

## Django CRUD App (Without forms.py) – Step-by-Step Guide

This document will guide you through creating a Django CRUD (Create, Read, Update, Delete) application from scratch without using forms.py. We'll use only HTML forms for user input.

### Step 1 – Project and App Setup

1. Create a Django project and an app:

- ```
django-admin startproject crud_project
cd crud_project
python manage.py startapp students
```

2. Add 'students' in INSTALLED\_APPS inside crud\_project/settings.py

### Step 2 – Create the Student Model

In students/models.py:

```
from django.db import models

class Student(models.Model):
    name = models.CharField(max_length=100)
    age = models.IntegerField()
    department = models.CharField(max_length=100)

    def __str__(self):
        return self.name
```

- Run migrations to create the table:

```
python manage.py makemigrations
python manage.py migrate
```

### Step 3 – Configure URLs

students/urls.py:

```
from django.urls import path
from . import views

urlpatterns = [
    path('', views.student_list, name='student_list'),
    path('add/', views.student_add, name='student_add'),
    path('edit/<int:id>/', views.student_edit, name='student_edit'),
```

```
    path('delete/<int:id>/', views.student_delete, name='student_delete'),  
]
```

crud\_project/urls.py:

```
from django.contrib import admin  
from django.urls import path, include
```

```
urlpatterns = [  
    path('admin/', admin.site.urls),  
    path("", include('students.urls')),  
]
```

## Step 4 – Create Views

students/views.py:

```
from django.shortcuts import render, redirect, get_object_or_404  
from .models import Student
```

```
def student_list(request):  
    students = Student.objects.all()  
    return render(request, 'students/student_list.html', {'students': students})
```

```
def student_add(request):  
    if request.method == 'POST':  
        name = request.POST['name']  
        age = request.POST['age']  
        department = request.POST['department']  
        Student.objects.create(name=name, age=age, department=department)  
        return redirect('student_list')  
    return render(request, 'students/student_add.html')
```

```
def student_edit(request, id):  
    student = get_object_or_404(Student, id=id)  
    if request.method == 'POST':  
        student.name = request.POST['name']  
        student.age = request.POST['age']  
        student.department = request.POST['department']  
        student.save()  
        return redirect('student_list')  
    return render(request, 'students/student_edit.html', {'student': student})
```

```
def student_delete(request, id):
    student = get_object_or_404(Student, id=id)
    if request.method == 'POST':
        student.delete()
        return redirect('student_list')
    return render(request, 'students/student_delete.html', {'student': student})
```

## Step 5 – Create Templates

Create folder structure: students/templates/students/

Create these HTML files inside it:

### student\_list.html

```
<!DOCTYPE html>
<html>
<head><title>Student List</title></head>
<body>
    <h1>All Students</h1>
    <a href="{% url 'student_add' %}">Add New Student</a>
    <table border="1" cellpadding="8">

<tr><th>ID</th><th>Name</th><th>Age</th><th>Department</th><th>Actions</th></tr>
    {% for s in students %}
    <tr>
        <td>{{ s.id }}</td><td>{{ s.name }}</td><td>{{ s.age }}</td><td>{{ s.department
    }}</td>
        <td><a href="{% url 'student_edit' s.id %}">Edit</a> | <a href="{% url 'student_delete'
s.id %}">Delete</a></td>
    </tr>
    {% endfor %}
    </table>
</body></html>
```

### student\_add.html

```
<!DOCTYPE html>
<html>
<head><title>Add Student</title></head>
<body>
    <h1>Add Student</h1>
    <form method="POST">
        {% csrf_token %}
        Name: <input type="text" name="name" required><br>
```

```

Age: <input type="number" name="age" required><br>
Department: <input type="text" name="department" required><br><br>
<button type="submit">Save</button>
</form>
<a href="{% url 'student_list' %}">Back</a>
</body></html>

```

#### student\_edit.html

```

<!DOCTYPE html>
<html>
<head><title>Edit Student</title></head>
<body>
<h1>Edit Student</h1>
<form method="POST">
  {% csrf_token %}
  Name: <input type="text" name="name" value="{{ student.name }}" required><br>
  Age: <input type="number" name="age" value="{{ student.age }}" required><br>
  Department: <input type="text" name="department" value="{{ student.department }}"
required><br><br>
  <button type="submit">Update</button>
</form>
<a href="{% url 'student_list' %}">Back</a>
</body></html>

```

#### student\_delete.html

```

<!DOCTYPE html>
<html>
<head><title>Delete Student</title></head>
<body>
<h1>Delete Confirmation</h1>
<p>Are you sure you want to delete <b>{{ student.name }}</b>?</p>
<form method="POST">
  {% csrf_token %}
  <button type="submit">Yes, Delete</button>
</form>
<a href="{% url 'student_list' %}">Cancel</a>
</body></html>

```

## Step 6 – Run the Server

Run the Django server with:

- `python manage.py runserver`

Visit `http://127.0.0.1:8000/` in your browser to test the CRUD operations.