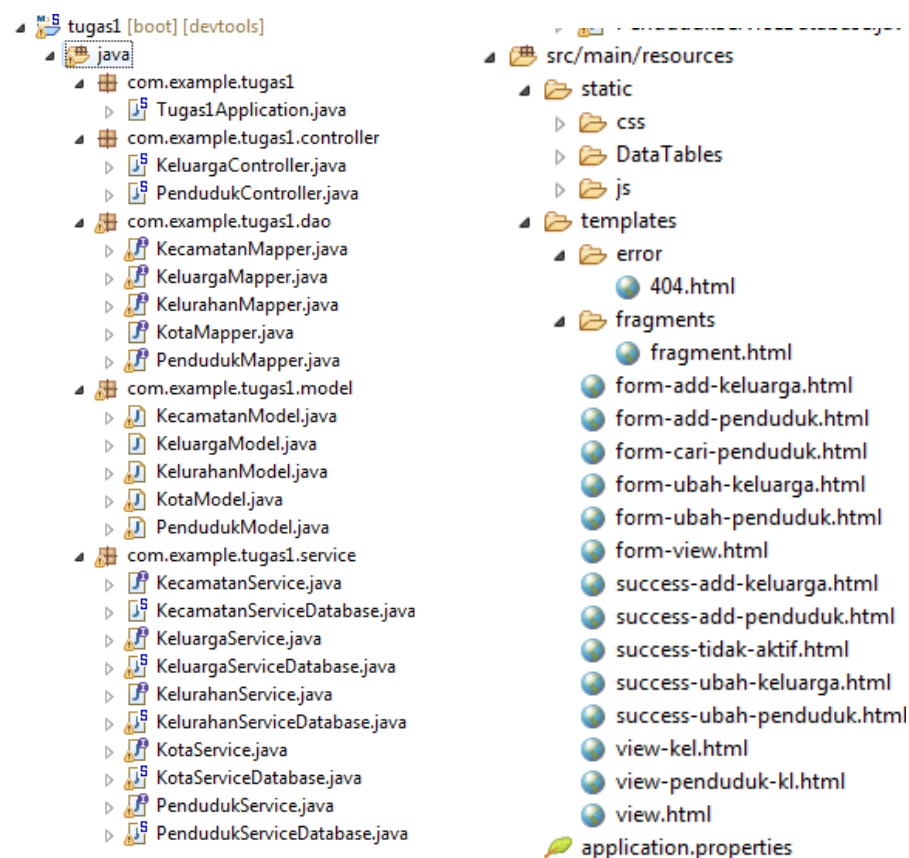


PROSES Pengerjaan

Pengerjaan dimulai dengan setting database, yaitu nama yang dipakai adalah tugas1. Setelah mengimport database, saya membuat model dari semua tabel yang ada beserta atribut-atributnya.

Setelah itu, ketika ingin membuat fungsi, memulai dengan membuat request mapping pada controller. Jika melibatkan database, maka saya kemudian membuat mapper dan service terlebih dahulu, lalu mengimplementasi fungsi. Setelah itu, mengirim variabel yang dibutuhkan ke view, dan ditampilkan.

STRUKTUR PROJECT DAN PACKAGE



IMPLEMENTASI MVC

Pengerjaan tugas ini dilakukan menggunakan model MVC, yang terdiri dari 3 komponen utama yaitu Model, View, dan Controller.

Model berisi representasi dari objek-objek yang ada, lengkap dengan atribut-atribut yang dimiliki, serta method setter getter untuk semua atribut yang tergenerate otomatis.

Lalu ada Controller yang akan memanggil fungsi sesuai halaman yang dipanggil. Saya membuat dua controller yaitu penduduk controller untuk fungsi yang berkaitan dengan penduduk, dan keluarga controller untuk fungsi yang berkaitan dengan keluarga. Terdapat juga Mapper, yang berfungsi untuk menjalankan query ke database. Mapper dipanggil di Controller, melalui Service. Saya membuat mapper dan service untuk semua tabel, dan mengelompokkan mapper berdasarkan akses query ke tabel.

Untuk view, terdapat folder static untuk menyimpan sources seperti bootstrap dan data tables. Terdapat juga templates yang berisi folder error untuk mengatasi error, folder fragment untuk menyimpan kode yang bisa digunakan kembali, dan view untuk halaman-halaman yang ada.

