

LAPORAN PRAKTIKUM SISTEM INFORMASI
MATERI KE – 2
Website CRUD menggunakan django



OLEH:
Moch Alfian Miftachul Huda
220605110088
KELAS G

1. Pembuatan Program Django

- a) Install Django dan mysqlclient terlebih dahulu

```
PS C:\laragon\www\djangoku\mahasiswa_project> pip install django
```

```
PS C:\laragon\www\djangoku\mahasiswa_project> pip install django mysqlclient
```

- b) Buat virtual environment python dan aktifkan virtual environmentnya

```
PS C:\laragon\www\djangoku\mahasiswa_project> python -m venv venv
```

```
venv\Scripts\activate
```

- c) Buat project djangonya,serta tambahkan beberapa program seperti manage.py.

```
django-admin startproject mahasiswa project
```

```
python manage.py startapp mahasiswa
```

- d) Lalu buat folder templates untuk menampung tampilan programnya ,untuk struktur programnya kurang lebih seperti ini:

```
▼ mahasiswa_project
  ▼ mahasiswa
    ▼ templates\mahasiswa
      <> mahasiswa_confirm_delete.html
      <> mahasiswa_detail.html
      <> mahasiswa_form.html
      <> mahasiswa_list.html
    __init__.py
    admin.py
    apps.py
    forms.py
    models.py
    tests.py
    urls.py
    views.py
  ▼ mahasiswa_project
    > __pycache__
    __init__.py
    asgi.py
    settings.py
    urls.py
    wsgi.py
    manage.py
  > venv
```

2. Create data

- **Mahasiswa_forms.html**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-
scale=1.0">
<title>Form Mahasiswa</title>
<!-- Tambahkan Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<div class="container mt-4">
<h1 class="text-center">Form Mahasiswa</h1>
<form method="post">
    {% csrf_token %}

    <!-- NIM Field -->
    <div class="mb-3">
        <label for="{{ form.nim.id_for_label }}" class="form-
label">NIM</label>
        {{ form.nim|default_if_none:""|safe }}
    </div>

    <!-- Nama Field -->
    <div class="mb-3">
        <label for="{{ form.nama.id_for_label }}" class="form-
label">Nama</label>
        {{ form.nama|default_if_none:""|safe }}
    </div>

    <!-- Jurusan Field -->
    <div class="mb-3">
        <label for="{{ form.jurusan.id_for_label }}" class="form-
label">Jurusan</label>
        {{ form.jurusan|default_if_none:""|safe }}
    </div>

    <!-- Alamat Field -->
    <div class="mb-3">
        <label for="{{ form.alamat.id_for_label }}" class="form-
label">Alamat</label>
        {{ form.alamat|default_if_none:""|safe }}
    </div>

    <button type="submit" class="btn btn-primary">Simpan
Data</button>
```

```

        <!-- Back to List Button -->
        <a href="{% url 'mahasiswa_list' %}" class="btn btn-secondary
ms-2">Kembali ke Daftar Mahasiswa</a>
    </form>
</div>

<!-- Tambahkan Bootstrap JS dan Popper.js -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

Program mahasiswa_forms.html untuk membuat tampilan inputan data lalu konfigurasi tipe inputan diproses oleh forms.py dan models.py

- **Forms.py**

```

from django import forms
from .models import Mahasiswa

class MahasiswaForm(forms.ModelForm):
    JURUSAN_CHOICES = [
        ('TI', 'Teknik Informatika'),
        ('BIO', 'Biologi'),
        ('FIS', 'Fisika'),
        ('ARS', 'Arsitektur'),
    ]

    # Menggunakan widget Select untuk jurusan
    jurusan = forms.ChoiceField(choices=JURUSAN_CHOICES,
widget=forms.Select(attrs={'class': 'form-select'}))
    class Meta:
        model = Mahasiswa
        fields = ['nim', 'nama', 'jurusan', 'alamat']
        widgets = {
            'nim': forms.TextInput(attrs={'class': 'form-control'}),
            'nama': forms.TextInput(attrs={'class': 'form-control'}),
            'jurusan': forms.TextInput(attrs={'class': 'form-control'}),
            'alamat': forms.Textarea(attrs={'class': 'form-control'}),
        }

