



Django Framework do Zero

Como Criar um Aplicativo Web

Canal [youtube.com/@VamosCodar](https://www.youtube.com/@VamosCodar) - Wanderson Reis

15 - Relacionamento 1 para Muitos

Requisitos

- 1) Aulas 01 a 14 - Django Framework do Zero - Playlist:

<https://www.youtube.com/playlist?list=PLFOqHo8oljzewcT23HCxJV0xWO451CTJe>

Nesta aula:

- 1) Implementar novo model Choice (alternativas para a Enquete)
- 2) Cadastrar algumas alternativas via ORM API
- 3) Exibir as alternativas na página da pergunta

Observação: Lembre-se de estar com o ambiente virtual ativado (venv).

Passo 1: Implementar novo model Choice

Neste passo vamos incluir o model Choice responsável pelo cadastro das alternativas vinculadas às perguntas (model Question).

Edite o arquivo **django-do-zero/polls/models.py** e acrescente o novo model da seguinte forma (**atenção apenas ao conteúdo novo**):

```
# (...manter tudo o que já existe...)
```

```
# na parte inferior do arquivo
```

```
class Choice(models.Model):
```

```
    # chave estrangeira que vincula cada alternativa a uma pergunta
```

```
    question = models.ForeignKey(Question, on_delete=models.CASCADE)
```

```
    choice_text = models.CharField(max_length=200) # texto da alternativa
```

```
    votes = models.IntegerField(default=0) # contagem de votos na alternativa
```

```
    def __str__(self):
```

```
        return self.choice_text
```

(salve o arquivo).

Após definir ou modificar qualquer model é necessário gerar a migração de dados e aplicar as mudanças no banco de dados. Para mais referência, consulte a Aula 5 sobre introdução aos Models.

Execute no terminal os comandos:

`python manage.py makemigrations` (ENTER)

```
wanderson >> python manage.py makemigrations
Migrations for 'polls':
  polls/migrations/0003_choice.py
  - Create model Choice
```

`python manage.py migrate` (ENTER)

```
wanderson >> python manage.py migrate
Operations to perform:
  Apply all migrations: accounts, admin, auth, contenttypes, polls, sessions
Running migrations:
  Applying polls.0003_choice... OK
```

Passo 2: Cadastrar algumas alternativas via ORM API

Para demonstrar o funcionamento do novo model e como criar o relacionamento com a pergunta, vamos criar algumas alternativas para a Question com ID = 1.

Faremos direto no terminal para aprendermos também a manipulação dos registros pelo prompt do Python (pode ser útil em diversas situações):

Execute no terminal os comandos (**atenção apenas os comandos, pressione o ENTER para executar**):

