

Set Up SQL Scripts in properties file



SQL for Sample Data

- Currently the SQL for sample data is hard-coded in our tests
- We would like to move the SQL to our properties file

@BeforeEach and @AfterEach

StudentAndGradeServiceTest.java

```
import org.springframework.jdbc.core.JdbcTemplate;
import org.junit.jupiter.api.AfterEach;
import org.junit.jupiter.api.BeforeEach;
...
@TestPropertySource("/application.properties")
@SpringBootTest
public class StudentAndGradeServiceTest {

    @Autowired
    private JdbcTemplate jdbc;

    @BeforeEach
    public void setupDatabase() {
        jdbc.execute("insert into student(id, firstname, lastname, email_address) " +
                    "values (1, 'Eric', 'Roby', 'eric.roby@luv2code_school.com')");
        ...
    }

    @AfterEach
    public void setupAfterTransaction() {
        jdbc.execute("DELETE FROM student");
        ...
    }
}
```

This is what we currently have

Insert sample data

Delete sample data

SQL is hard coded

Development Process

Step-By-Step

1. Add SQL to application.properties
2. Inject SQL into test using @Value
3. Refactor @BeforeEach and @AfterEach

Step 1: Add SQL to application.properties

application.properties

```
...  
  
sql.script.create.student=insert into student(id,firstname,lastname,email_address) \  
  values (1,'Eric', 'Roby', 'eric.roby@luv2code_school.com')  
  
sql.script.delete.student=DELETE FROM student  
  
...
```

Step 2: Inject SQL into test using @Value

StudentAndGradeServiceTest.java

```
...
@TestPropertySource("/application.properties")
@SpringBootTest
public class StudentAndGradeServiceTest {

    @Autowired
    private JdbcTemplate jdbc;

    @Value("${sql.script.create.student}")
    private String sqlAddStudent;

    @Value("${sql.script.delete.student}")
    private String sqlDeleteStudent;

    ...
}
```

application.properties

```
...
sql.script.create.student=insert into student(id,firstname,lastname,email_address) \
    values (1,'Eric', 'Roby', 'eric.roby@luv2code_school.com')

sql.script.delete.student=DELETE FROM student
...
```

Step 3: Refactor @BeforeEach and @AfterEach

StudentAndGradeServiceTest.java

```
...
@TestPropertySource("/application.properties")
@SpringBootTest
public class StudentAndGradeServiceTest {

    @Autowired
    private JdbcTemplate jdbc;

    @Value("${sql.script.create.student}")
    private String sqlAddStudent;

    @Value("${sql.script.delete.student}")
    private String sqlDeleteStudent;

    @BeforeEach
    public void setupDatabase() {
        jdbc.execute(sqlAddStudent);
    }

    @AfterEach
    public void setupAfterTransaction() {
        jdbc.execute(sqlDeleteStudent);
    }
}
```

Refactored code

Refactored code