Forms example

To create environment 🡪 python -m venv myenv

Activate the environment 🡪 .\myenv\Scripts\activate

#Instal Django 🡪 if u created environment newly

**Create a new Django project**:

django-admin startproject feedback\_project

cd feedback\_project

**Create a new Django app**

python manage.py startapp feedback

**Add the new app to your project's settings**:

settings.py, add feedback to the INSTALLED\_APPS

INSTALLED\_APPS = [

...

' feedback',

]

**Create the models** in feedback/model.py

from django.db import models

class Feedback(models.Model):

name = models.CharField(max\_length=100)

email = models.EmailField()

feedback = models.TextField()

question1 = models.CharField(max\_length=10, choices=[

('yes', 'Yes'),

('no', 'No'),

])

question2 = models.CharField(max\_length=20, choices=[

('excellent', 'Excellent'),

('good', 'Good'),

('average', 'Average'),

('poor', 'Poor'),

])

question3 = models.CharField(max\_length=10, choices=[

('yes', 'Yes'),

('no', 'No'),

])

created\_at = models.DateTimeField(auto\_now\_add=True)

def \_\_str\_\_(self):

return f'Feedback from {self.name}'

**feedback/forms.py**

**from django import forms**

**from .models import Feedback**

**from django.core.exceptions import ValidationError**

**class FeedbackForm(forms.ModelForm):**

**class Meta:**

**model = Feedback**

**fields = ['name', 'email', 'feedback', 'question1', 'question2', 'question3']**

**# Custom validation for the email field**

**def clean\_email(self):**

**email = self.cleaned\_data.get('email')**

**if not email.endswith('@acharya.ac.in’):**

**raise ValidationError('Email must be from the domain @acharya.ac.in')**

**return email**

**# Custom validation for the feedback field**

**def clean\_feedback(self):**

**feedback = self.cleaned\_data.get('feedback')**

**prohibited\_words = [‘xyz’, 'zzz'] # List of prohibited words**

**for word in prohibited\_words:**

**if word in feedback:**

**raise ValidationError(f'Feedback contains prohibited word: {word}')**

**return feedback**

**Create views feedback/views.py**

from django.shortcuts import render, redirect

from .forms import FeedbackForm

def feedback\_view(request):

if request.method == 'POST':

form = FeedbackForm(request.POST)

if form.is\_valid():

form.save()

return redirect('feedback\_thanks')

else:

form = FeedbackForm()

return render(request, 'feedback/feedback\_form.html', {'form': form})

def feedback\_thanks\_view(request):

return render(request, 'feedback/feedback\_thanks.html')

**Create URLs for these views** in feedback/urls.py:

from django.urls import path

from .views import feedback\_view, feedback\_thanks\_view

urlpatterns = [

path('', feedback\_view, name='feedback\_form'),

path('thanks/', feedback\_thanks\_view, name='feedback\_thanks'),

]

**Create templates** for each view:

**feedback/templates/feedback/feedback\_form.html**

<!DOCTYPE html>

<html>

<head>

<title>Feedback Form</title>

</head>

<body>

<h1>Feedback Form</h1>

<form method="post">

{% csrf\_token %}

{{ form.as\_p }}

<fieldset>

<legend>Questionnaire</legend>

<div>

<label for="id\_question1">Was the course content helpful?</label><br>

<input type="radio" id="id\_question1\_0" name="question1" value="yes"> Yes<br>

<input type="radio" id="id\_question1\_1" name="question1" value="no"> No<br>

</div>

<div>

<label for="id\_question2">How would you rate the instructor's teaching?</label><br>

<input type="radio" id="id\_question2\_0" name="question2" value="excellent"> Excellent<br>

<input type="radio" id="id\_question2\_1" name="question2" value="good"> Good<br>

<input type="radio" id="id\_question2\_2" name="question2" value="average"> Average<br>

<input type="radio" id="id\_question2\_3" name="question2" value="poor"> Poor<br>

</div>

<div>

<label for="id\_question3">Would you recommend this course to others?</label><br>

<input type="radio" id="id\_question3\_0" name="question3" value="yes"> Yes<br>

<input type="radio" id="id\_question3\_1" name="question3" value="no"> No<br>

</div>

</fieldset>

<button type="submit">Submit</button>

</form>

{% if form.errors %}

<div style="color: red;">

<h2>Form errors:</h2>

<ul>

{% for field in form %}

{% if field.errors %}

<li>{{ field.label }}: {{ field.errors|striptags }}</li>

{% endif %}

{% endfor %}

{% if form.non\_field\_errors %}

<li>{{ form.non\_field\_errors|striptags }}</li>

{% endif %}

</ul>

</div>

{% endif %}

</body>

</html>

**feedback/templates/feedback/feedback\_thanks.html**

<!DOCTYPE html>

<html>

<head>

<title>Thank You</title>

</head>

<body>

<h1>Thank You for Your Feedback!</h1>

<p>Your feedback has been submitted successfully.</p>

</body>

</html>

**Feedback\_project/urls.py**

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

path('admin/', admin.site.urls),

path('feedback/', include('feedback.urls')),

]

**Apply migrations**

python manage.py makemigrations feedback

python manage.py migrate

**To run**

Python manage.py runserver

**Ouput link 🡪** <http://127.0.0.1:8000/feedback/>