Dengan penggunaan layering seperti MVC, kode akan lebih terstruktur dan mendukung enkapsulasi.

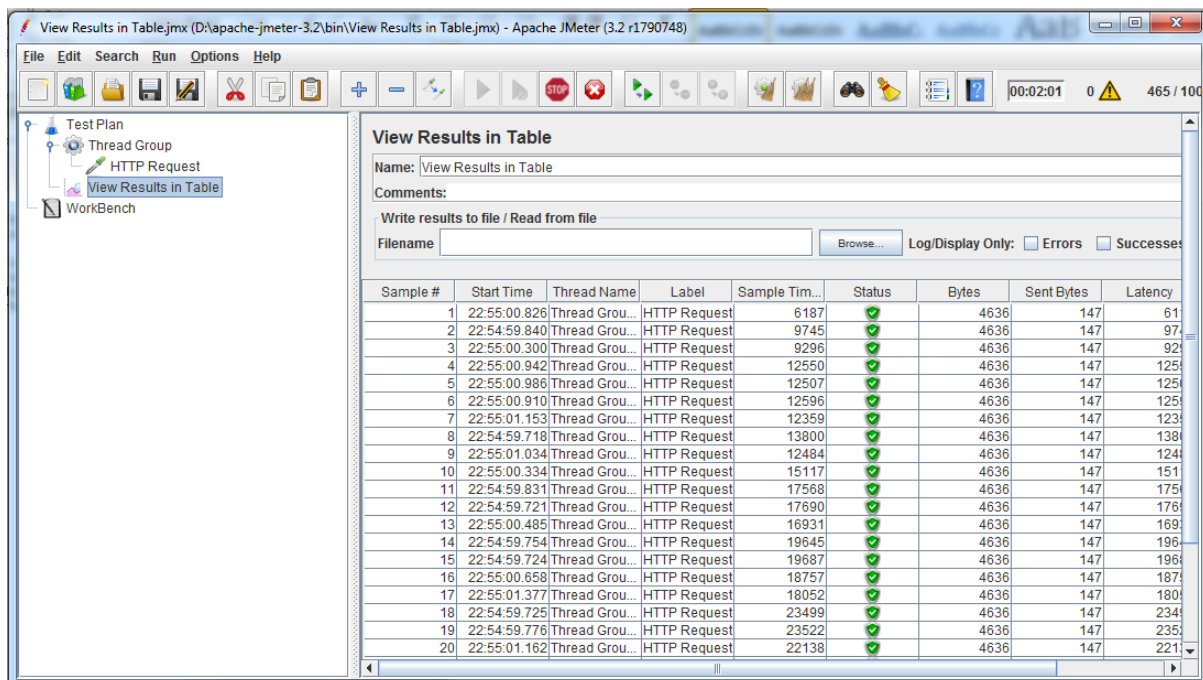
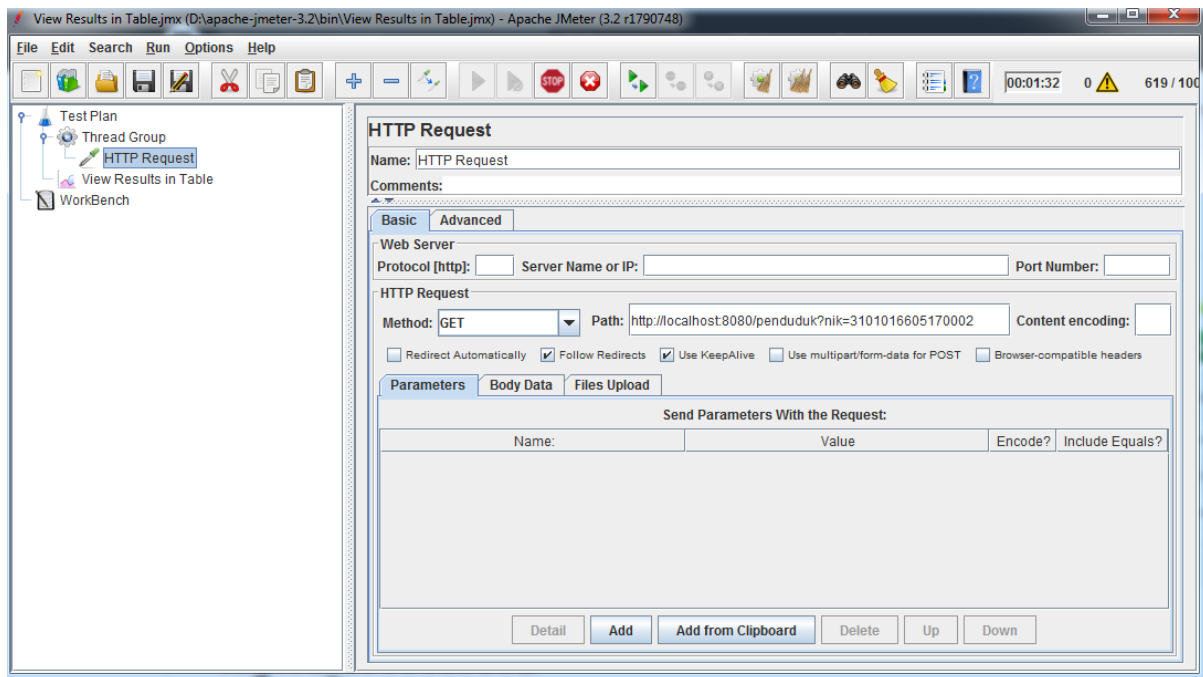
OPTIMISASI DATABASE

