

# WriteUp Tugas 1

---

## 1. [70] Fitur Aplikasi

Pada pembuatan tugas 1 ini, saya memulai dengan membuat sebuah proyek pada Eclipse. Kemudian, Saya mulai membuat model-model pada com.example.model berdasarkan data-data yang ada pada siduk\_dki.sql. Setelah membuatnya, saya mencoba untuk integrasi database dengan import siduk\_dki pada localhost/phpmyadmin, kemudian melakukan konfigurasi untuk primary key, indexing, serta auto increment untuk IDE seperti id\_penduduk serta id\_keluarga. Setelah itu melakukan konfigurasi pada eclipse untuk application properties proyek. Setelah itu buat sebuah index.html yang berisikan sebuah halaman utama berisi 2 field input, yaitu pencarian penduduk berdasarkan NIK, serta pencarian keluarga berdasarkan Nomor KK.

### a. [7] Tampilkan Data Penduduk Berdasarkan NIK

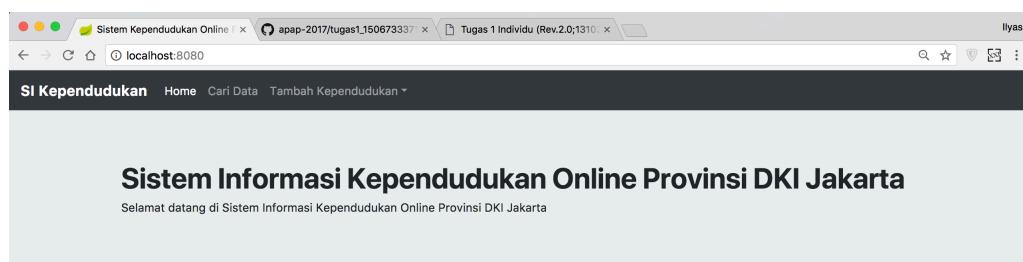
Untuk menampilkan data penduduk, saya membuat sebuah mapper pada com.example.dao, saya membuat PendudukMapper serta Keluarga Mapper. Pada PendudukMapper saya membuat sebuah query method untuk melakukan seleksi pada penduduk berdasarkan NIK-nya. Kemudian membuat sevice pada com.example.service, saya membuat PendudukService. Pada PendudukService saya membuat sebuah method berdasarkan service tersebut. Untuk controller, saya juga membuat sebuah SIKependudukanController pada com.example.controller, dan membuat method untuk menampilkan Penduduk dengan method RequestParam pada /penduduk?{nik}. Untuk mengambil lokasi penduduk, Saya juga membuat KelurahanMapper, KecamatanMapper, serta KotaMapper beserta service-service nya untuk melakukan query mengambil nama berdasarkan ID lokasi-lokasi tersebut.

```

@RequestMapping("/penduduk")
public String viewPenduduk (Model model, @RequestParam(value = "nik") String nik){
    PendudukModel penduduk = pendudukDAO.selectPenduduk(nik);
    if (penduduk != null) {
        KeluargaModel keluarga = keluargaDAO.selectKeluargabyID(penduduk.getId_keluarga());
        KelurahanModel kelurahan = kelurahanDAO.selectKelurahanbyID(keluarga.getId_kelurahan());
        KecamatanModel kecamatan = kecamatanDAO.selectKecamatanbyID(kelurahan.getId_kecamatan());
        KotaModel kota = kotaDAO.selectKotabyID(kecamatan.getId_kota());
        model.addAttribute ("penduduk", penduduk);
        model.addAttribute("keluarga", keluarga);
        model.addAttribute("kelurahan", kelurahan);
        model.addAttribute("kecamatan", kecamatan);
        model.addAttribute("kota", kota);
        return "view-penduduk";
    }
    return "not-found";
}

```

Serta sebuah halaman view-penduduk.html yang akan menampilkan data penduduk.



#### Lihat Data Penduduk Berdasarkan NIK

Masukkan Nomor Induk Kependudukan

**Lihat**

#### Lihat Data Keluarga Berdasarkan NKK

Masukkan Nomor Kartu Keluarga

**Lihat**

© Ilyas Fahreza / 1506733371 | Kelas APAP B



#### Lihat Data Penduduk - 3101011108170002

**Ubah Data Penduduk**

**Non Aktifkan Penduduk**

NIK	3101011108170002
Nama	Panca Muhammad Pratama
Tempat/Tanggal lahir	Jakarta/2017-08-11
Alamat	Jln. Jayawijaya No. 549
RT/RW	077/023
Kelurahan/Desa	PULAU PARI
Kecamatan	KEPULAUAN SERIBU SELATAN
Kota/Kabupaten	KABUPATEN KEPULAUAN SERIBU
Golongan Darah	A-

