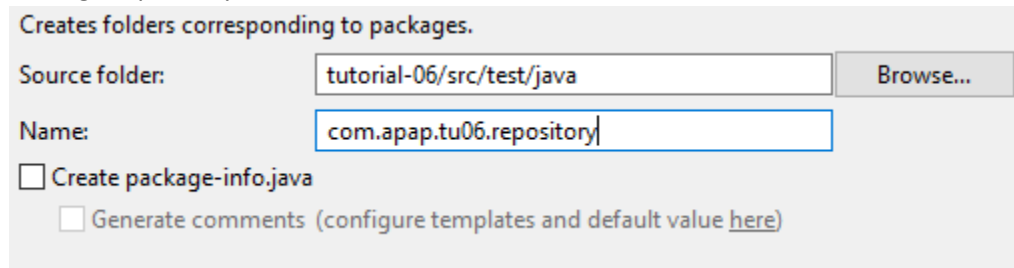


Nama : Izzatunnisa Arwadi

NPM : 1806269726

3. Testing untuk JPA

- Package repository



Creates folders corresponding to packages.

Source folder:

Name:

☐ Create package-info.java

☐ Generate comments (configure templates and default value [here](#))

Dilakukan pembuatan package repository pada src/test/java

- Class FlightDbTest

```
@RunWith(SpringRunner.class)
@DataJpaTest
@AutoConfigureTestDatabase(replace = Replace.NONE)
public class FlightDbTest {
    @Autowired
    private TestEntityManager entityManager;

    @Autowired
    private FlightDb flightDb;

    @Test
    public void whenFindByFlightNumber_thenReturnFlight() {
        //Given
        PilotModel pilotModel = new PilotModel();
        pilotModel.setLicenseNumber("1234");
        pilotModel.setName("Anto");
        pilotModel.setFlyHour(50);
        entityManager.persist(pilotModel);
        entityManager.flush();
        FlightModel flightModel = new FlightModel();
        flightModel.setFlightNumber("1765");
        flightModel.setOrigin("Jakarta");
        flightModel.setDestination("Bali");
        flightModel.setTime(new Date(new java.util.Date().getTime()));
        flightModel.setPilot(pilotModel);
```

Dilakukan pembuatan class FlightDbTest dengan isi kode seperti di atas.

- 1) Mengapa perlu menginisiasi object PilotModel, sedangkan yang di test hanya FlightModel?

Answer:

Karena terdapat hubungan pada FlightModel dengan PilotModel. Pada FlightModel terdapat juga field pilot yakni license number sehingga kedua model tersebut bisa saling berhubungan

- 2) Jelaskan apa yang akan terjadi jika object PilotModel dihapus dan tidak dilakukan setPilot pada FlightModel?

Answer:

Maka object FlightModel tidak dapat dibuat juga. Hal ini dikarenakan FlightModel memiliki ketergantungan pada object PilotModel. Oleh karena itu object PilotModel harus dibuat terlebih dahulu agar object FlightModel bisa dibuat.

4. Testing untuk Service

- Package service

Creates folders corresponding to packages.

Source folder:

Name:

☐ Create package-info.java

☐ Generate comments (configure templates and default value [here](#))

Membuat package service pada src/test/java

- Class FlightServiceTest

```
@RunWith(SpringRunner.class)
public class FlightServiceTest {
    @Autowired
    private FlightService flightService;

    @MockBean
    private FlightDb flightDb;

    @TestConfiguration //Membatasi scope Bean yang didefinisikan menjadi local class
    static class FlightServiceTestContextConfiguration{
        @Bean //Initiate flightService sebagai Bean
        public FlightService flightService() {
            return new FlightServiceImpl();
        }
    }

    @Test
    public void whenValidFlightNumber_thenFlightShouldBeFound() {
        //Given
        FlightModel flightModel = new FlightModel();
        flightModel.setFlightNumber("I765");
        flightModel.setOrigin("Jakarta");
        flightModel.setDestination("Bali");
        flightModel.setTime(new Date(new java.util.Date().getTime()));
        Optional<FlightModel> flight = Optional.of(flightModel);
        Mockito.when(flightDb.findByFlightNumber(flight.get().getFlightNumber())).thenReturn(flight);
    }
}
```

Dilakukan pembuatan class FlightServiceTest dengan isi kode seperti di atas.

