. **Note que essa rota é dinâmica e depende de qual clube o usuário selecionou.**
  - A rota "/professores" deve renderizar um template professores.html que mostra uma lista de todos os professores.
  - A rota "/estudantes" deve renderizar um template estudantes.html que mostra uma lista de todos os estudantes.
- Crie os templates correspondentes para cada rota usando Jinja2.
- Utilize estruturas de controle como loops e condicionais para exibir as informações corretamente.
- Cada página deve incluir um menu de navegação com links para todas as outras páginas.

**Dica:** Para gerar links automaticamente nos templates, você pode usar a função **url\_for** do Flask. Sempre que precisar escrever o caminho de um arquivo, você pode usar a função **url\_for** apontando para a def da rota desejada. Exemplo:

```
<a href="{ { url_for('nome_da_def_da_rota') } }">Página X</a>
```

Para testar seu aplicativo, crie alguns objetos de Clube, Estudante e Professor e use-os para preencher seus templates.