**b. [7] Tampilkan Data Keluarga Beserta Daftar Anggotanya Berdasarkan Nomor KK**

Untuk menampilkan data keluarga, saya membuat sebuah mapper pada com.example.dao, saya membuat KeluargaMapper. Pada PendudukMapper saya membuat sebuah query method untuk melakukan seleksi pada penduduk berdasarkan NKK-nya, serta membuat sebuah query untuk mengambil seluruh keluarganya pada selectAnggotaKeluarga. Kemudian membuat sevice pada com.example.service, saya membuat keluargaService. Pada KeluargaService saya membuat sebuah method berdasarkan service tersebut. Untuk controller, saya juga membuat method untuk menampilkan Keluarga dengan method RequestParam pada /keluarga?{nkk}.

```
@RequestMapping("/penduduk")
public String viewPenduduk(Model model, @RequestParam(value = "nik") String nik){
    PendudukModel penduduk = pendudukDAO.selectPenduduk(nik);
    if (penduduk != null) {
        KeluargaModel keluarga = keluargaDAO.selectKeluargabyID(penduduk.getId_keluarga());
        KelurahanModel kelurahan = kelurahanDAO.selectKelurahambyID(keluarga.getId_kelurahan());
        KecamatanModel kecamatan = kecamatanDAO.selectKecamatantbyID(kelurahan.getId_kecamatan());
        KotaModel kota = kotaDAO.selectKotabyID(kecamatan.getId_kota());
        model.addAttribute("penduduk", penduduk);
        model.addAttribute("keluarga", keluarga);
        model.addAttribute("kelurahan", kelurahan);
        model.addAttribute("kecamatan", kecamatan);
        model.addAttribute("kota", kota);
        return "view-penduduk";
    }
    return "not-found";
}
```

Serta sebuah halaman view-keluarga.html yang akan menampilkan data keluarga.



### Lihat Data Penduduk Berdasarkan NIK

Masukkan Nomor Induk Kependudukan

Lihat

### Lihat Data Keluarga Berdasarkan NKK

Masukkan Nomor Kartu Keluarga

Lihat

© Ilyas Fahreza / 1506733371 / Kelas APAP B



### Lihat Data Keluarga - 3101010104070001

Ubah Data Keluarga

NKK 3101010104070001  
Alamat Gg. Bappenas No. 9  
RT/RW 013/008  
Kelurahan/Desa PULAU PARI  
Kecamatan KEPULAUAN SERIBU SELATAN  
Kota KABUPATEN KEPULAUAN SERIBU

No	Nama	NIK	Jenis Kelamin	Tempat Lahir	Tanggal Lahir	Agama	Pekerjaan	Status Perkawinan	Status dalam Keluarga	Kewarganegaraan
1	Galar Permadi	3101012309120001	Laki-laki	Langsa	2012-09-23	Islam	BELUM/TIDAK BEKERJA	Belum Kawin	Anak	WNI
2	Ratna Suartini	3101015806690001	Perempuan	Jakarta	1969-06-18	Katholik	PEMBANTU RUMAH TANGGA	Cerai Hidup	Pembantu	WNI

### c. [10] Menambahkan Penduduk Baru Sebagai Anggota Keluarga

Untuk menambahkan sebuah Penduduk, Saya menambahkan sebuah method insert pada PendudukMapper. Kemudian method berdasarkan service tersebut pada service juga dibuat sebuah method creatNIK untuk pembuatan NIK berdasarkan aturan, yaitu dengan melakukan substring dan concat pada tanggal lahir, serta kode wilayah.

```

public PendudukModel addPenduduk(PendudukModel penduduk) {
    String nik_penduduk = createNik(penduduk);
    penduduk.setNik(nik_penduduk);
    pendudukMapper.addPenduduk(penduduk);
    return penduduk;
}

public String createNik(PendudukModel penduduk) {
    KeluargaModel keluarga = keluargaMapper.selectKeluargaByID(penduduk.getId_keluarga());
    KelurahanModel kelurahan = kelurahanMapper.selectKelurahanbyID(keluarga.getId_kelurahan());
    KecamatanModel kecamatan = kecamatanMapper.selectKecamatanbyID(kelurahan.getId_kecamatan());
    String kode_lokasi = kecamatan.getKode_kecamatan();

    String tanggal_lahir = penduduk.getTanggal_lahir();
    String tanggal = tanggal_lahir.substring(8, 10);
    if (penduduk.getJenis_kelamin() == 1) {
        int tgl = Integer.parseInt(tanggal);
        tgl = tgl + 40;
        tanggal = Integer.toString(tgl);
    }

    String tempNik = kode_lokasi.substring(0, kode_lokasi.length()-1) + tanggal + tanggal_lahir.substring(5, 7) + tanggal_lahir.substring(2,4);

    String nik_sebelum = pendudukMapper.getNIKSebelum(tempNik);
    Long nik = Long.parseLong(tempNik + "0001");
    if(nik_sebelum != null) {
        nik = Long.parseLong(nik_sebelum) + 1;
    }

    String nik_penduduk = Long.toString(nik);
    return nik_penduduk;
}

```

Saya membuat controller untuk melakukan insert berdasarkan mapper tersebut untuk melakukan insert pada database. Sebelumnya, pada localhost/phpmyadmin untuk ID penduduk juga dibuat auto increment. Untuk menampilkannya, buat form dengan nama form-add-penduduk.html yang akan meretrieve pada halaman sukses apabila berhasil.

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	<b>id</b>	bigint(20)		UNSIGNED	No	None	AUTO_INCREMENT	Change  Drop  Primary  Unique  More
2	<b>nik</b>	char(16)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
3	<b>nama</b>	varchar(128)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
4	<b>tempat_lahir</b>	varchar(128)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
5	<b>tanggal_lahir</b>	date			No	None		Change  Drop  Primary  Unique  More
6	<b>jenis_kelamin</b>	int(1)			No	None		Change  Drop  Primary  Unique  More
7	<b>is_wni</b>	tinyint(1)			No	None		Change  Drop  Primary  Unique  More
8	<b>id_keluarga</b>	bigint(20)		UNSIGNED	No	None		Change  Drop  Primary  Unique  More
9	<b>agama</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
10	<b>pekerjaan</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
11	<b>status_perkawinan</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
12	<b>status_dalam_keluarga</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
13	<b>golongan_darah</b>	varchar(32)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
14	<b>is_wafat</b>	tinyint(1)			No	0		Change  Drop  Primary  Unique  More

