Django é um arcabouço de desenvolvimento Web de alto nível construído em python. É um dos mais utilizados no mercado junto ao node.js, pela velocidade e praticidade de desenvolvimento.

* Guia de instalação:
  1. <https://docs.djangoproject.com/pt-br/3.1/intro/install/>

* Tutorial necessário:
  1. <https://docs.djangoproject.com/pt-br/3.1/intro/tutorial01/>

○ <https://docs.djangoproject.com/pt-br/3.1/intro/tutorial02/>

**Importante:**

* **Nenhum conhecimento de HTML, CSS, ou javascript é necessário para esta prova, todas questões podem ser feitas somente utilizando python, alterando os arquivos “models.py”, “admin.py” e “settings.py” do tutorial.**
* **Deixe os valores padrão na seção “Configuração do banco de dados” e ignore completamente a seção “Brincando com a API”**

**Entregáveis:** Um .zip, contendo todos os códigos realizados. Um print da página no navegador para as questões 3, 4 e 5.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Questão 1:** Faça o tutorial, até a parte 2, até que consiga ver em seu navegador a seguinte tela:



**Questão 2:** No tutorial, foram criados os modelos "Question" e "Choice". Adicione mais dois modelos:

* Equipamento, com os atributos "nome", "ip" e "cidade";
* Cidade, com o atributo "nome";

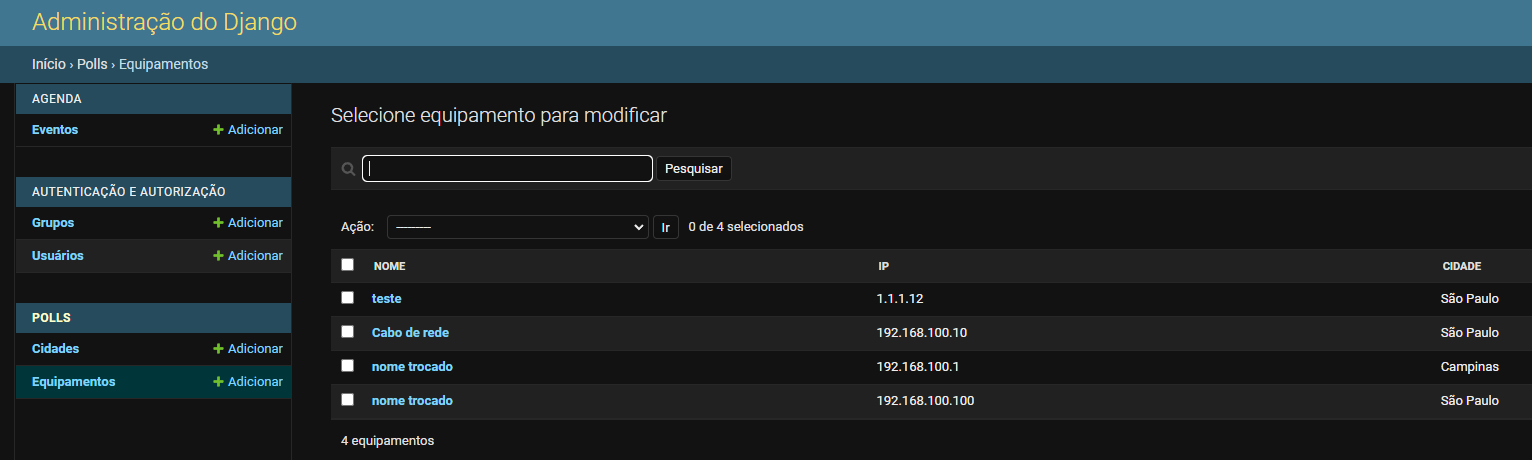
**dica**: Não esqueça de dar estes comandos após alterar o arquivo **models.py** de seu app

"*python manage.py makemigrations*"

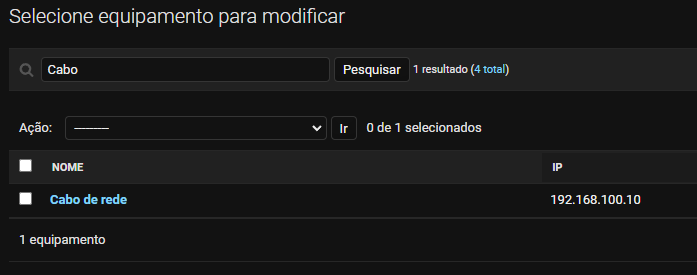
"*python manage.py migrate*"

**Questão 3:** Pule direto para a parte 7 do tutorial ([Link da parte 7)](https://docs.djangoproject.com/pt-br/3.1/intro/tutorial07/). Baseado na parte 7, faça o ModelAdmin do modelo Equipamento (EquipamentoAdmin):