3) Jelaskan apa yang dilakukan oleh code

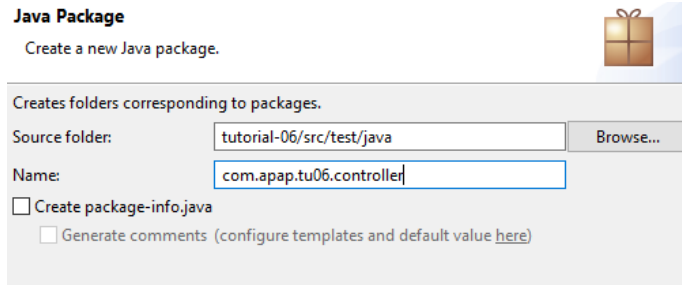
Mockito.when(flightDb.findByFlightNumber(flight.get().getFlightNumber())).thenReturn(flight);

Answer:

Penggunaan Mockito berfungsi untuk pengujian atau testing framework. Mockito memungkinkan untuk melakukan mock object pada class serta melakukan operation didalamnya. Potongan kode di atas menjelaskan bahwa ketika Mockito melakukan pencarian Flight berdasarkan FlightNumber dengan menggunakan method FlightDb.findByFlightNumber maka akan dilakukan return object flight tersebut.

5. Testing untuk Controller

- Package controller



Membuat package controller pada src/test/java

- FlightControllerTest

```
@RunWith(SpringRunner.class)
@WebMvcTest(FlightController.class)
public class FlightControllerTest {
    @Autowired
    private MockMvc mvc;

    @MockBean
    private FlightService flightService;

    @MockBean
    private PilotService pilotService;

    @Test
    public void givenFlightNumber_whenViewFlight_thenReturnJsonFlight() throws Exception {
        //Given
        FlightModel flightModel = new FlightModel();
        flightModel.setFlightNumber("I765");
        flightModel.setOrigin("Jakarta");
        flightModel.setDestination("Bali");
        flightModel.setTime(new Date(new java.util.Date().getTime()));
        Optional<FlightModel> flight = Optional.of(flightModel);
        Mockito.when(flightService.getFlightDetailByFlightNumber(flight.get().getFlightNumber())).thenReturn(flight);

        //When
        mvc.perform(MockMvcRequestBuilders.get("/flight/view")
            .param("flightNumber", flight.get().getFlightNumber())
            .contentType(MediaType.APPLICATION_JSON))
```

- Dilakukan pembuatan class FlightControllerTest dengan isi kode seperti di atas.

4) Jelaskan apa yang dilakukan oleh code

`Mockito.when(flightService.getFlightDetailByFlightNumber(flight.get().getFlightNumber())).thenReturn(flight);`

Answer:

Potongan kode di atas menjelaskan bahwa ketika Mockito ingin mendapatkan flight detail dengan menggunakan method `flightService.getFlightDetailByFlightNumber` maka akan dilakukan return object flight berdasarkan flight number yang telah didapatkan.

5) Jelaskan apa yang di-test oleh code

`.andExpect(MockMvcResultMatchers.status().isOk())`

Answer:

Potongan code di atas menunjukkan ketika dilakukan test pada `"/flight/view"` maka expectation dari test tersebut berupa verify status `MockMvcResultMatchers` dengan menyatakan bahwa response status code adalah `HttpStatus.OK (200)`

6) Jelaskan apa yang di-test oleh code

```
.andExpect(MockMvcResultMatchers.jsonPath("$.flightNumber",  
Matchers.is(flight.get().getFlightNumber())));
```

Answer:

Potongan code di atas menunjukkan ketika dilakukan test pada `"/flight/view"` maka expectation dari test tersebut untuk melakukan verify bahwa pernyataan yang berada di dalam method `MockMvcResultMatchers.jsonPath` itu benar. Pada code didalam method tersebut dilakukan pencocokan antara `$.flightNumber` dengan hasil method dari `flight.get().getFlightNumber()`, kedua value tersebut harus sama (match).