Tambah Penduduk

Nama  
Ilyas Fahreza

Tempat Lahir  
Bekasi

Tanggal Lahir  
14/06/1997

Jenis Kelamin  
Laki-laki

Golongan Darah  
O+

Status Perkawinan  
Belum Kawin

Agama  
Islam

Pekerjaan  
MAHASISWA

Status Kewarganegaraan  
WNI

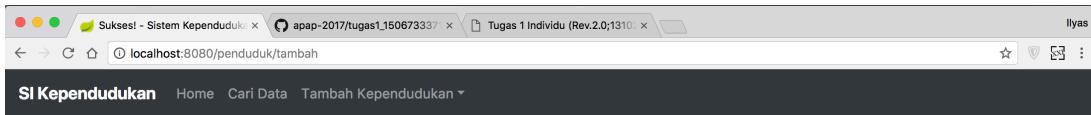
Status Kematian  
Hidup

Id Keluarga  
18

Status dalam Keluarga  
Anak

**Tambah**

© Ilyas Fahreza | 1506733371 | Kelas APAP B



## Sukses

Penduduk dengan NIK 3101011406970001 berhasil ditambahkan

© Ilyas Fahreza | 1506733371 | Kelas APAP B

### d. [7] Menambahkan Keluarga Baru

Untuk menambahkan sebuah Keluarga, Saya menambahkan sebuah method insert pada KeluargaMapper. Kemudian method berdasarkan service tersebut, pada KeluargaService juga dibuat sebuah NKK untuk pembuatan

NKK berdasarkan aturan, yaitu dengan melakukan substring dan concat pada tanggal pembuatan, serta kode wilayah.

```
public String addKeluarga(KeluargaModel keluarga) {  
    KelurahanModel kelurahan = kelurahanMapper.selectKelurahanbyID(keluarga.getId_kelurahan());  
    KecamatanModel kecamatan = kecamatanMapper.selectKecamatanbyID(kelurahan.getId_kecamatan());  
    String kode_kecamatan = kecamatan.getKode_kecamatan();  
  
    LocalDate localDate = LocalDate.now();  
    String tanggal = DateTimeFormatter.ofPattern("yyyy/MM/dd").format(localDate);  
  
    String tempNkk = kode_kecamatan.substring(0, kode_kecamatan.length()-1) + tanggal.substring(8, 10) + tanggal.substring(5, 7) + tanggal.substring(2,4);  
  
    String nkk_sebelum = keluargaMapper.getNKKSebelum(tempNkk);  
    Long nkk = Long.parseLong(tempNkk + "0001");  
    if(nkk_sebelum != null) {  
        nkk = Long.parseLong(nkk_sebelum) + 1;  
    }  
  
    String nkk_keluarga = Long.toString(nkk);  
    keluarga.setNomor_kk(nkk_keluarga);  
    keluarga.setId_kelurahan(kelurahan.getId());  
    keluargaMapper.addKeluarga(keluarga);  
    return nkk_keluarga;  
}
```

Saya membuat controller untuk melakukan insert berdasarkan mapper tersebut untuk melakukan insert pada database. Sebelumnya, pada localhost/phpmyadmin untuk ID penduduk juga dibuat auto increment. Sama seperti pada table Penduduk. Untuk menampilkannya, buat form dengan nama form-add-keluarga.html yang akan meretrieve pada halaman sukses apabila berhasil.



### Tambah Keluarga

Alamat

RT

RW

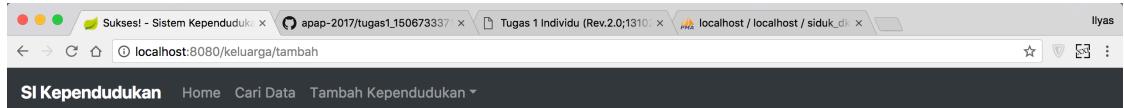
Kelurahan

Kecamatan

Kota

**Tambah**

© Ilyas Fahreza / 1506733371 / Kelas APAP B



### Sukses

Keluarga dengan Nomor KK 3171022210170001 berhasil ditambahkan

© Ilyas Fahreza / 1506733371 / Kelas APAP B

### e. [8] Mengubah Data Penduduk

Untuk mengubah data sebuah Penduduk, Saya menambahkan sebuah method update pada PendudukMapper. Kemudian method berdasarkan service tersebut, pada PendudukService juga ada pemeriksaan untuk memeriksa apakah ada perubahan yang menyebabkan pergantian NIK.

```
public void updatePenduduk(PendudukModel penduduk) {
    PendudukModel selectedPenduduk = pendudukMapper.selectPendudukById(penduduk.getId());
    if(penduduk.getGolongan_darah() == null) {
        penduduk.setGolongan_darah(selectedPenduduk.getGolongan_darah());
    } if(penduduk.getStatus_perkawinan() == null) {
        penduduk.setStatus_perkawinan(selectedPenduduk.getStatus_perkawinan());
    } if(penduduk.getId_keluarga() == null) {
        penduduk.setStatus_dalam_keluarga(selectedPenduduk.getStatus_dalam_keluarga());
    } if(penduduk.getAgama() == null) {
        penduduk.setAgama(selectedPenduduk.getAgama());
    }

    BigInteger id_penduduk = selectedPenduduk.getId();
    if (!selectedPenduduk.equals(penduduk)) {
        if ((selectedPenduduk.getTanggal_lahir().equals(penduduk.getTanggal_lahir()) && selectedPenduduk.getId_keluarga() == penduduk.getId_keluarga() && selectedPenduduk.getJenis_kelamin() == penduduk.getJenis_kelamin())) {
            penduduk.setNIK(createNIK(penduduk));
        }
        pendudukMapper.updatePenduduk(penduduk, id_penduduk);
    }
}
```

