Createview – Class Based Views Django:-

Consider a project named geeksforgeeks having an app named geeks.

After you have a project and an app, let's create a model of which we will be creating instances through our view. In geeks/models.py:-

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

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

    # fields of the model
    title = models.CharField(max_length = 200)
    description = models.TextField()

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

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
```

Class Based Views automatically setup everything from A to Z. One just needs to specify which model to create View for and the fields. Then Class based CreateView will automatically try to find a template in app_name/modelname_form.html

In our case it is geeks/templates/geeks/geeksmodel_form.html

Let’s create our class based view. In geeks/views.py,

```
from django.views.generic.edit import CreateView
from .models import GeeksModel

class GeeksCreate(CreateView):

    # specify the model for create view
    model = GeeksModel

    # specify the fields to be displayed

    fields = ['title', 'description']
```

Now create a url path to map the view. In geeks/urls.py,

```
from django.urls import path

# importing views from views..py
from .views import GeeksCreate
urlpatterns = [
    path('', GeeksCreate.as_view() ),
]
```

Create a template in templates/geeks/geeksmodel_form.html:-

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

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

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

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