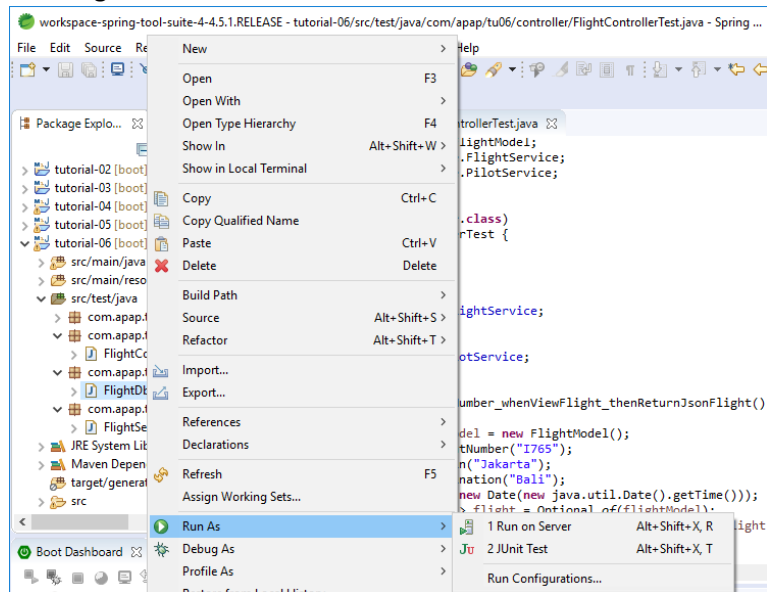
7) Jelaskan anotasi `@ResponseBody` yang ada pada route `"/flight/view"`

Answer:

`@ResponseBody` merupakan anotasi pada Spring yang dapat mengikat method yang memiliki nilai return ke web response body. Nilai return akan dilakukan konversi ke HTTP response body dengan bantuan HTTP Message converter. Pada potongan code method di atas, akan dilakukan pengiriman value dari object archive ke web response body

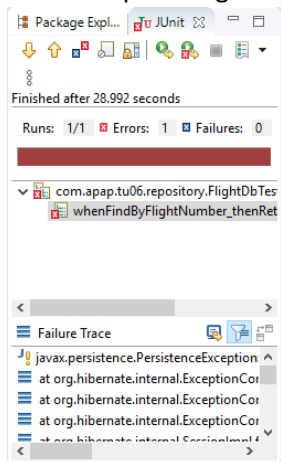
6. Melakukan Testing (Unit Testing)

- Running JUnit



Dilakukan running JUnit dengan melakukan klik kanan pada class >> Run As >> JUnit Test

- Hasil test pada FlightDbTest



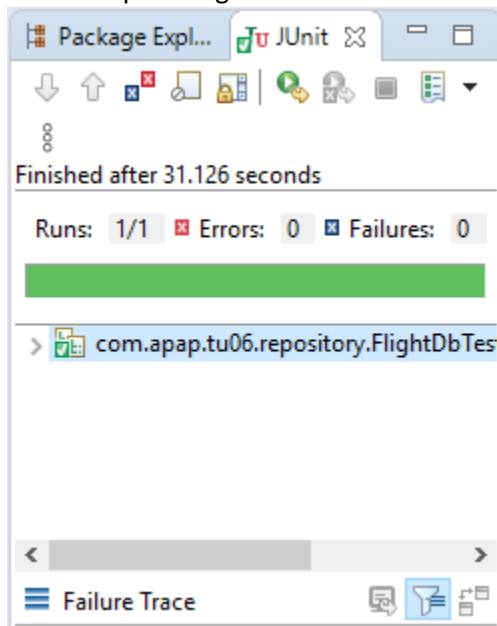
```
Hibernate: insert into pilot (fly_hour, license_number, name) values (?, ?, ?)
2020-03-10 15:35:35.122 WARN 6024 --- [main] o.h.engine.jdbc.spi.SqlExceptionHelper : SQL Error: 1062, SQLState: 23000
2020-03-10 15:35:35.123 ERROR 6024 --- [main] o.h.engine.jdbc.spi.SqlExceptionHelper : Duplicate entry '1234' for key 'UK_sk71fi4dx8k8a21mbd93
2020-03-10 15:35:35.207 INFO 6024 --- [main] o.s.t.c.transaction.TransactionContext : Rolled back transaction for test: [DefaultTestContext@3
2020-03-10 15:35:35.262 INFO 6024 --- [Thread-3] j.LocalContainerEntityManagerFactoryBean : Closing JPA EntityManagerFactory for persistence unit 'c
```

Ketika dilakukan test pada FlightDbTest menunjukkan adanya error. Error ini dikarenakan adanya duplicate entry dalam melakukan insert value ke table pilot di kolom license_number.