SI Kependudukan Home Cari Data Tambah Kependudukan ▾

**Ubah Penduduk**

NIK Penduduk  
3101011406970001

Nama  
Ilyas Fahreza

Tempat Lahir  
Bekasi

Tanggal Lahir  
14/06/1997

Jenis Kelamin  
Laki-laki

Golongan Darah  
O+

Status Perkawinan  
Belum Kawin

Agama  
Islam

Pekerjaan  
MAHASISWA

Status Kewarganegaraan  
WNI

Id Keluarga  
18

Status dalam Keluarga  
Anak

**Ubah**



## Sukses

Penduduk dengan NIK 3101011406970001 berhasil diubah

© Ilyas Fahreza | 1506733371 | Kelas APAP

### f. [8] Mengubah Data Keluarga

Untuk mengubah data sebuah Keluarga, Saya menambahkan sebuah method update pada KeluargaMapper. Kemudian method berdasarkan service tersebut, pada KeluargaService juga ada pemeriksaan untuk memeriksa apakah ada perubahan yang menyebabkan pergantian Nomor KK. Untuk mengambil waktu pada hari ini, menggunakan LocalDate serta DateTimeFormatter untuk mengubah bentuk datanya sesuai dengan yang ada pada database.

```
public void updateKeluarga(KeluargaModel keluarga) {
    KeluargaModel selectedKeluarga = keluargaMapper.selectKeluargabyNKK(keluarga.getNomor_kk());
    KelurahanModel kelurahan = kelurahanMapper.selectKelurahanbyID(keluarga.getId_kelurahan());
    KecamatanModel kecamatan = kecamatanMapper.selectKecamatanbyID(kelurahan.getId_kecamatan());
    selectedKeluarga.setId_kelurahan(kelurahan.getId());
    if (!selectedKeluarga.equals(keluarga)) {
        String nkk = keluarga.getNomor_kk();
        String current_nkk = nkk.substring(0, 12);
        String kode_kecamatan = kecamatan.getKode_kecamatan();

        LocalDate localDate = LocalDate.now();
        String tanggal = DateTimeFormatter.ofPattern("yyyy/MM/dd").format(localDate);

        String rilis = tanggal.substring(8, 10) + tanggal.substring(5, 7) + tanggal.substring(2, 4);
        String tempNkk = kode_kecamatan.substring(0, kode_kecamatan.length() - 1) + rilis;

        if (!tempNkk.equals(current_nkk)) {
            String nkk_sebelum = keluargaMapper.getNKKSebelum(tempNkk);
            Long nkk_baru = Long.parseLong(tempNkk + "0001");
            if (nkk_sebelum != null) {
                nkk_baru = Long.parseLong(nkk_sebelum) + 1;
            }
            keluarga.setNomor_kk(Long.toString(nkk_baru));
        }
        keluarga.setId_kelurahan(kelurahan.getId());
        keluargaMapper.updateKeluarga(keluarga);
    }
}
```

NKK  
3101010204020001

Alamat  
Jln. Abang No. 391

RT  
013

RW  
069

Kelurahan  
PULAU PARI

Ubah

© Ilyas Fahreza | 1506733371 | Kelas APAP B



## Sukses

Keluarga dengan Nomor KK 3101010204020001 berhasil diubah

© Ilyas Fahreza | 1506733371 | Kelas APAP B

### g. [8] Mengubah Status Kematian Penduduk

Untuk mengubah data sebuah status kematian Penduduk, Saya menambahkan sebuah button pada view-penduduk yang berfungsi menonaktifkan penduduk, dan mengirim method post pada controller dengan path penduduk/mati/. Method updateIsWafat juga dibuat pada PendudukMapper. Kemudian method berdasarkan service tersebut, pada PendudukService juga ada pemeriksaan untuk memeriksa apakah semua penduduk pada anggota sebuah keluarga sudah mati semua apa belum, apabila iya, maka status keluarga akan dinonaktifkan.

```
public void updateIsWafat(PendudukModel penduduk) {
    BigInteger id_keluarga = penduduk.getId_keluarga();
    List<PendudukModel> anggota_keluarga = pendudukMapper.selectAnggotaKeluarga(id_keluarga);
    boolean keluargaNonAktif = false;
    int counter = 0;

    for (PendudukModel p : anggota_keluarga) {
        if (p.getIs_wafat() != 0) {
            counter = counter + 1;
        }
    }
    if (counter == anggota_keluarga.size()) {
        keluargaNonAktif = true;
    }

    if (keluargaNonAktif) {
        keluargaMapper.updateStatusBerlaku(id_keluarga);
    }

    pendudukMapper.updateIsWafat(penduduk.getNik());
}
```



#### Lihat Data Penduduk - 3101013108860001

[Ubah Data Penduduk](#)

[Non Aktifkan Penduduk](#)