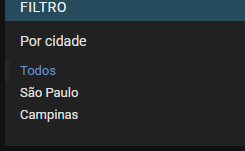
* Mostrar na tabela os campos “nome”, “IP”, “cidade”

****

* Deixar a tabela pesquisável por “IP” e “nome”



* Criar um filtro para “cidade”

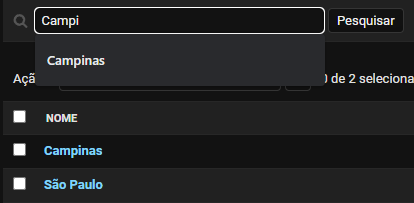


**Questão 4:** Crie um “EquipamentoInline” e associe ao ModelAdmin de cidade, assim como na parte 7 ([Link da parte 7)](https://docs.djangoproject.com/pt-br/3.1/intro/tutorial07/), de forma a poder adicionar equipamentos diretamente no formulário de alterar uma Cidade.

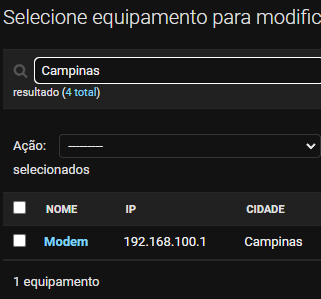
****

**Questão 5:** Pesquise e execute:

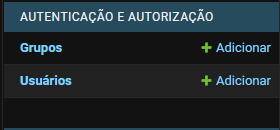
* Adicionar “cidade” na lista de autocomplete\_fields (e fazer funcionar)



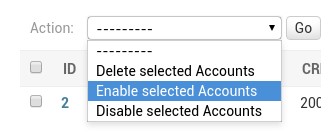
* Fazer um equipamento ser pesquisável no search\_fields pelo nome da cidade

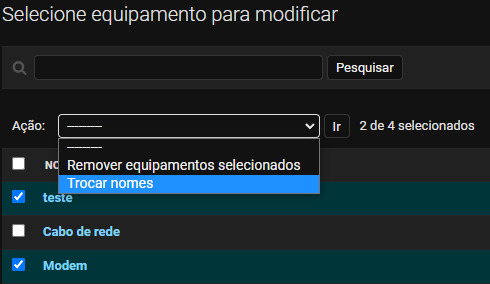


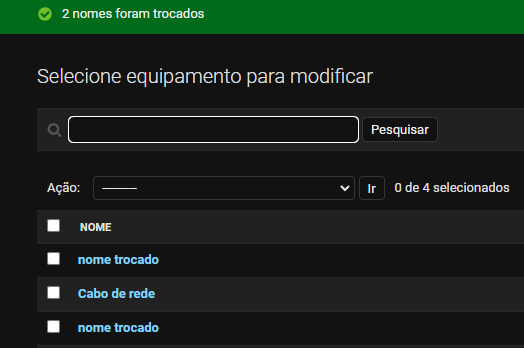
* Alterar a linguagem do django admin para português



* Realize um “override” no método \_\_str\_\_ do modelo “cidade”, para que seja printável
* Uma ação no django admin para trocar o nome de um equipamento adicionado na página web para ‘nome trocado’



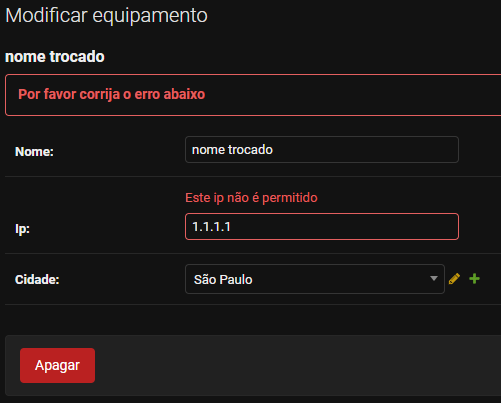
**R:** ****

****

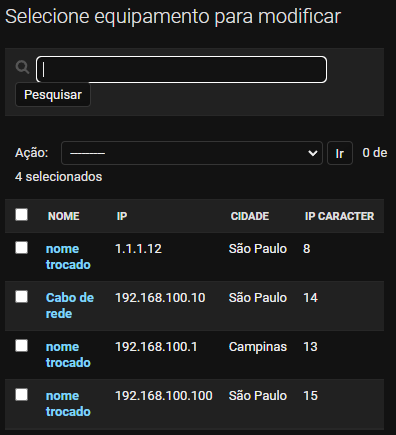
* Toda vez que um equipamento for salvo na página, printar “ok” no terminal.



* Não permitir que seja cadastrado nenhum equipamento com o IP “1.1.1.1”



* Mostrar quantos caracteres tem o IP do equipamento em uma das colunas de sua tabela.



* Crie outro app, com modelos quaisquer, que apareçam na página de administração