Ketika pada FlightDbTest dilakukan perubahan value license_number seperti di bawah ini

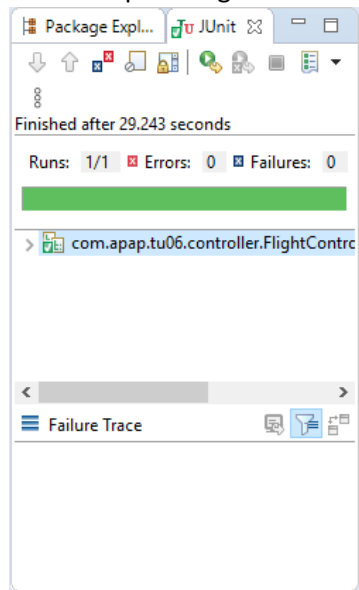
```
pilotModel.setLicenseNumber("1233");
```

Hasil test pada FlightDbTest



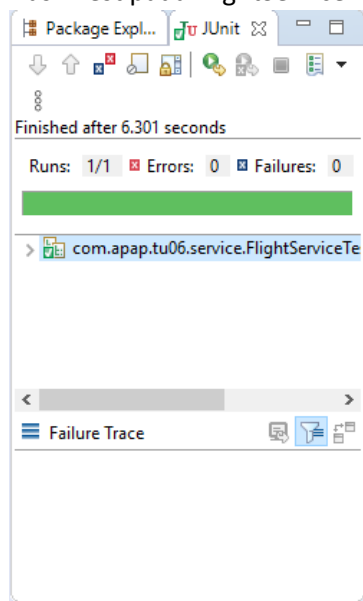
FlightDbTest menunjukkan bahwa tidak terdapat error maupun kesalahan

- Hasil test pada FlightControllerTest



Ketika dilakukan test pada FlightControllerTest menunjukkan bahwa tidak terdapat error maupun kesalahan