#### Sukses

Status kematian penduduk dengan NIK 3101013108860001 menjadi wafat

© Ilyas Fahreza | 1506733371 | Kelas APAP B

#### h. [15] Tampilkan Data Penduduk Berdasarkan Kota/Kabupaten, Kecamatan, dan Kelurahan Tertentu

Untuk menampilkan data penduduk pada suatu kelurahan tertentu, pertama adalah membuat sebuah select List<[Lokasi]Model> yang sesuai dengan Kecamatan, Kelurahan, serta Kota untuk meretrieve semua informasi kelurahan dan kecamatan yang ada. Setelah itu buat sebuah method pada controller untuk menerima masukan-masukan kota, kecamatan, dan kelurahan. Setiap kali ID terpilih, maka akan dicek apakah id\_kelurahan, id\_kota maupun id\_kecamatan telah dipilih.

```

@RequestMapping(value = "/penduduk/cari")
public String cariPendudukKelurahan (Model model,
    @RequestParam(value = "kt", required = false, defaultValue = "0") BigInteger id_kota,
    @RequestParam(value = "kc", required = false, defaultValue = "0") BigInteger id_kecamatan,
    @RequestParam(value = "kl", required = false, defaultValue = "0") BigInteger id_kelurahan){

List<KotaModel> kota_list = kotaDAO.selectKotaList();
model.addAttribute("kota_list", kota_list);

if (!id_kelurahan.equals(BigInteger.ZERO)) {
    model.addAttribute("nama_kota", kotaDAO.selectKotabyID(id_kota).getNama_kota());
    model.addAttribute("id_kota", id_kota);
    model.addAttribute("nama_kecamatan", kecamatanDAO.selectKecamatanbyID(id_kecamatan).getNama_kecamatan());
    model.addAttribute("id_kecamatan", id_kecamatan);
    model.addAttribute("nama_kelurahan", kelurahanDAO.selectKelurahanbyID(id_kelurahan).getNama_kelurahan());
    model.addAttribute("id_kelurahan", id_kelurahan);
    model.addAttribute("penduduk_list", pendudukDAO.selectPendudukbyIdKelurahan(id_kelurahan));
    model.addAttribute("showTable", "showTable");
}

else if (!id_kecamatan.equals(BigInteger.ZERO)) {
    model.addAttribute("nama_kota", kotaDAO.selectKotabyID(id_kota).getNama_kota());
    model.addAttribute("id_kota", id_kota);
    model.addAttribute("nama_kecamatan", kecamatanDAO.selectKecamatanbyID(id_kecamatan).getNama_kecamatan());
    model.addAttribute("id_kecamatan", id_kecamatan);
    model.addAttribute("kelurahan_list", kelurahanDAO.selectKelurahanList(id_kecamatan));
}

else if (!id_kota.equals(BigInteger.ZERO)) {
    model.addAttribute("nama_kota", kotaDAO.selectKotabyID(id_kota).getNama_kota());
    model.addAttribute("id_kota", id_kota);
    model.addAttribute("kecamatan_list", kecamatanDAO.selectKecamatanList(id_kota));
}
return "cari-penduduk";
}

```

Setelah menerima masukan-masukan tersebut, maka akan ditampilkan sebuah halaman cari-penduduk yang menerima masukan dari setiap perintah submit yang ada pada tampilan.



## Cari Penduduk

Pilih Kota

Pilih Kecamatan

Pilih Kelurahan/Desa

© Ilyas Fahreza / 1506733371 / Kelas APAP B

The screenshot shows a web application titled "Lihat Penduduk di Kelurahan PASAR MINGGU". The page displays a table of 750 entries, each containing a NIK number, a name, and a gender (Jenis Kelamin). The table is paginated, showing entries 1 through 10 of 750. The application interface includes a search bar, a navigation bar with links for Home, Cari Data, and Tambah Kependudukan, and a header with the user's name, Ilyas.

NIK	Nama	Jenis Kelamin
3171020101800001	Taufik Ghani Situmorang	Laki-laki
3171020101860001	Aris Utama	Laki-laki
3171020101940001	Harimurti Ikin Situmorang S.I.Kom	Laki-laki
3171020102950002	Irsad Narpati	Laki-laki
3171020103660001	Asmuni Firmansyah M.Pd	Laki-laki
3171020103920001	Darimin Adikara Kusumo S.Ked	Laki-laki
3171020104140001	Baktiono Saefullah	Laki-laki
3171020104150001	Mursita Situmorang	Laki-laki
3171020105900001	Lantar Octo Budiyanto S.Kom	Laki-laki
3171020106140004	Najam Mangunsong	Laki-laki