```

- **Models.py**

```
from django.db import models

class Mahasiswa(models.Model):
    nim = models.CharField(max_length=15, unique=True)
    nama = models.CharField(max_length=100)
    jurusan = models.CharField(max_length=100)
    alamat = models.TextField()

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

- **Hasil**

Form Mahasiswa

NIM

Nama

Jurusan

Alamat

- Atur database supaya terhubung dengan phpMyAdmin pada **setting.py**

```
DATABASES = {
    'default': {
        'ENGINE': 'django.db.backends.mysql',
        'NAME': 'mhs',
        'USER': 'root',
        'PASSWORD': '',
```

```

        'HOST': 'localhost'
        'PORT': '3306',
    }
}

```

- Database yang telah dibuat

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	bigint		No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	nim	varchar(15)	utf8mb4_0900_ai_ci	No	None			Change Drop More
<input type="checkbox"/>	3	nama	varchar(100)	utf8mb4_0900_ai_ci	No	None			Change Drop More
<input type="checkbox"/>	4	jurusan	varchar(100)	utf8mb4_0900_ai_ci	No	None			Change Drop More
<input type="checkbox"/>	5	alamat	longtext	utf8mb4_0900_ai_ci	No	None			Change Drop More

- Buat **Views.py** untuk mengatur jalannya CRUD

```

from django.shortcuts import render, get_object_or_404, redirect
from .models import Mahasiswa
from .forms import MahasiswaForm

def mahasiswa_list(request):
    mahasiswa = Mahasiswa.objects.all()
    return render(request, 'mahasiswa/mahasiswa_list.html', {'mahasiswa':
mahasiswa})

def mahasiswa_detail(request, pk):
    mahasiswa = get_object_or_404(Mahasiswa, pk=pk)
    return render(request, 'mahasiswa/mahasiswa_detail.html',
{'mahasiswa': mahasiswa})

def mahasiswa_create(request):
    if request.method == 'POST':
        form = MahasiswaForm(request.POST)
        if form.is_valid():
            form.save()
            return redirect('mahasiswa_list')
    else:
        form = MahasiswaForm()
    return render(request, 'mahasiswa/mahasiswa_form.html', {'form':
form})

```

```

def mahasiswa_update(request, pk):
    mahasiswa = get_object_or_404(Mahasiswa, pk=pk)
    if request.method == 'POST':
        form = MahasiswaForm(request.POST, instance=mahasiswa)
        if form.is_valid():
            form.save()
            return redirect('mahasiswa_list')
    else:
        form = MahasiswaForm(instance=mahasiswa)
    return render(request, 'mahasiswa/mahasiswa_form.html', {'form':
form})

def mahasiswa_delete(request, pk):
    mahasiswa = get_object_or_404(Mahasiswa, pk=pk)
    if request.method == 'POST':
        mahasiswa.delete()
        return redirect('mahasiswa_list')
    return render(request, 'mahasiswa/mahasiswa_confirm_delete.html',
{'mahasiswa': mahasiswa})

```

- Jangan lupa atur pathnya pada **urls.py**

```

from django.urls import path
from . import views

urlpatterns = [
    path("", views.mahasiswa_list, name='mahasiswa_list'),
    path('mahasiswa/<int:pk>/', views.mahasiswa_detail,
name='mahasiswa_detail'),
    path('mahasiswa/new/', views.mahasiswa_create,
name='mahasiswa_create'),
    path('mahasiswa/<int:pk>/edit/', views.mahasiswa_update,
name='mahasiswa_update'),
    path('mahasiswa/<int:pk>/delete/', views.mahasiswa_delete,
name='mahasiswa_delete'),
]

```

3. Read data

Buat tampilan readnya pada program mahasiswa_list.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">