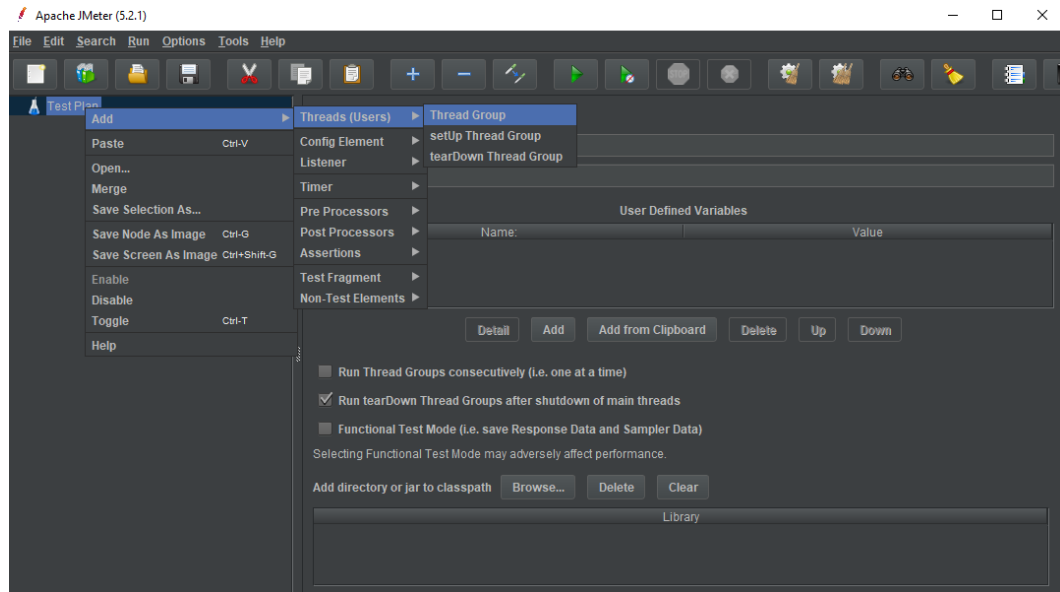
- Hasil Test pada FlightServiceTest



Ketika dilakukan test pada FlightServiceTest menunjukkan bahwa tidak terdapat error maupun kesalahan

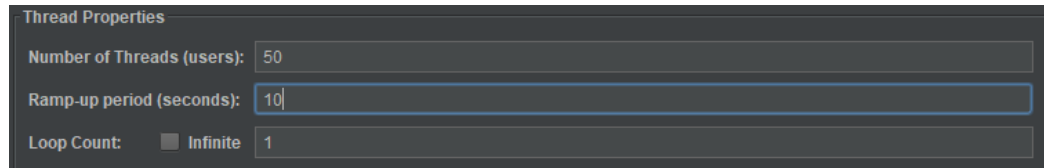
7. Load Testing

- Thread Group



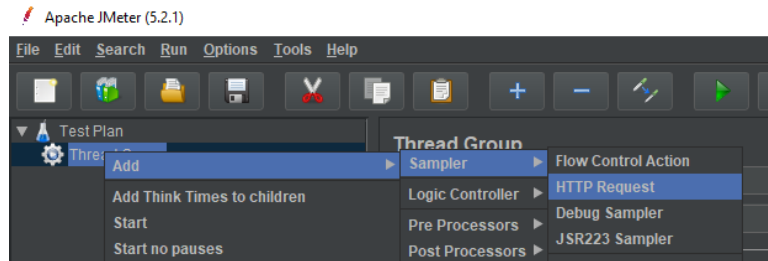
Melakukan penambahan test plan thread group

- Thread Properties

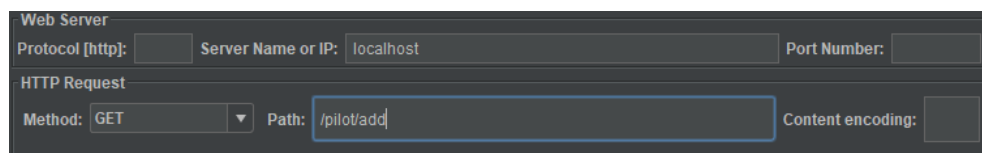


Melakukan pengisian properties pada number of threads, ram-up period dan loop-count

- Http Request

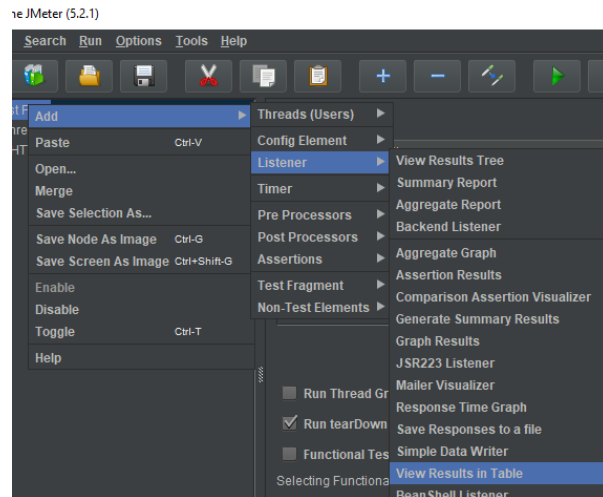


Untuk melakukan pengesetan http request dilakukan dengan cara Thread Group >> Add >> Sampler >> HTTP Request