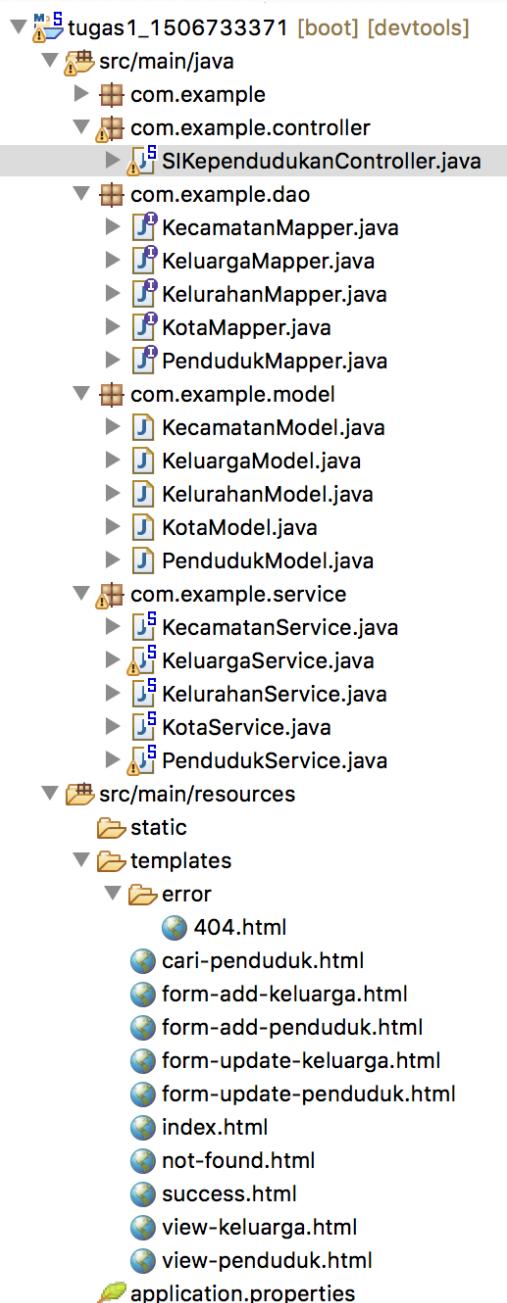
Showing 1 to 10 of 750 entries

Previous **1** 2 3 4 5 ... 75 Next

© Ilyas Fahreza | 1506733371 | Kelas APAP B

## 2. [20] Penerapan konsep MVC dan Layering

Pada tugas ini, saya melakukan MVC layering sesuai dengan yang telah diajarkan di kelas. Proyek terdiri dari 4 folder utama, yaitu com.example.dao, com.example.model, com.example.service, serta sebuah folder resouce/template yang berisi semua tampilan-tampilan front end untuk proyek. Pada proyek ini Saya juga membuat sebuah error handler untuk 404 pada template/error/ yang berfungsi menerima error halaman yang tidak ditemukan.



### 3. [10] Optimasi Database dan Stress Testing

Pada tugas kali ini, saya juga melakukan indexing pada setiap primary key pada database.

**Table structure for penduduk:**

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	<b>id</b>	bigint(20)		UNSIGNED	No	None	AUTO_INCREMENT	Change  Drop  Primary  Unique  More
2	<b>nik</b>	char(16)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
3	<b>nama</b>	varchar(128)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
4	<b>tempat_lahir</b>	varchar(128)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
5	<b>tanggal_lahir</b>	date			No	None		Change  Drop  Primary  Unique  More
6	<b>jenis_kelamin</b>	int(1)			No	None		Change  Drop  Primary  Unique  More
7	<b>is_wni</b>	tinyint(1)			No	None		Change  Drop  Primary  Unique  More
8	<b>id_keluarga</b>	bigint(20)		UNSIGNED	No	None		Change  Drop  Primary  Unique  More
9	<b>agama</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
10	<b>pekerjaan</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
11	<b>status_perkawinan</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
12	<b>status_dalam_keluarga</b>	varchar(64)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
13	<b>golongan_darah</b>	varchar(32)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  More
14	<b>is_wafat</b>	tinyint(1)			No	0		Change  Drop  Primary  Unique  More

**Table structure for kota:**

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	<b>id</b>	bigint(20)		UNSIGNED	No	None		Change  Drop  Primary  Unique  Index  Spatial  Fulltext  More
2	<b>kode_kota</b>	char(4)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  Index  Spatial  Fulltext  More
3	<b>nama_kota</b>	varchar(255)	utf8_unicode_ci		No	None		Change  Drop  Primary  Unique  Index  Spatial  Fulltext  More

**phpMyAdmin**

Recent | Favorites

Server: localhost » Database: siduk\_dki » Table: kelurahan

**Table structure** | **Relation view**

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	bigint(20)	UNDEFINED	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
2	id_kecamatan	bigint(20)	UNDEFINED	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
3	kode_kelurahan	char(10)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
4	nama_kelurahan	varchar(255)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
5	kode_pos	char(5)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Add to central columns](#)

[Remove from central columns](#)

[Print view](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Improve table structure](#)

[Add 1 column\(s\) after kode\\_pos](#) [Go](#)

+ Indexes

**Information**

Space usage		Row statistics	
Data	48 KiB	Format	Compact
Index	0 B	Collation	utf8_unicode_ci
Total	48 KiB	Creation	Oct 22, 2017 at 01:21 PM

**phpMyAdmin**

Recent | Favorites

Server: localhost » Database: siduk\_dki » Table: keluarga

**Table structure** | **Relation view**

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	bigint(20)	UNDEFINED	No	None	AUTO_INCREMENT		<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
2	nomor_kk	char(16)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
3	alamat	varchar(256)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
4	RT	char(3)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
5	RW	char(3)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
6	id_kelurahan	bigint(20)	UNDEFINED	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>
7	is_tidak_berlaku	tinyint(1)	UNDEFINED	No	0			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">More</a>

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Add to central columns](#)

[Remove from central columns](#)

[Print view](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Improve table structure](#)

[Add 1 column\(s\) after is\\_tidak\\_berlaku](#) [Go](#)

+ Indexes

**Information**

Space usage		Row statistics	
Data	5.5 MiB	Format	Compact
Index	6 MiB	Collation	utf8_unicode_ci
Total	11.5 MiB	Next autoworker	63,487
		Creation	Oct 22, 2017 at 04:05 PM

**phpMyAdmin**

Recent | Favorites

Server: localhost » Database: siduk\_dki » Table: kecamatan

**Table structure** | **Relation view**

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	bigint(20)	UNDEFINED	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
2	id_kota	bigint(20)	UNDEFINED	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
3	kode_kecamatan	char(7)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>
4	nama_kecamatan	varchar(255)	utf8_unicode_ci	No	None			<a href="#">Change</a> <a href="#">Drop</a> <a href="#">Primary</a> <a href="#">Unique</a> <a href="#">Index</a> <a href="#">Spatial</a> <a href="#">More</a>

[Check all](#) [With selected:](#) [Browse](#) [Change](#) [Drop](#) [Primary](#) [Unique](#) [Index](#) [Add to central columns](#)

[Remove from central columns](#)

[Print view](#) [Propose table structure](#) [Track table](#) [Move columns](#) [Improve table structure](#)

[Add 1 column\(s\) after nama\\_kecamatan](#) [Go](#)

+ Indexes

**Information**

Space usage		Row statistics	
Data	16 KiB	Format	Compact
Index	0 B	Collation	utf8_unicode_ci
Total	16 KiB	Creation	Oct 22, 2017 at 01:21 PM

Untuk melakukan StressTesting saya menggunakan fitur ke [1], ke [2], serta ke [8] untuk memeriksa setiap proses yang dilakukan.