Optimisasi database dilakukan dengan beberapa cara, antarlain:

- Primary key
Menjadikan id unik dalam tabel menjadi primary key. Dalam hal ini, primary key untuk semua tabel adalah kolom "id" yang ada di setiap tabel. Dengan primary key, pencarian pada data yang banyak akan lebih menghemat waktu.
- Indexing
Menjadikan atribut sebagai index, untuk mempercepat proses pencarian yang dilakukan, terutama dengan pencarian yang melibatkan banyak data. Atribut-atribut yang akan diakses untuk pencarian, dijadikan index.
- Auto Increment
Auto increment diterapkan pada primary key yaitu ID, untuk setiap tabel. Dengan Auto increment, maka id akan ter-increment secara otomatis sehingga primary key tidak perlu di set kembali secara manual, sehingga mempercepat waktu.

STRESS TESTING

- Lihat penduduk



- Lihat Keluarga

ble

HTTP Request

Name: HTTP Request

Comments:

Basic Advanced

Web Server

Protocol [http]: Server Name or IP: Port Number:

HTTP Request

Method: GET Path: http://localhost:8080/keluarga?nkk=3101010205100001 Content encoding:

☐ Redirect Automatically ☒ Follow Redirects ☒ Use KeepAlive ☐ Use multipart/form-data for POST ☐ Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value	Encode?	Include Equals?
-------	-------	---------	-----------------

Detail Add Add from Clipboard Delete Up Down

View Results in Table.jmx (D:\apache-jmeter-3.2\bin\View Results in Table.jmx) - Apache JMeter (3.2 r1790748)

File Edit Search Run Options Help

00:00:49 0 0 / 1000

Test Plan

- Thread Group
 - HTTP Request
 - View Results in Table
 - View Results in Table
- WorkBench

View Results in Table

Name: View Results in Table

Comments:

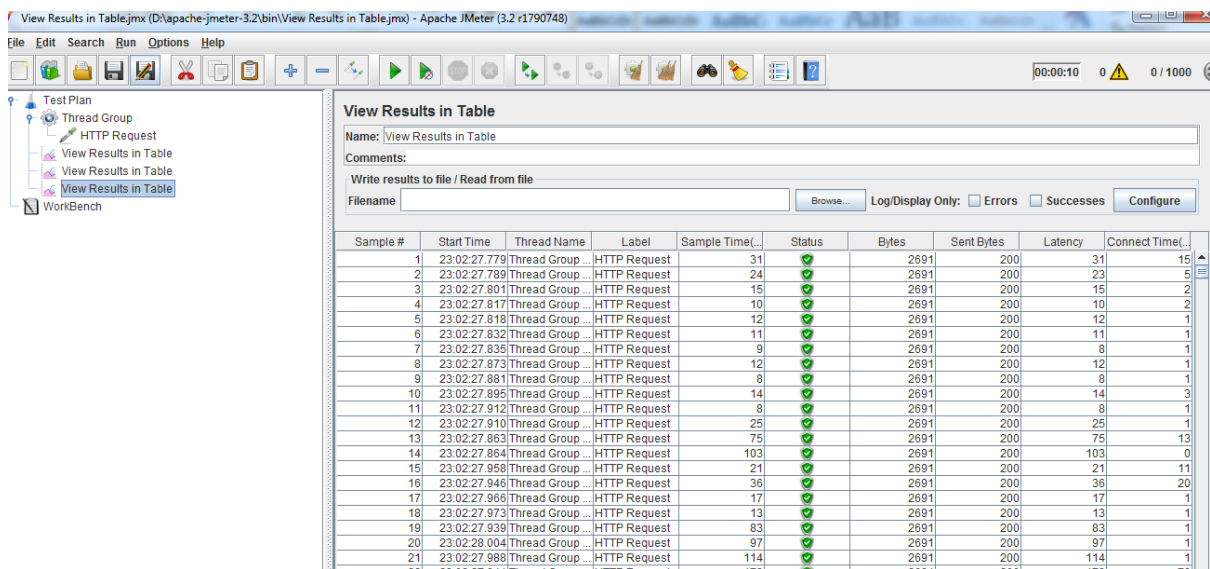
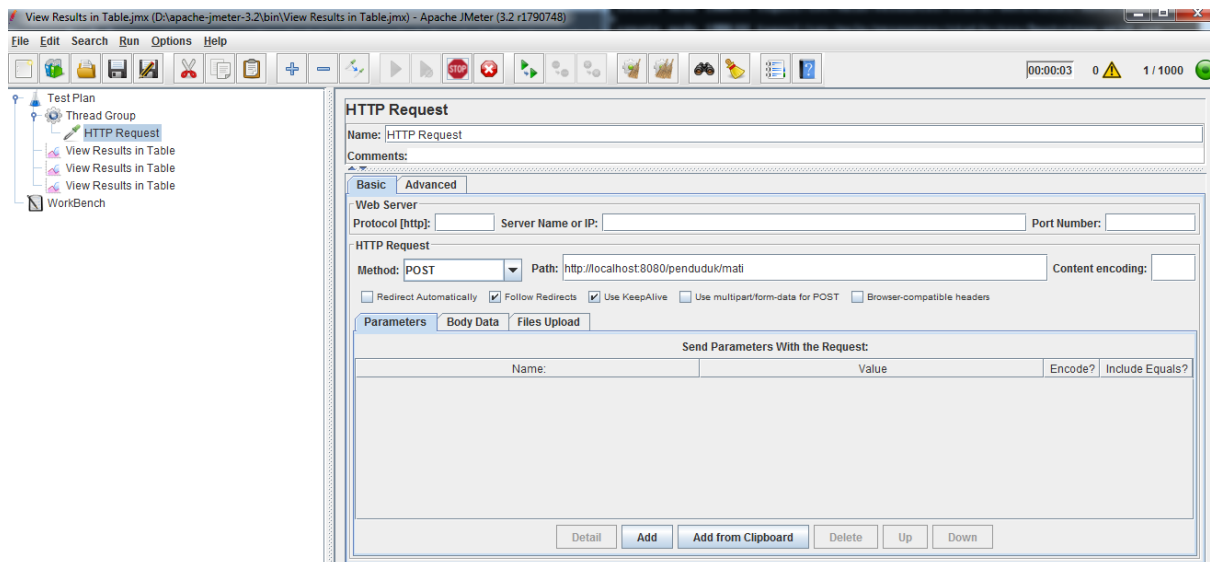
Write results to file / Read from file

Filename: Browse...

