

1. Screenshot hasil JMeter fitur soal latihan Load Testing nomor 1 sebelum dioptimasi (3 file)

- `jmeter_selectStudentBerhasil`

[illegible]

- `jmeter_selectStudentGagal`

View Results in Table										
Results - View Results in Table										
Comments										
Write results to the Read-Write file										
<div> <div>Resources</div> <div> Home Log/Display Only Errors Resources Outlines </div> </div>										
Runstep #	Start Time	Elapsed Time	Label	Runstep Threshold	Status	Notes	Send Status	Colored	Q	Colored Time (ms)
1	25.12.30.077	Thread Group 1-1	HTTP Request	2014	Success	2014	Q	Q	Q	2014
2	25.12.30.072	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
3	25.12.30.070	Thread Group 1-1	HTTP Request	3002	Success	2014	Q	Q	Q	3002
4	25.12.30.069	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
5	25.12.30.062	Thread Group 1-1	HTTP Request	2007	Success	2014	Q	Q	Q	2007
6	25.12.30.059	Thread Group 1-1	HTTP Request	2016	Success	2014	Q	Q	Q	2016
7	25.12.30.056	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
8	25.12.30.050	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
9	25.12.30.030	Thread Group 1-1	HTTP Request	2009	Success	2014	Q	Q	Q	2009
10	25.12.30.020	Thread Group 1-1	HTTP Request	2002	Success	2014	Q	Q	Q	2002
11	25.12.30.008	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
12	25.12.30.001	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
13	25.12.30.000	Thread Group 1-1	HTTP Request	2004	Success	2014	Q	Q	Q	2004
14	25.12.30.001	Thread Group 1-1	HTTP Request	2001	Success	2014	Q	Q	Q	2001
15	25.12.30.000	Thread Group 1-1	HTTP Request	2001	Success	2014	Q	Q	Q	2001
16	25.12.30.000	Thread Group 1-1	HTTP Request	2001	Success	2014	Q	Q	Q	2001
17	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
18	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
19	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
20	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
21	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
22	25.12.30.000	Thread Group 1-1	HTTP Request	2004	Success	2014	Q	Q	Q	2004
23	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
24	25.12.30.000	Thread Group 1-1	HTTP Request	2001	Success	2014	Q	Q	Q	2001
25	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
26	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
27	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
28	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
29	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
30	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
31	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
32	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
33	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
34	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
35	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
36	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
37	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
38	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
39	25.12.30.000	Thread Group 1-1	HTTP Request	2000	Success	2014	Q	Q	Q	2000
40	25.12.30.000	Thread Group 1-1	HTTP Request	2014	Success	2014	Q	Q	Q	2014

- `jmeter_selectAllStudents`

2. Screenshot hasil JMeter fitur soal latihan Load Testing nomor 1 sebelum dioptimasi (3 file)

- jmeter_selectStudentBerhasil
- jmeter_selectStudentGagal
- jmeter_selectAllStudents

Finished after 0,238 seconds.

```

public void deleteStudent() {
    //Given
    StudentModel studentModel = new StudentModel ("1506737823", "Chanek", 3.5);
    StudentModel check = new StudentModel("1506737823", "Chanek", 3.5);
    BDDMockito.given(studentService.deleteStudent("1506737823")).willReturn(true);

    //When
    boolean test=studentService.deleteStudent("1506737823");

    //Then
    BDDMockito.then(studentMapper).should().deleteStudent("1506737823");
    assertThat(test, equalTo(true)); // Check if Same
}

@Test
public void updateStudent() {
    //Given
    StudentModel studentModel = new StudentModel ("1506737823", "Chanek", 3.5);
    StudentModel check = new StudentModel("1506737823", "Chanek", 3.5);
    BDDMockito.given(studentService.updateStudent(studentModel)).willReturn(true);

    //When
    boolean test=studentService.updateStudent(studentModel);

    //Then
    BDDMockito.then(studentMapper).should().updateStudent(studentModel);
    assertThat(test, equalTo(true)); // Check if Same
}
}

```

Rangkuman

Melakukan testing sederhana terhadap service dengan plugin JUnit dan Mockito

Melakukan load testing sederhana dengan aplikasi Apache JMeter

melakukan load testing untuk mengukur performa web yang Anda buat. Load testing akan mengakses web Anda secara bersamaan (multithreading) dengan parameter yang telah ditentukan dan menghitung response time-nya. Pada tutorial ini kita akan menggunakan aplikasi Apache Jmeter untuk melakukan load testing