Tutorial 6

Nama : Sofyan Hadi

NPM: 1706107011

- Jelaskan apa saja hal yang Anda pelajari dari tutorial ini.
 Melakukan unit testing dan Load testing. Unit testing dilakukan dengan JUnit dan Mockito, ,melakukan pengecekan pada method yang dibuat. Load Testing dilakukan dengan Apache Jmeter, melakukan untuk mengukur performa dan respon dari server.
- 2. Jawab pertanyaan pada bagian Pertanyaan dan penjelasannya.

Latihan Unit Testing

Pertanyaan

a. Tambahkan code berikut pada StudentServiceDatabaseTest dan jelaskan apa yang terjadi pada method tersebut berikut dengan hasil testing-nya!

```
@Test
public void selectAllStudents() {
       // Given
       List<StudentModel> studentModels = new ArrayList<>();
       StudentModel studentModel = new StudentModel("1506737823", "Chanek", 3.5);
       studentModels.add(studentModel);
       List<StudentModel> checks = new ArrayList<>();
       StudentModel check = new StudentModel("1506737823", "Chanek", 3.5);
       checks.add(check);
        BDDMockito.given(studentMapper.selectAllStudents()).willReturn(studentModels);
       // When
       List<StudentModel> test = studentService.selectAllStudents();
       assertThat(test, notNullValue()); // Check if Not Null
       assertThat(test.isEmpty(), equalTo(false)); // Check kalo ngga kosong
       assertThat(test.size(), equalTo(1)); // Check if Size same
       assertThat(test, equalTo(checks)); // Check kalo konten sama
```

//Given

Inisialisasi list StudentModels dan checks, lalu inisialisasi object StudentModel dengan nama StudentModel dan check, lalu memasukan masing masing object ke dalam list. BDDMockito memanipulasi object dengan return stidentModels.

//When

Memasukan data ke dalam list test.

//Then

Dilakukan pengecekan Not Null, tidak kosong, ukuran sama, Konten sama.

b. Tambahkan code berikut pada StudentServiceDatabaseTest dan jelaskan apa yang terjadi pada method tersebut berikut dengan hasil testing-nya!

```
@Test
public void addStudent() {

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

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

    // Then
    BDDMockito.then(studentMapper).should().addStudent(check);
    assertThat(test, equalTo(true)); // Check if Same
}
```

//Given

Inisialisasi object StudentModel dengan nama StudentModel dan check. BDDMockito memanipulasi object dengan return True.

//When

Memasukan nilai boolean dari addStudent dan ke dalam test.

//Then

Dilakukan pengecekan jika sama.

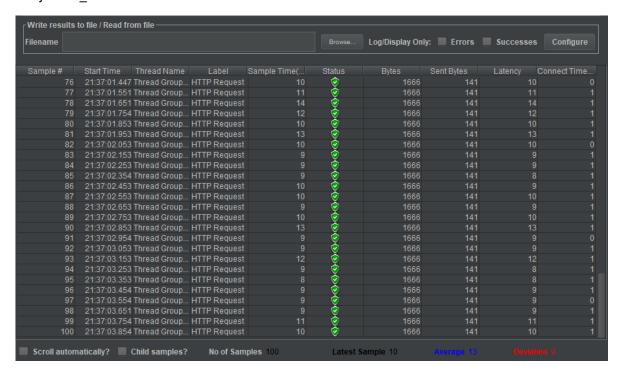
c. Lakukan testing terhadap method delete student dan update student!

```
com.example.service.StudentServiceDatabaseTest [Runner: JUnit 4] (0,245 s)

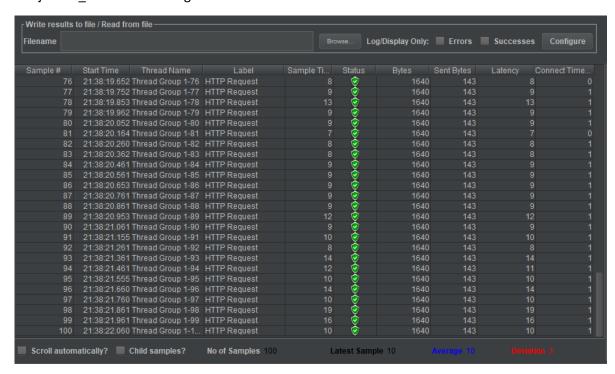
deleteStudent (0,228 s)
selectStudent (0,012 s)
selectAllStudents (0,003 s)
addStudent (0,000 s)
updateStudent (0,002 s)
```