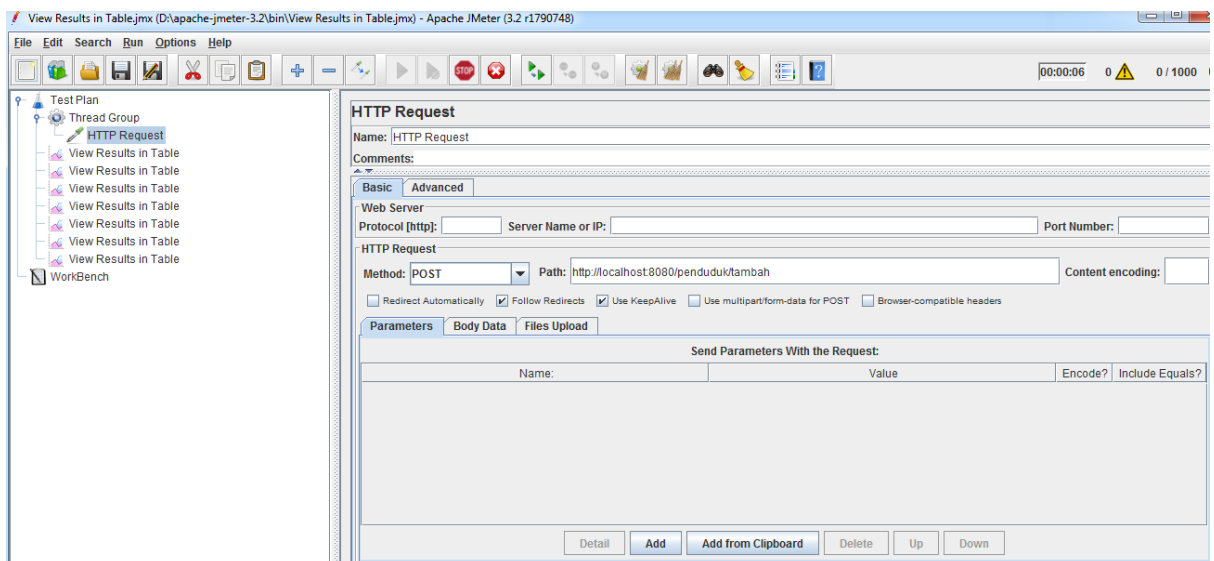
Log/Display Only: ☐ Errors ☐ Successes Configure

Sample #	Start Time	Thread Name	Label	Sample Time(Status	Bytes	Sent Bytes	Latency	Connect Time(
887	23:00:09.838	Thread Group	HTTP Request	40538	✓	6499	147	40538	0
888	23:00:11.132	Thread Group	HTTP Request	39245	✓	6499	147	39245	1
889	23:00:11.089	Thread Group	HTTP Request	39296	✓	6499	147	39296	1
890	23:00:12.181	Thread Group	HTTP Request	38226	✓	6499	147	38226	0
891	23:00:11.176	Thread Group	HTTP Request	39232	✓	6499	147	39232	1
892	23:00:09.584	Thread Group	HTTP Request	40825	✓	6499	147	40825	0
893	23:00:12.056	Thread Group	HTTP Request	38361	✓	6499	147	38358	1
894	23:00:11.981	Thread Group	HTTP Request	38439	✓	6499	147	38439	0
895	23:00:11.338	Thread Group	HTTP Request	39086	✓	6499	147	39086	1
896	23:00:12.099	Thread Group	HTTP Request	38355	✓	6499	147	38354	1
897	23:00:10.188	Thread Group	HTTP Request	40266	✓	6499	147	40265	1
898	23:00:10.954	Thread Group	HTTP Request	39501	✓	6499	147	39500	0
899	23:00:09.990	Thread Group	HTTP Request	40465	✓	6499	147	40465	1
900	23:00:11.780	Thread Group	HTTP Request	38674	✓	6499	147	38674	1
901	23:00:09.785	Thread Group	HTTP Request	40725	✓	6499	147	40725	1
902	23:00:09.866	Thread Group	HTTP Request	40649	✓	6499	147	40649	1
903	23:00:09.987	Thread Group	HTTP Request	40535	✓	6499	147	40535	1
904	23:00:09.999	Thread Group	HTTP Request	40530	✓	6499	147	40530	1
905	23:00:09.797	Thread Group	HTTP Request	40752	✓	6499	147	40752	1
906	23:00:12.186	Thread Group	HTTP Request	40496	✓	6499	147	40496	1
907	23:00:10.292	Thread Group	HTTP Request	42390	✓	6499	147	42390	1
908	23:00:11.821	Thread Group	HTTP Request	40861	✓	6499	147	40861	1