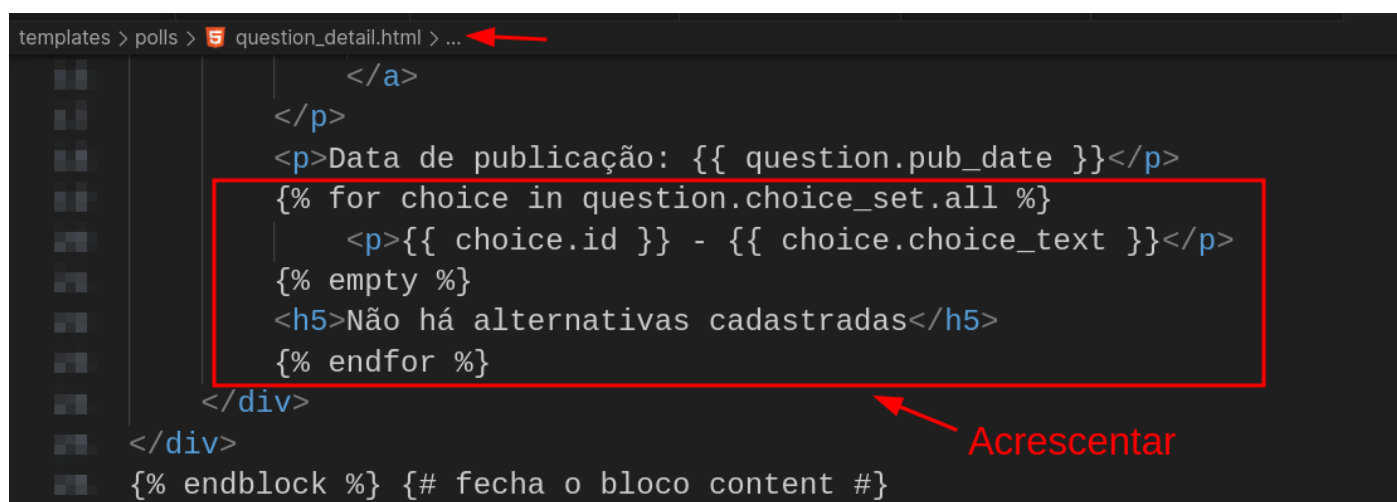
`python manage.py shell` (ENTER)

```
>>> from polls.models import Question, Choice # importa os models
>>> question_1 = Question.objects.filter(pk=1) # recupera a questão com ID 1
>>> question_1 # exibe a question com ID 1
<QuerySet [<Question: Question object (1)>]>
>>> question_1[0].question_text # exibe o texto da pergunta 1, acessando via lista - posição 0
>>> question_1 = Question.objects.filter(pk=1).first() # busca a pergunta com id 1 e retorna o 1º item
>>> question_1.question_text # exibe o texto da pergunta 1
'Qual sua cor favorita ?'
>>> question_1 # exibe a representação do objeto, método __str__
<Question: Question object (1)>
>>> choice_1 = Choice(question=question_1, choice_text="Vermelho") # cria uma alternativa
>>> choice_1 # o objeto está somente na memória
<Choice: Vermelho>
>>> choice_1.save() # salva os dados no banco
>>> choice_2 = Choice(question=question_1, choice_text="Azul") # cria uma segunda alternativa
```

```
>>> choice_1.id # alternativa 1 possui ID pois já foi salvo no banco
1
>>> choice_2.id # está ainda está na memória e ainda não tem id (chave primária)
>>> choice_2.save()
>>> choice_2.id
2
>>> choice_3 = Choice(question=question_1, choice_text="Amarelo") # cria a terceira alternativa
>>> choice_3.save()
>>> question_1.choice_set.all() # através da pergunta é possível acessar todas as alternativa vinculadas
<QuerySet [<Choice: Vermelho>, <Choice: Azul>, <Choice: Amarelo>]>
>>> quit() # sai do shell (terminal python)
```

Passo 3: Exibir as alternativas na página da pergunta

Edite o arquivo `django-do-zero/templates/polls/question_detail.html` e acrescente o bloco de códigos conforme indicado na próxima imagem (...manter tudo o que já existe...):



```
templates > polls > question_detail.html > ...
</a>
</p>
<p>Data de publicação: {{ question.pub_date }}</p>
{% for choice in question.choice_set.all %}
    <p>{{ choice.id }} - {{ choice.choice_text }}</p>
{% empty %}
    <h5>Não há alternativas cadastradas</h5>
{% endfor %}
</div>
</div>
{% endblock %} {# fecha o bloco content #}
```

Acrescentar

(salve o arquivo).

Em seguida suba o servidor de desenvolvimento e acesse a nova rota.

`python manage.py runserver` (ENTER)

No navegador acesse <http://127.0.0.1:8000/enquete/1/show>

Faça o teste também com o ID de uma pergunta que não possui alternativas cadastradas.

Exemplo de como será exibida as alternativas.

127.0.0.1:8000/enquete/1/show

Django do Zero

Pergunta

Qual sua cor favorita ? [Editar](#) [Excluir](#)

Data de publicação: 2 de Outubro de 2023 às 00:00

1 - Vermelho

2 - Azul

3 - Amarelo

Rodapé...