HTTP Request

Method: GET Path: http://localhost:8080/penduduk?nik=3101010101690001 Content encoding:

Apache JMeter (3.2 r1790748)

File Edit Search Run Options Help

Test Plan Thread Group View Results in Table WorkBench

**View Results in Table**

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename:

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(...)
1	20:38:29.531	Thread Grou...	HTTP Request	173	✓	4155	147	172	115
2	20:38:29.531	Thread Grou...	HTTP Request	173	✓	4155	147	172	115
3	20:38:29.531	Thread Grou...	HTTP Request	173	✓	4155	147	172	115
4	20:38:29.619	Thread Grou...	HTTP Request	85	✓	4155	147	84	27
5	20:38:29.719	Thread Grou...	HTTP Request	17	✓	4155	147	17	1
6	20:38:29.843	Thread Grou...	HTTP Request	22	✓	4155	147	22	1
7	20:38:29.943	Thread Grou...	HTTP Request	16	✓	4155	147	16	2
8	20:38:30.043	Thread Grou...	HTTP Request	17	✓	4155	147	16	1
9	20:38:30.143	Thread Grou...	HTTP Request	14	✓	4155	147	14	0
10	20:38:30.246	Thread Grou...	HTTP Request	16	✓	4155	147	16	1
11	20:38:30.344	Thread Grou...	HTTP Request	14	✓	4155	147	14	1
12	20:38:30.447	Thread Grou...	HTTP Request	14	✓	4155	147	14	1
13	20:38:30.544	Thread Grou...	HTTP Request	18	✓	4155	147	18	1
14	20:38:30.648	Thread Grou...	HTTP Request	15	✓	4155	147	15	0
15	20:38:30.744	Thread Grou...	HTTP Request	10	✓	4155	147	10	1
16	20:38:30.849	Thread Grou...	HTTP Request	10	✓	4155	147	10	1
17	20:38:30.945	Thread Grou...	HTTP Request	10	✓	4155	147	10	1
18	20:38:31.049	Thread Grou...	HTTP Request	10	✓	4155	147	9	0
19	20:38:31.145	Thread Grou...	HTTP Request	31	✓	4155	147	31	1
20	20:38:31.244	Thread Grou...	HTTP Request	11	✓	4155	147	10	1
21	20:38:31.347	Thread Grou...	HTTP Request	11	✓	4155	147	11	1
22	20:38:31.448	Thread Grou...	HTTP Request	10	✓	4155	147	9	0
23	20:38:31.545	Thread Grou...	HTTP Request	10	✓	4155	147	10	1
24	20:38:31.644	Thread Grou...	HTTP Request	12	✓	4155	147	12	0
25	20:38:31.745	Thread Grou...	HTTP Request	10	✓	4155	147	10	1
26	20:38:31.847	Thread Grou...	HTTP Request	12	✓	4155	147	12	1
27	20:38:31.944	Thread Grou...	HTTP Request	11	✓	4155	147	11	1
28	20:38:32.048	Thread Grou...	HTTP Request	10	✓	4155	147	10	1
29	20:38:32.148	Thread Grou...	HTTP Request	14	✓	4155	147	14	1

Scroll automatically? Child samples? No of Samples 100 Latest Sample 16 Average 17 Deviation 28

HTTP Request

Method: GET Path: http://localhost:8080/keluarga?nomor\_kk=3101010104070001 Content encoding:

Apache JMeter (3.2 r1790748)