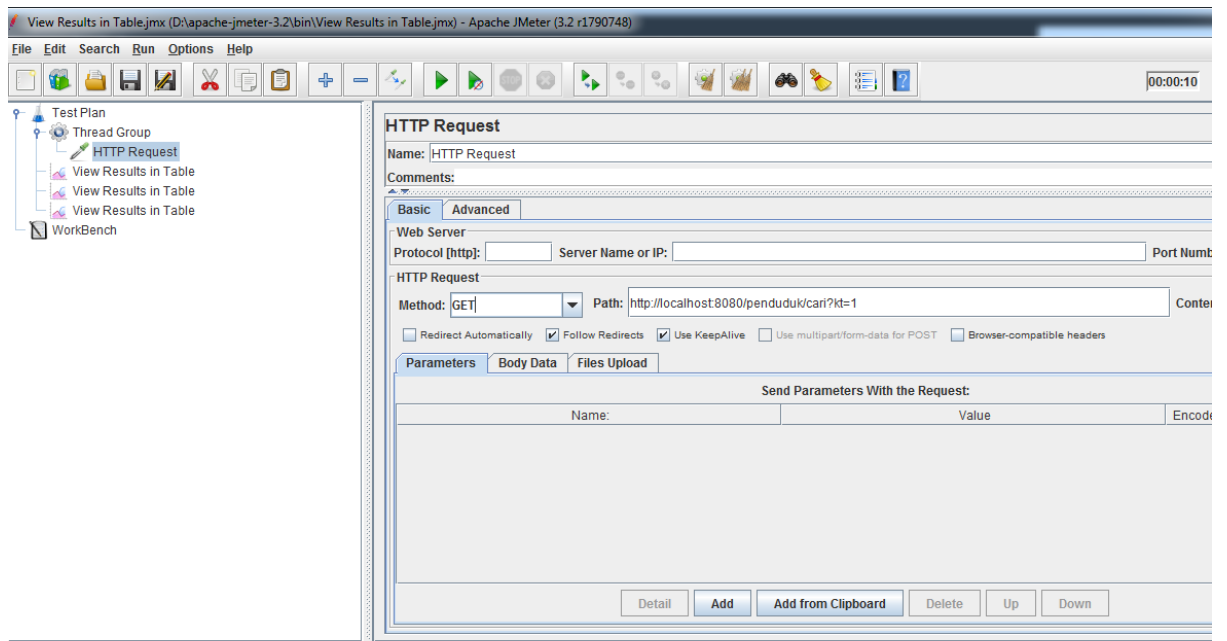
- Menon-aktifkan penduduk



- Menambah Penduduk



- Cari Berdasarkan Kota



View Results in Table.jmx (D:\apache-jmeter-3.2\bin\View Results in Table.jmx) - Apache JMeter (3.2 r1790748)

le Edit Search Run Options Help

00:00:12 0 0 / 1000

Test Plan

- Thread Group
 - HTTP Request
 - View Results in Table
 - View Results in Table
 - View Results in Table
 - View Results in Table
- WorkBench

View Results in Table

Name: View Results in Table

Comments:

Write results to file / Read from file

Filename: Browse...

Log/Display Only: ☐ Errors ☐ Successes

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time(...)
1	23:04:30.786	Thread Group ...	HTTP Request	483	✓	3696	136	483	1
2	23:04:30.744	Thread Group ...	HTTP Request	533	✓	3696	136	533	17
3	23:04:30.767	Thread Group ...	HTTP Request	525	✓	3696	136	522	1
4	23:04:30.753	Thread Group ...	HTTP Request	543	✓	3696	136	543	8
5	23:04:30.778	Thread Group ...	HTTP Request	521	✓	3696	136	521	0
6	23:04:30.793	Thread Group ...	HTTP Request	527	✓	3696	136	527	1
7	23:04:30.814	Thread Group ...	HTTP Request	586	✓	3696	136	586	0
8	23:04:30.856	Thread Group ...	HTTP Request	595	✓	3696	136	595	0
9	23:04:30.933	Thread Group ...	HTTP Request	519	✓	3696	136	519	1
10	23:04:30.839	Thread Group ...	HTTP Request	615	✓	3696	136	614	1
11	23:04:30.948	Thread Group ...	HTTP Request	506	✓	3696	136	505	7
12	23:04:30.867	Thread Group ...	HTTP Request	589	✓	3696	136	589	1
13	23:04:30.936	Thread Group ...	HTTP Request	522	✓	3696	136	522	19
14	23:04:31.057	Thread Group ...	HTTP Request	402	✓	3696	136	402	1
15	23:04:30.950	Thread Group ...	HTTP Request	509	✓	3696	136	509	6
16	23:04:30.958	Thread Group ...	HTTP Request	502	✓	3696	136	502	1
17	23:04:30.956	Thread Group ...	HTTP Request	506	✓	3696	136	505	1
18	23:04:30.946	Thread Group ...	HTTP Request	517	✓	3696	136	517	19
19	23:04:30.948	Thread Group ...	HTTP Request	517	✓	3696	136	517	8
20	23:04:31.098	Thread Group ...	HTTP Request	367	✓	3696	136	367	1
21	23:04:31.107	Thread Group ...	HTTP Request	360	✓	3696	136	359	0
22	23:04:30.970	Thread Group ...	HTTP Request	400	✓	3696	136	400	1

