



LISTA 09 – Exercício de Programação com Flask e Jinja2 - YOUTUBE

Crie um site em Flask que funcione como um diretório do Youtube. O site deve ser estruturado assim:

- **Página Inicial:** Exibe uma lista de todos os canais. Ao clicar em um canal, você é levado para a página de detalhes do canal.
- **Página de Detalhes do Canal:** Mostra informações sobre o canal e uma lista dos vídeos postados no canal. Ao clicar em um vídeo, você é levado à página de detalhes do vídeo.
- **Página de Detalhes do Vídeo:** Exibe detalhes sobre o vídeo, como título, duração, quantidade de likes e dislikes, e uma seção para comentários.

Você vai precisar das seguintes classes:

- **Classe Youtube:** Funciona como um banco de dados de todos os canais no YouTube.
 - Atributos: lista de objetos da classe Canal.
 - Métodos: adicionar novos canais, buscar (e retornar) um canal pelo nome, retornar a quantidade total de vídeos (soma dos vídeos de todos os canais).
- **Classe Canal:** Cada objeto Canal representa um canal no YouTube.
 - Atributos: uma lista de Vídeos (cada item na lista de vídeos é um objeto Video) e uma descrição.
 - Métodos: um método para adicionar novos vídeos ao canal e outro para calcular a quantidade total de likes em todos os vídeos do canal.
- **Classe Video:** Cada objeto Video representa um vídeo no YouTube.
 - Atributos: titulo, duracao, visualizações, likes, dislikes e uma lista de comentarios.
 - Métodos: adicionar e listar comentários, aumentar likes e aumentar dislikes.

Você vai precisar das seguintes Rotas no app:

- A rota **"/" (página inicial)** deve exibir uma lista de todos os canais e informar o total de vídeos que o site possui.
- A rota **"/canal/<nome_canal>"** deve exibir a página de detalhes do canal especificado, com informações sobre o canal e uma lista de vídeos postados nesse canal. Cada vídeo listado deve ser um link para a respectiva página de detalhes do vídeo.
- A rota **"/video/<nome_video>"** deve exibir a página de detalhes do vídeo especificado, com informações sobre o vídeo, como título, duração, quantidade de likes e dislikes, e uma seção para mostrar os comentários.



Você vai precisar das seguintes templates:

- Template da página inicial (index.html):
 - Exibir uma lista dos canais, com o nome do canal, que é um link para a página de detalhes do canal.
 - Exibir a quantidade total de vídeos presentes em todos os canais.
 - Parâmetro que deve ser enviado na função `render_template`: objeto Youtube.
- Template da página de detalhes do Canal (canal.html):
 - Informações sobre o canal: nome e descrição.
 - Título da lista: "Vídeos".
 - Exibir uma lista dos vídeos deste canal, com o título do vídeo (que é um link para a página de detalhes do vídeo), a duração e o número de visualizações.
 - Parâmetro que deve ser enviado na função `render_template`: objeto Canal, que contém uma lista de objetos Vídeo.
- Template da página de detalhes do Vídeo (video.html):
 - Informações sobre o vídeo: título, duração, número de visualizações, likes, dislikes.
 - Título da seção: "Comentários".
 - Exibir uma lista de comentários para este vídeo.
 - Parâmetro que deve ser enviado na função `render_template`: objeto Vídeo, que contém uma lista de comentários.

Além disso, todos os templates deverão conter uma navegação consistente, como um cabeçalho ou um menu que permita ao usuário retornar à página inicial ou acessar outros canais ou vídeos.

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 os objetos necessários no programa principal e use-os para preencher seus templates.