```

```

<meta name="viewport" content="width=device-width, initial-
scale=1.0">
<title>Data Mahasiswa</title>
<!-- Tambahkan Bootstrap CSS -->
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
<div class="container mt-4">
<h1 class="text-center">Daftar Mahasiswa</h1>

<!-- Baris ini akan memastikan tombol "Tambah Mahasiswa"
berada di pojok kanan -->
<div class="row">
<div class="col text-end">
<a href="{% url 'mahasiswa_create' %}" class="btn btn-
primary">Tambah Mahasiswa</a>
</div>
</div>

<table class="table table-striped mt-3">
<thead>
<tr>
<th scope="col">#</th>
<th scope="col">NIM</th>
<th scope="col">Nama</th>
<th scope="col">Alamat</th>
<th scope="col">Jurusan</th>
<th scope="col" class="text-end">Aksi</th>
</tr>
</thead>
<tbody>
{% for mhs in mahasiswa %}
<tr>
<th scope="row">{{ forloop.counter }}</th>
<td>{{ mhs.nim }}</td> <!-- Menampilkan NIM -->
<td>
<a href="{% url 'mahasiswa_detail' mhs.pk %}"
class="text-decoration-none">{{ mhs.nama }}</a>
</td>
<td>{{ mhs.alamat }}</td> <!-- Menampilkan Alamat -->
<td>{{ mhs.jurusan }}</td> <!-- Menampilkan Jurusan -->
<td class="text-end">

```



```

        <a href="{% url 'mahasiswa_update' mhs.pk %}"
class="btn btn-sm btn-warning">Edit</a>
        <a href="{% url 'mahasiswa_delete' mhs.pk %}"
class="btn btn-sm btn-danger">Hapus</a>
    </td>
</tr>
{% endfor %}
</tbody>
</table>
</div>

<!-- Tambahkan Bootstrap JS dan Popper.js -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- **Hasil**

Daftar Mahasiswa

					Tambah Mahasiswa	
#	NIM	Nama	Alamat	Jurusan	Aksi	
1	343837584	el mangrove	portugal	BIO	Edit	Hapus
2	345678	el cekek	argentina	TI	Edit	Hapus
3	474349858	Martin Paees	kediri, jl jalan	FIS	Edit	Hapus
4	3243456	el gasing	lostarika	ARS	Edit	Hapus
5	3434566	el deargod	switster	BIO	Edit	Hapus
6	7689809	el silahkan	giorno selatan	ARS	Edit	Hapus
7	33487348	el reaction	bradinton grewwen	TI	Edit	Hapus

4. Update

Ini adalah logika update sudah dibuat pada program views.py sebelumnya

```

def mahasiswa_update(request, pk):
    mahasiswa = get_object_or_404(Mahasiswa, pk=pk)
    if request.method == 'POST':
        form = MahasiswaForm(request.POST, instance=mahasiswa)

```

```

        if form.is_valid():
            form.save()
            return redirect('mahasiswa_list')
        else:
            form = MahasiswaForm(instance=mahasiswa)
            return render(request, 'mahasiswa/mahasiswa_form.html', {'form':
form})

```

Ketika user menekan tombol edit maka program akan menjalankan request post dan menuju ke program mahasiswa_form.html untuk mengupdate data mahasiswanya

5. Delete

```

def mahasiswa_delete(request, pk):
    mahasiswa = get_object_or_404(Mahasiswa, pk=pk)
    if request.method == 'POST':
        mahasiswa.delete()
        return redirect('mahasiswa_list')
    return render(request,
'mahasiswa/mahasiswa_confirm_delete.html', {'mahasiswa':
mahasiswa})

```

pada logika delete program melakukan request post pada mahasiswa_confirm_delete.html untuk melakukan konfirmasi penghapusan data

- mahasiswa_confirm_delete.html

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Konfirmasi Hapus</title>
    <!-- Tambahkan Bootstrap CSS -->
    <link                href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
<body>
    <div class="container mt-5">
        <div class="row justify-content-center">
            <div class="col-md-6">

```

```

<div class="card">
  <div class="card-body text-center">
    <h1 class="card-title">Konfirmasi Hapus Mahasiswa</h1>
    <p>Apakah Anda yakin ingin menghapus data mahasiswa
<strong>{{ mahasiswa.nama }}</strong>?</p>
    <form method="post" class="d-inline-block">
      {% csrf_token %}
      <button          type="submit"          class="btn          btn-
danger">Hapus</button>
    </form>
    <a href="{% url 'mahasiswa_list' %}" class="btn btn-secondary
ms-2">Batal</a>
  </div>
</div>
</div>
</div>
</div>

<!-- Tambahkan Bootstrap JS dan Popper.js -->
<script          src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-
alpha3/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

- Hasil

Konfirmasi Hapus Mahasiswa

Apakah Anda yakin ingin menghapus data mahasiswa **el cekek**?

Hapus

Batal