- Cari Berdasarkan Kecamatan

The top screenshot shows the Apache JMeter 3.2 interface with the 'HTTP Request' configuration window. The 'Basic' tab is active, showing the following details:

- Name:** HTTP Request
- Protocol [http]:** http
- Server Name or IP:** localhost.8080
- Path:** penduduk/cari?kt=1&kc=1
- Method:** GET
- Options:** ☐ Redirect Automatically, ☒ Follow Redirects, ☒ Use KeepAlive, ☐ Use multipart/form-data for POST, ☐ Browser-compatible headers

The bottom screenshot shows the 'View Results in Table' window, displaying a table of test results for the 'HTTP Request'.

Sample #	Start Time	Thread Name	Label	Sample Time	Status	Bytes	Sent Bytes	Latency	Connect Time
1	23:06:35.879	Thread Group ...	HTTP Request	217	✓	3881	141	217	5
2	23:06:35.951	Thread Group ...	HTTP Request	145	✓	3881	141	145	1
3	23:06:35.974	Thread Group ...	HTTP Request	137	✓	3881	141	137	0
4	23:06:35.972	Thread Group ...	HTTP Request	141	✓	3881	141	140	1
5	23:06:35.841	Thread Group ...	HTTP Request	276	✓	3881	141	276	49
6	23:06:35.837	Thread Group ...	HTTP Request	281	✓	3881	141	281	46
7	23:06:35.920	Thread Group ...	HTTP Request	199	✓	3881	141	199	1
8	23:06:36.094	Thread Group ...	HTTP Request	53	✓	3881	141	53	1
9	23:06:36.023	Thread Group ...	HTTP Request	130	✓	3881	141	130	1
10	23:06:35.961	Thread Group ...	HTTP Request	195	✓	3881	141	195	1
11	23:06:35.932	Thread Group ...	HTTP Request	343	✓	3881	141	343	1
12	23:06:36.093	Thread Group ...	HTTP Request	186	✓	3881	141	186	1
13	23:06:35.868	Thread Group ...	HTTP Request	412	✓	3881	141	412	17
14	23:06:36.053	Thread Group ...	HTTP Request	229	✓	3881	141	229	1
15	23:06:36.123	Thread Group ...	HTTP Request	188	✓	3881	141	188	1
16	23:06:35.881	Thread Group ...	HTTP Request	467	✓	3881	141	466	3
17	23:06:36.119	Thread Group ...	HTTP Request	241	✓	3881	141	241	1
18	23:06:36.001	Thread Group ...	HTTP Request	362	✓	3881	141	361	75
19	23:06:36.174	Thread Group ...	HTTP Request	196	✓	3881	141	196	1
20	23:06:36.039	Thread Group ...	HTTP Request	333	✓	3881	141	333	1
21	23:06:36.204	Thread Group ...	HTTP Request	170	✓	3881	141	170	1
22	23:06:36.172	Thread Group ...	HTTP Request	202	✓	3881	141	202	0

Kesimpulan: Meskipun melibatkan data yang banyak, dan di test untuk digunakan oleh banyak user, karena sudah melakukan beberapa optimisasi database, proses dapat berjalan dengan efisien.

BONUS

Menampilkan Penduduk Paling Muda dan Paling Tua di Suatu Kelurahan

Menambahkan method untuk mencari penduduk tertua atau termuda di suatu kelurahan. Query di kelurahanMapper:

```



```

```



```

Setelah itu, menambahkan method ke interface kelurahanService dan class kelurahanServiceDatabase. Hasil dari query disimpan ke dalam list karena ada kemungkinan terdapat lebih dari 1 orang dengan tanggal lahir yang sama. Selanjutnya, service dipanggil di controller dan hasil list penduduk model tertua dan termuda akan dikirim ke view melalui attribute model.

```

List<PendudukModel> termuda = kelurahanDAO.selectPendudukTermuda(kl);
List<PendudukModel> tertua = kelurahanDAO.selectPendudukTertua(kl);

model.addAttribute("termuda", termuda);
model.addAttribute("tertua", tertua);

return "view-penduduk-kl";