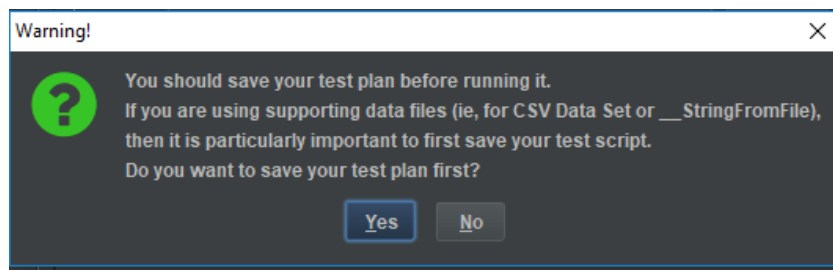


Dilakukan pengisian server name serta path

- Hasil



Untuk memantau hasil dapat dilakukan dengan klik kanan pada Test Plan >> Add >> Listener >> View Results in Table



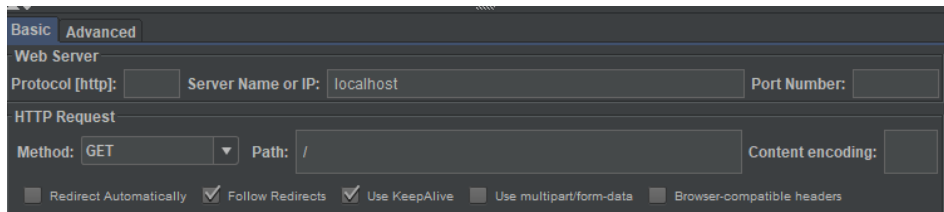
Ketika dilakukan run akan muncul notifikasi untuk menyimpan test plan

Sample #	Start Time	Thread Name	Label	Sample Time...	Status	Bytes	Sent Bytes	Latency
1	16:58:35.834	Thread Grou...	HTTP Request	1603	Failure	1482	123	1571
2	16:58:35.483	Thread Grou...	HTTP Request	1960	Failure	1482	123	195
3	16:58:36.058	Thread Grou...	HTTP Request	1389	Failure	1482	123	1381
4	16:58:35.241	Thread Grou...	HTTP Request	2211	Failure	1482	123	2211
5	16:58:36.206	Thread Grou...	HTTP Request	1246	Failure	1482	123	1241
6	16:58:35.236	Thread Grou...	HTTP Request	2226	Failure	1482	123	217
7	16:58:35.242	Thread Grou...	HTTP Request	2219	Failure	1482	123	2161
8	16:58:36.635	Thread Grou...	HTTP Request	829	Failure	1482	123	771
9	16:58:35.316	Thread Grou...	HTTP Request	2152	Failure	1482	123	215
10	16:58:35.656	Thread Grou...	HTTP Request	1813	Failure	1482	123	175
11	16:58:36.404	Thread Grou...	HTTP Request	1068	Failure	1482	123	1001
12	16:58:35.242	Thread Grou...	HTTP Request	2234	Failure	1482	123	2161
13	16:58:35.237	Thread Grou...	HTTP Request	2250	Failure	1482	123	217
14	16:58:37.002	Thread Grou...	HTTP Request	510	Failure	1482	123	461
15	16:58:36.839	Thread Grou...	HTTP Request	675	Failure	1482	123	63
16	16:58:35.237	Thread Grou...	HTTP Request	2289	Failure	1482	123	217
17	16:58:35.241	Thread Grou...	HTTP Request	2288	Failure	1482	123	2161
18	16:58:37.292	Thread Grou...	HTTP Request	345	Failure	1482	123	34
19	16:58:37.717	Thread Grou...	HTTP Request	28	Failure	1482	123	21

Pada view result table akan ditampilkan result seperti di atas

Percobaan

1. Percobaan Pertama



Basic Advanced

Web Server

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

HTTP Request

Method: GET Path: / Content encoding:

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

Dilakukan percobaan pada path / (index)



