

Spring Jdbc @DataJdbcTest

Using product repository

```
package integration.tests.databasesjdbc;
```

```
import java.util.List;
```

```
import org.assertj.core.api.Assertions;
```

```
import org.junit.jupiter.api.AfterEach;
```

```
import org.junit.jupiter.api.Test;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.boot.test.autoconfigure.data.jdbc.DataJdbcTest;
```

```
import org.springframework.test.context.jdbc.Sql;
```

```
import integration.tests.databasesjdbc.product.Product;
```

```
import integration.tests.databasesjdbc.product.ProductRepository;
```

```
@DataJdbcTest
```

```
class JdbcIntegrationDataJdbcTest {
```

```
    @Autowired
```

```
    private ProductRepository productRepository;
```

```
    @AfterEach
```

```
    public void cleanUp() {
```

```
        productRepository.deleteAll();
```

```
    }
```

```
    @Test
```

```
    @Sql(statements = "insert into product (id, name, price)
```

```
values(1, 'Mouse', 15)")
```

```
    public void findByNameIgnoreCaseReturnProduct() {
```

```
        Product product =
```

```
productRepository.findByNameIgnoreCase("mouse");
```

```
        Assertions.assertThat(product.getName()).isEqualTo("Mouse");
```

```
        Assertions.assertThat(product.getPrice()).isEqualTo(15);
```

```
    }
```

```
    @Test
```

```
    @Sql(statements = "insert into product (id, name, price)
```

```
values(1, 'Scanner', 40)")
```

```
    @Sql(statements = "insert into product (id, name, price)
```

```
values(2, 'Mouse', 15)")
```

```
    public void returnProductsSortByPriceAscendent() {
```

```
        List<Product> products =
```

```
productRepository.findAllOrderedByPriceAsc();
```

```

    Assertions.assertThat(products).extracting(Product::getName).containsExactly
        ("Mouse", "Scanner");
    }

}

```

Using jdbc template

```

package integration.tests.databasesjdbc;

import org.assertj.core.api.Assertions;
import org.junit.jupiter.api.BeforeEach;
import org.junit.jupiter.api.Test;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.autoconfigure.data.jdbc.DataJdbcTest;
import org.springframework.jdbc.core.JdbcTemplate;
import org.springframework.test.context.jdbc.Sql;

import integration.tests.databasesjdbc.product.Product;
import integration.tests.databasesjdbc.product.ProductDAO;
import integration.tests.databasesjdbc.product.ProductRowMapper;

@DataJdbcTest
class JdbTemplateIntegrationDataJdbcTest {

    @Autowired
    private JdbcTemplate jdbcTemplate;

    private ProductDAO productDAO;

    @BeforeEach
    public void setup() {
        this.productDAO = new ProductDAO(jdbcTemplate);
    }

    @Test
    @Sql(statements = "insert into product (id, name, price)
values(1,'Mouse',15)")
    public void productShouldBeRemovedWhenDeleteIsCalled() {
        productDAO.deleteById(1);

        Integer countAfterDelete = jdbcTemplate.queryForObject("SELECT
COUNT(*) FROM product", Integer.class);
        Assertions.assertThat(countAfterDelete).isEqualTo(0);
    }
}

```

```

@Test
public void productShouldBeInsertedWhenSaveIsCalled() {
    Product product = productDAO.save(new Product("Webcam",5));

    Product insertedProduct = jdbcTemplate.queryForObject("SELECT id,
name, price FROM product WHERE id = ?",
        new ProductRowMapper(), product.getId());
    Assertions.assertThat(insertedProduct).isEqualTo(product);
}

@Test
@Sql(statements = "insert into product (id, name,price)
values(1,'Mouse',15)")
public void findByIdReturnProduct() {
    Product product = productDAO.findById(1);

    Assertions.assertThat(product.getName()).isEqualTo("Mouse");
    Assertions.assertThat(product.getPrice()).isEqualTo(15);
}
}

```