Latihan Load Testing Pertanyaan

- a. Lakukan load testing pada semua kemungkinan hasil dari fitur selectStudent() dengan kondisi berhasil ditemukan, selectStudent() dengan kondisi gagal ditemukan, dan selectAllStudents() menggunakan Apache JMeter. Gunakan Number of Threads sama dengan 100 dan Ramp-up period 10 detik. Anda berarti perlu melakukan 3x test, 1 test untuk setiap kemungkinan hasil. Screenshot window JMeter Anda.
 - jmeter selectStudentBerhasil

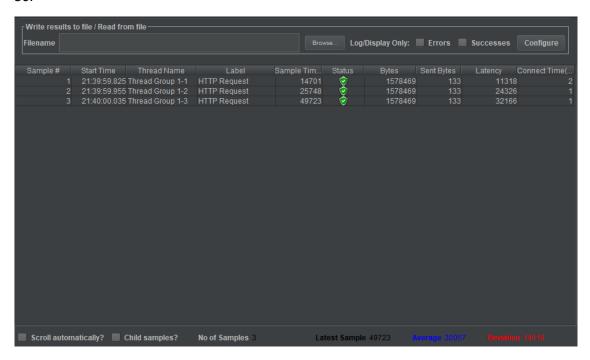


- jmeter_selectStudentGagal

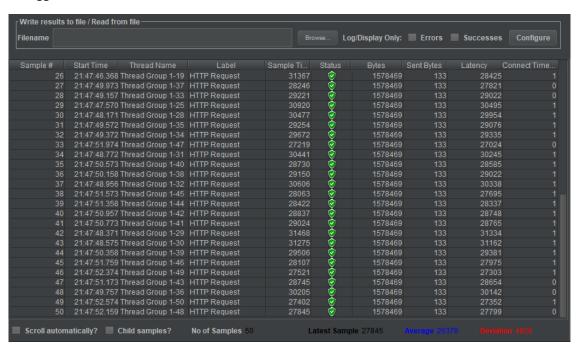


jmeter selectAllStudents

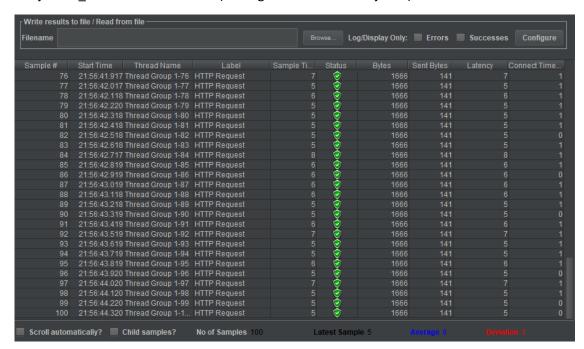
Menggunakan Number of Threads 100, proses berjalan sangat lama jadi saya turunkan menjadi 50.



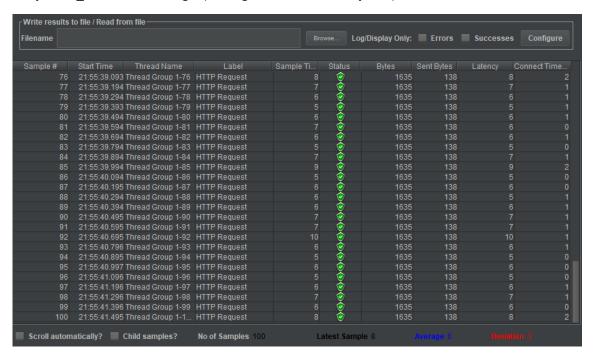
Menggunakan Number of Threads 50



- b. Lakukan load testing kembali untuk semua kemungkinan hasil dengan kode dan database yang sudah Anda optimasi. Gunakan Number of Threads sama dengan 100 dan Ramp-up period 10 detik. Screenshot kembali window JMeter Anda. (menggunakan Optimize Table pada database terlihat perbandingan Average nya lebih kecil)
 - jmeter selectStudentBerhasil (Average Sebelum 13 menjadi 6)



jmeter_selectStudentGagal (Average Sebelum 10 menjadi 6)



jmeter_selectAllStudents (Average Sebelum 26378 menjadi 21738)

