Create View – Class based Views Django

Create View refers to a view (logic) to create an instance of a table in the database. We have already discussed basics of Create View in [Create View – Function based Views Django](). Class-based views provide an alternative way to implement views as Python objects instead of functions. They do not replace function-based views, but have certain differences and advantages when compared to function-based views:

Django Create View – Class Based Views

Consider a project named project5 having an app named myapp.

In `myapp/models.py,`

```python
# Create your models here.
# import the standard Django Model
# from built-in library
from django.db import models

# declare a new model with a name "GeeksModel"
class LoginTable(models.Model):

  # fields of the model
  username = models.CharField(max_length = 200)
  password = models.TextField()

  # renames the instances of the model
  # with their title name
  def __str__(self):
    return self.username
```

After creating this model, we need to run two commands in order to create Database for the same.

Python manage.py makemigrations

Python manage.py migrate

Now let's create some instances of this model using shell, run form bash,or terminal

Python manage.py shell
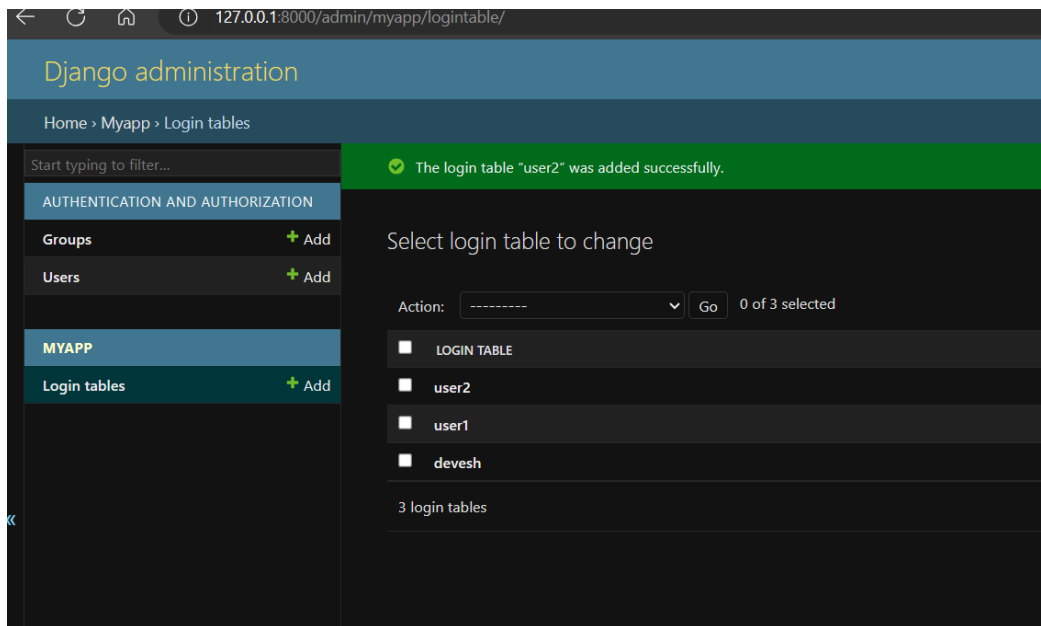
Enter following commands

```python
>>> from myapp.models import LoginTable
>>> LoginModel.objects.create(
            username="user1",
            password="pass1").save()
>>> LoginModel.objects.create(
            username="user2",
            password="pass2").save()
>>> LoginModel.objects.create(
            username="user3",
            password="pass3").save()
```

Let's register this model. In geeks/admin.py

```python
from django.contrib import admin
from .models import LoginTable
# Register your models here.
admin.site.register(LoginTable)
```

Now we have everything ready for the back end. Verify that instances have been created from http://localhost:8000/admin/

(please create superuser)

Django administration

Home › Myapp › Login tables

Start typing to filter...

AUTHENTICATION AND AUTHORIZATION

Groups   **+** Add
Users   **+** Add

MYAPP

Login tables   **+** Add

«

✓ The login table "user2" was added successfully.

Select login table to change

Action: ---------   Go   0 of 3 selected

☐   LOGIN TABLE

☐   user2

☐   user1

☐   devesh

3 login tables

Let's create a view and template for the same.

```python
from django.shortcuts import render
from django.views.generic.edit import CreateView
from .models import LoginTable

class LoginCreate(CreateView):
    # specify the model for create view
    model = LoginTable

    # specify the fields to be displayed
    fields = ['username', 'password']

    # specify the template name
    template_name = 'login.html'  # Replace 'your_template_name.html' with the
actual template name
```

Create a template in templates/login.html

```html
<form method="POST" enctype="multipart/form-data">

  <!-- Security token -->
  {% csrf_token %}

  <!-- Using the formset -->
  {{ form.as_p }}

  <input type="submit" value="Submit">
</form>
```

Let's check what is there on *http://localhost:8000/create/*

Username:

Password:

Submit