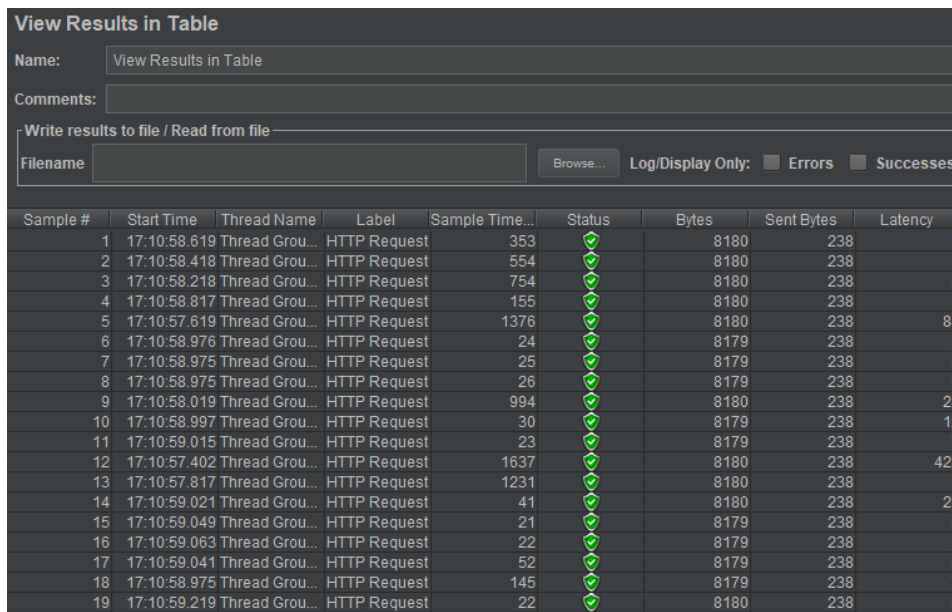
Thread Properties

Number of Threads (users): 25

Ramp-up period (seconds): 5

Loop Count: ☐ Infinite ☒ 2

Dilakukan pengisian properties pada number of threads=25, ram-up period=5 dan loop-count=2



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
1	17:10:58.619	Thread Grou...	HTTP Request	353	✓	8180	238	
2	17:10:58.418	Thread Grou...	HTTP Request	554	✓	8180	238	
3	17:10:58.218	Thread Grou...	HTTP Request	754	✓	8180	238	
4	17:10:58.817	Thread Grou...	HTTP Request	155	✓	8180	238	
5	17:10:57.619	Thread Grou...	HTTP Request	1376	✓	8180	238	84
6	17:10:58.976	Thread Grou...	HTTP Request	24	✓	8179	238	
7	17:10:58.975	Thread Grou...	HTTP Request	25	✓	8179	238	
8	17:10:58.975	Thread Grou...	HTTP Request	26	✓	8179	238	
9	17:10:58.019	Thread Grou...	HTTP Request	994	✓	8180	238	24
10	17:10:58.997	Thread Grou...	HTTP Request	30	✓	8179	238	12
11	17:10:59.015	Thread Grou...	HTTP Request	23	✓	8179	238	7
12	17:10:57.402	Thread Grou...	HTTP Request	1637	✓	8180	238	422
13	17:10:57.817	Thread Grou...	HTTP Request	1231	✓	8180	238	
14	17:10:59.021	Thread Grou...	HTTP Request	41	✓	8180	238	21
15	17:10:59.049	Thread Grou...	HTTP Request	21	✓	8179	238	
16	17:10:59.063	Thread Grou...	HTTP Request	22	✓	8179	238	
17	17:10:59.041	Thread Grou...	HTTP Request	52	✓	8179	238	
18	17:10:58.975	Thread Grou...	HTTP Request	145	✓	8179	238	
19	17:10:59.219	Thread Grou...	HTTP Request	22	✓	8180	238	

Ketika dilakukan percobaan pertama, didapatkan result seperti di atas dengan status success. Hal ini dikarenakan jumlah thread yang lebih sedikit.

2. Percobaan Kedua

Basic Advanced

Web Server

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

HTTP Request

Method: GET Path: /pilot/view Content encoding:

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

Parameters Body Data Files Upload

Dilakukan percobaan pada path /pilot/view

Thread Properties

Number of Threads (users): 30

Ramp-up period (seconds): 15

Loop Count: ☐ Infinite 3

Dilakukan pengisian properties pada number of threads=30, ram-up period=15 dan loop-count=3

