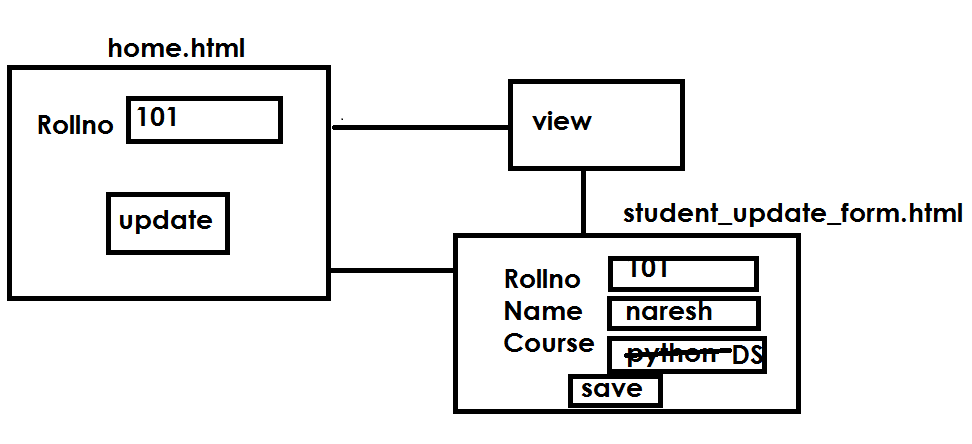
**UpdateView**

A view that displays a form for editing an existing object, redisplaying the form with validation errors (if there are any) and saving changes to the object. This uses a form automatically generated from the object’s model class (unless a form class is manually specified).



**Models.py**

from django.db import models

# Create your models here.

class Student(models.Model):

    rollno=models.IntegerField(primary\_key=True)

    name=models.CharField(max\_length=20)

    course=models.CharField(max\_length=20)

**Views.py**

class StudentView(UpdateView):

    model=Student

    fields="\_\_all\_\_"

    template\_name="stud\_update.html"

    success\_url="/home"

template

**stud\_update.html**

<form method="post">

    {%csrf\_token%}

    {{form.as\_p}}

    <input type="submit" value="save">

</form>

**Home.html**

<html>

    <body>

        <form action="/update/"{{rno}} method="post">

            Rollno <input type="text" name="rno">

            <input type="submit" value="update">

        </form>

    </body>

</html>

Urls.py

from django.contrib import admin

from django.urls import path

from cbvapp import views

urlpatterns = [

    path('admin/', admin.site.urls),

    path('home/',views.home),

    path('update/<int:pk>',views.StudentView.as\_view()),

]

Settings.py

INSTALLED\_APPS = [

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

    'cbvapp',

]

TEMPLATE\_PATH=os.path.join(BASE\_DIR,"templates")

TEMPLATES = [

    {

        'BACKEND': 'django.template.backends.django.DjangoTemplates',

        'DIRS': [TEMPLATE\_PATH],

        'APP\_DIRS': True,

        'OPTIONS': {

            'context\_processors': [

                'django.template.context\_processors.debug',

                'django.template.context\_processors.request',

                'django.contrib.auth.context\_processors.auth',

                'django.contrib.messages.context\_processors.messages',

            ],

        },

    },

]

How to the application?

http:localhost:8000/update/101 🡪 rollno

**DeleteView**

A view that displays a confirmation page and deletes an existing object. The given object will only be deleted if the request method is **POST**. If this view is fetched via **GET**, it will display a confirmation page that should contain a form that POSTs to the same URL.

Models.py

from django.db import models

# Create your models here.

class Employee(models.Model):

    empno=models.IntegerField(primary\_key=True)

    ename=models.CharField(max\_length=20)

**urls.py**

from django.contrib import admin

from django.urls import path

from cbvapp import views

urlpatterns = [

    path('admin', admin.site.urls),

    path('delete/<int:pk>',views.EmpDeleteView.as\_view()),

    path("s/",views.display)

]

Views.py

from django.shortcuts import render

from cbvapp.models import Employee

from django.http import HttpResponse

# Create your views here.

from django.views.generic import DeleteView

class EmpDeleteView(DeleteView):

    model=Employee

    template\_name="confirmation.html"

    success\_url="/s"

def display(request):

    msg="<h1>Employee Deleted</h1> "

    return HttpResponse(msg)

template

<form method="post">{% csrf\_token %}

    <p>Are you sure you want to delete "{{ object }}"?</p>

    {{ form }}

    <input type="submit" value="Confirm">

</form>

Settings.py

INSTALLED\_APPS = [

    'django.contrib.admin',

    'django.contrib.auth',

    'django.contrib.contenttypes',

    'django.contrib.sessions',

    'django.contrib.messages',

    'django.contrib.staticfiles',

    'cbvapp',

]

import os.path

ROOT\_URLCONF = 'cbvproj4.urls'

TEMPLATEPATH=os.path.join(BASE\_DIR,"templates")

TEMPLATES = [

    {

        'BACKEND': 'django.template.backends.django.DjangoTemplates',

        'DIRS': [TEMPLATEPATH],

        'APP\_DIRS': True,

        'OPTIONS': {

            'context\_processors': [

                'django.template.context\_processors.debug',

                'django.template.context\_processors.request',

                'django.contrib.auth.context\_processors.auth',

                'djang

Running application

<http://localhost:8000/delete/1>