

 <p>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA PIAUI</p>	<p><b>INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ</b></p> <p><b>Curso: ADS</b></p> <p><b>Disciplina: Tópicos especiais em programação</b></p> <p><b>Professor: Ely</b></p>
--	---

## Exercícios

---

### *Slide django\_01*

---

Execute os passos abaixo e crie um projeto e uma aplicação:

1. Crie um projeto chamado **mysite**;
2. Crie o banco;
3. Suba o servidor;
4. Acesso o endereço da aplicação;
5. Crie uma aplicação chamada **pools**;
6. Registre a aplicação;
7. Defina uma view;
8. Defina uma url;
9. Suba o servidor;
10. Teste novamente o endereço da aplicação.

---

### *Slide django\_02*

---

1. Crie o arquivo de rotas da aplicação pools e faça as alterações necessárias no arquivo urls.py do projeto mysite;
2. Crie uma página de boas-vindas chamada index.html e exiba-a;
3. Crie uma classe Question com os campos "question\_text", closed (padrão False) e pub\_date, representando a data da publicação da pergunta;
4. Altere a página index para exibir todas as questões;
5. Crie uma página question.html, para exibir as questões segundo o padrão de url: /question/5/ ... onde o id é o id da questão;
6. Altere os arquivos de views, rotas etc. para exibir essa página;
7. Cadastre pelo menos 3 questões via prompt;
8. Altere a classe Question para ser persistida em banco;
9. Gere o esquema de migração;

10. Gere o banco;
11. Altere a view que exibe as questões estáticas para resgatar as questões do banco.
12. Exiba-as em um formato de grid/table. Ex:  
<http://getbootstrap.com.br/components/#panels-tables>

---

### *Slide django\_03*

---

1. Crie uma classe Choice com os atributos já herdando da classe model:
  - question: relacionamento muitos para um com a classe Question. No mapeamento, permita que esse atributo seja nulo;
  - choice\_text: o texto da escolha;
  - votes: a quantidade de votos dessa alternativa.
2. Adicione para a classe Question e Choice o método `__str__()`
3. Gere o esquema de migração;
4. Gere o banco;
5. Crie instancias de questões e associe a instâncias de choice com pelo menos 2 escolhas para cada questão;
6. Altere a página de questões para que além da questão exibida, sejam listadas as associadas à questão.

---

### *Slide django\_04 e 05*

---

1. Altere a página de listagem das questões para listar todas as questões, exibindo ordenadas em ordem decrescente de publicação. Dica, use `Question.objects.order_by('-pub_date')` ;
2. Crie as seguintes funcionalidades:
  - results: exibe uma questão, suas escolhas e o resultado das escolhas. Padrão de URL: `/question/5/results/` . Exiba o resultado com algum componente visual que mostre a quantidade de votos e percentual de votos para cada escolha.
  - vote: permite votar em uma escolha de uma questão. Padrão de url: `/question/5/vote/` .
  - Altere a página de questões para que ao lado de cada questão tenha um link para exibir os resultados apenas das questões fechadas;
  - Altere a página também para exibir ao lado de cada escolha um link que incremente em um a quantidade de votos apenas para as questões abertas. Defina um padrão de url para o voto;
3. Crie uma página de gerenciamento de questões onde as questões são listadas segundo o padrão `/question/5/manage`

- Ao lado de cada opção crie um links:
  - i. para “remover” a opção da questão;
  - ii. Para alterar o status de fechado da questão para True ou False
  - iii. Defina padrões de url para a remoção e abertura;
  - iv. Use uma confirmação via JavaScript para confirmar as operações acima. Exemplo:  
`https://stackoverflow.com/questions/10462839/javascript-confirmation-dialog-on-href-link`
- Exiba abaixo das opções já cadastradas as opções não associadas a uma questão específica e um link para associá-las a questão exibida. Defina um padrão de url também.