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
1	17:15:21.435	Thread Grou...	HTTP Request	383	Warning	1482	124	383
2	17:15:21.819	Thread Grou...	HTTP Request	25	Warning	1481	124	25
3	17:15:21.845	Thread Grou...	HTTP Request	26	Warning	1481	124	26
4	17:15:21.936	Thread Grou...	HTTP Request	25	Warning	1482	124	25
5	17:15:21.962	Thread Grou...	HTTP Request	29	Warning	1481	124	29
6	17:15:21.992	Thread Grou...	HTTP Request	28	Warning	1481	124	28
7	17:15:22.437	Thread Grou...	HTTP Request	43	Warning	1482	124	43
8	17:15:22.481	Thread Grou...	HTTP Request	51	Warning	1481	124	51
9	17:15:22.533	Thread Grou...	HTTP Request	35	Warning	1481	124	35
10	17:15:22.934	Thread Grou...	HTTP Request	35	Warning	1482	124	35
11	17:15:22.969	Thread Grou...	HTTP Request	29	Warning	1481	124	29
12	17:15:22.999	Thread Grou...	HTTP Request	29	Warning	1481	124	29
13	17:15:23.431	Thread Grou...	HTTP Request	29	Warning	1482	124	29
14	17:15:23.461	Thread Grou...	HTTP Request	27	Warning	1481	124	27
15	17:15:23.489	Thread Grou...	HTTP Request	29	Warning	1481	124	29
16	17:15:23.929	Thread Grou...	HTTP Request	26	Warning	1482	124	26
17	17:15:23.955	Thread Grou...	HTTP Request	28	Warning	1481	124	28
18	17:15:23.984	Thread Grou...	HTTP Request	76	Warning	1481	124	76
19	17:15:24.428	Thread Grou...	HTTP Request	31	Warning	1482	124	31

Ketika dilakukan percobaan kedua, didapatkan result seperti di atas dengan status warning. Hal ini dikarenakan jumlah thread yang lebih banyak dibandingkan sebelumnya. Hal tersebut dapat mempengaruhi jalannya web

3. Percobaan Ketiga

Basic Advanced

Web Server

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

HTTP Request

Method: GET Path: /pilot/view Content encoding:

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

Parameters Body Data Files Upload

Masih dilakukan percobaan pada path /pilot/view

Thread Properties

Number of Threads (users): 10

Ramp-up period (seconds): 5

Loop Count: ☐ Infinite 2

Dilakukan pengisian properties pada number of threads=10, ram-up period=5 dan loop-count=2

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
1	17:23:11.071	Thread Grou...	HTTP Request	50		1482	124	
2	17:23:11.121	Thread Grou...	HTTP Request	33		1481	124	
3	17:23:11.571	Thread Grou...	HTTP Request	53		1482	124	
4	17:23:11.625	Thread Grou...	HTTP Request	32		1481	124	
5	17:23:12.069	Thread Grou...	HTTP Request	48		1482	124	
6	17:23:12.117	Thread Grou...	HTTP Request	40		1481	124	
7	17:23:12.588	Thread Grou...	HTTP Request	68		1482	124	
8	17:23:12.656	Thread Grou...	HTTP Request	23		1481	124	
9	17:23:13.073	Thread Grou...	HTTP Request	31		1482	124	
10	17:23:13.104	Thread Grou...	HTTP Request	27		1481	124	
11	17:23:13.513	Thread Grou...	HTTP Request	27		1482	124	
12	17:23:13.541	Thread Grou...	HTTP Request	44		1481	124	
13	17:23:14.072	Thread Grou...	HTTP Request	47		1482	124	
14	17:23:14.120	Thread Grou...	HTTP Request	21		1481	124	
15	17:23:14.573	Thread Grou...	HTTP Request	29		1482	124	
16	17:23:14.602	Thread Grou...	HTTP Request	42		1481	124	
17	17:23:15.075	Thread Grou...	HTTP Request	29		1482	124	
18	17:23:15.104	Thread Grou...	HTTP Request	25		1481	124	
19	17:23:15.512	Thread Grou...	HTTP Request	29		1482	124	
20	17:23:15.541	Thread Grou...	HTTP Request	25		1481	124	

Ketika jumlah thread, ram-up dan loop diturunkan masih mendapatkan hasil warning. Hal ini dikarenakan performa mesin yang kurang baik saat melakukan akses /pilot/view