```


Di view, list penduduk yang ada di model akan di iterasi untuk ditampilkan semua penduduk termuda dan tertua dalam tabel.

localhost:8080/penduduk/cari?kt=1&kc=1&kl=1

Penduduk Dengan Usia Termuda

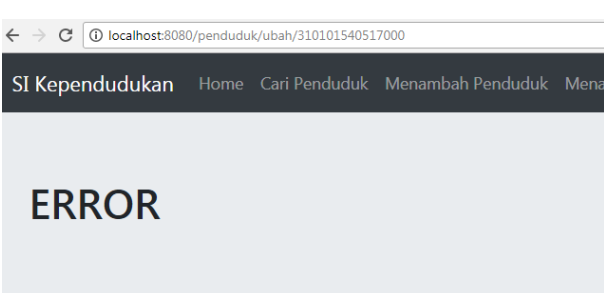
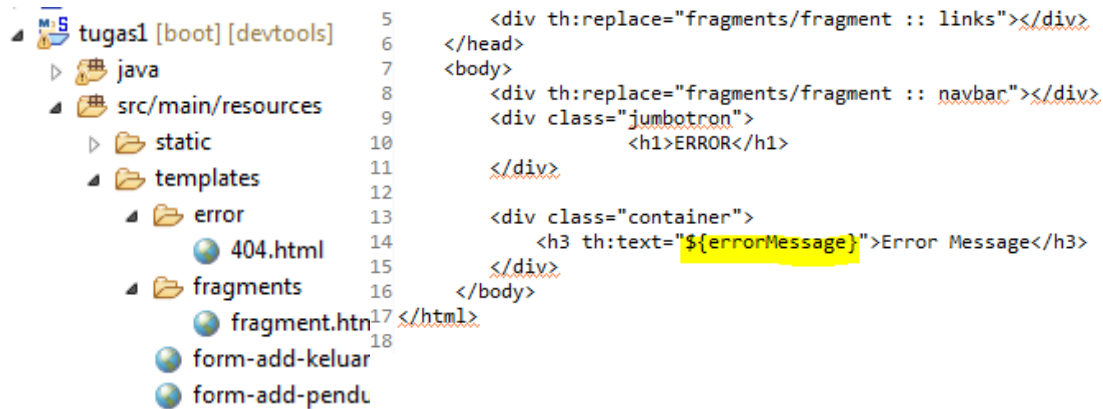
No	NIK	Nama Lengkap	Tanggal Lahir
1	3101015110170001	oje	2017-10-11

Penduduk Dengan Usia Tertua

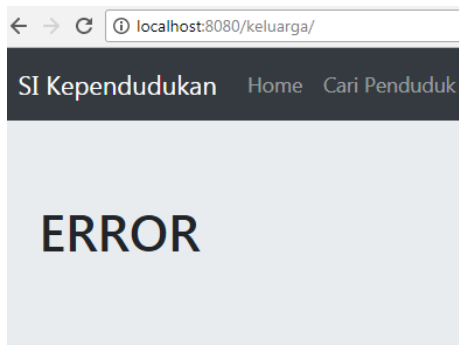
No	NIK	Nama Lengkap	Tanggal Lahir
1	3174046409520001	Hilda Agustina	1952-09-24
2	3173046409520001	Diana Mulyani M.Kom.	1952-09-24
3	3172022409520001	Bakiman Lantar Utama	1952-09-24

Menambahkan Error Page

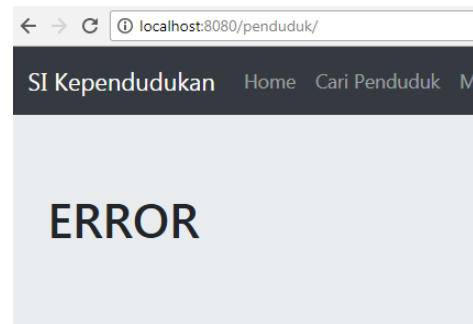
Menambahkan view error page yang akan menampilkan error message, sesuai error yang terjadi. errorMessage di kirim melalui model, dari penduduk dan keluarga Controller.



NIK 310101540517000 Tidak Ditemukan



NKK Tidak Bisa Kosong



NIK Tidak Bisa Kosong