File Edit Search Run Options Help

Test Plan Thread Group View Results in Table WorkBench

**View Results in Table**

Name: View Results in Table

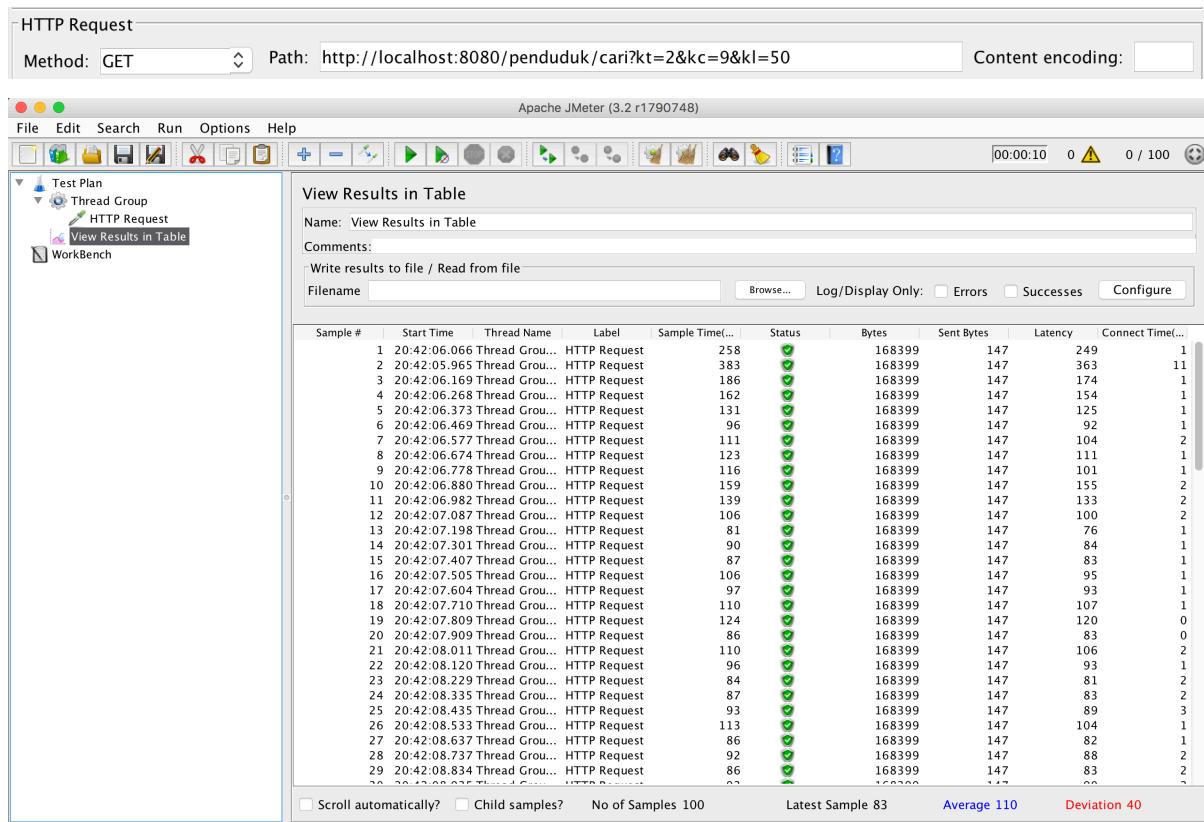
Comments:

Write results to file / Read from file

Filename:

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(...)
1	20:40:01.984	Thread Grou...	HTTP Request	38	✓	5244	152	38	3
2	20:40:02.088	Thread Grou...	HTTP Request	18	✓	5244	152	18	1
3	20:40:02.187	Thread Grou...	HTTP Request	14	✓	5244	152	14	1
4	20:40:02.288	Thread Grou...	HTTP Request	12	✓	5244	152	12	1
5	20:40:02.390	Thread Grou...	HTTP Request	12	✓	5244	152	12	1
6	20:40:02.487	Thread Grou...	HTTP Request	12	✓	5244	152	12	1
7	20:40:02.592	Thread Grou...	HTTP Request	16	✓	5244	152	16	1
8	20:40:02.688	Thread Grou...	HTTP Request	12	✓	5244	152	12	1
9	20:40:02.793	Thread Grou...	HTTP Request	17	✓	5244	152	17	1
10	20:40:02.889	Thread Grou...	HTTP Request	12	✓	5244	152	12	0
11	20:40:02.992	Thread Grou...	HTTP Request	11	✓	5244	152	10	1
12	20:40:03.088	Thread Grou...	HTTP Request	13	✓	5244	152	12	1
13	20:40:03.193	Thread Grou...	HTTP Request	15	✓	5244	152	15	1
14	20:40:03.290	Thread Grou...	HTTP Request	17	✓	5244	152	16	0
15	20:40:03.393	Thread Grou...	HTTP Request	15	✓	5244	152	15	1
16	20:40:03.490	Thread Grou...	HTTP Request	16	✓	5244	152	16	1
17	20:40:03.592	Thread Grou...	HTTP Request	15	✓	5244	152	15	1
18	20:40:03.694	Thread Grou...	HTTP Request	14	✓	5244	152	14	1
19	20:40:03.797	Thread Grou...	HTTP Request	16	✓	5244	152	16	1
20	20:40:03.898	Thread Grou...	HTTP Request	14	✓	5244	152	14	1
21	20:40:04.006	Thread Grou...	HTTP Request	14	✓	5244	152	14	1
22	20:40:04.109	Thread Grou...	HTTP Request	16	✓	5244	152	16	2
23	20:40:04.211	Thread Grou...	HTTP Request	27	✓	5244	152	27	2
24	20:40:04.310	Thread Grou...	HTTP Request	15	✓	5244	152	15	1
25	20:40:04.412	Thread Grou...	HTTP Request	21	✓	5244	152	21	2
26	20:40:04.514	Thread Grou...	HTTP Request	41	✓	5244	152	41	4
27	20:40:04.610	Thread Grou...	HTTP Request	18	✓	5244	152	18	1
28	20:40:04.714	Thread Grou...	HTTP Request	17	✓	5244	152	17	4
29	20:40:04.812	Thread Grou...	HTTP Request	13	✓	5244	152	12	1

Scroll automatically? Child samples? No of Samples 100 Latest Sample 15 Average 15 Deviation 5



## Bonus:

- Membuat error handler untuk 404, halaman yang tidak ditemukan. Menjadi 404.html pada template/error.
- Membuat beberapa validasi untuk fitur-fitur post dengan meretrieve pada halaman “not-found”

## Lesson Learned:

Pada proyek ini saya mempelajari proses pembuatan sebuah Sistem Informasi. Mulai dari pembuatan proyek, integrasi dengan database, serta membuat fitur-fitur pada sistem, yaitu Insert, Select, serta Update. Pada tugas ini juga saya banyak nostalgia dengan berbagai syntax-syntax java pada SDA, untuk membantu Saya dalam penggerjaan proyek. Pada tutorial ini juga saya banyak melakukan pencarian pada stackoverflow serta template-template bootstrap seperti versi 4 nya ini. Untuk beberapa framework saya menggunakannya secara online menggunakan CDN.