

ACTIVITY PERTEMUAN 6

NAMA : Fizri Rosdiansyah

NPM : 50421535

KELAS : 4IA06

MATERI : Implementasi AOP dan dependency injection pada project Spring dan Hibernate

MATA PRAKTIKUM : Rekayasa Perangkat Lunak 2

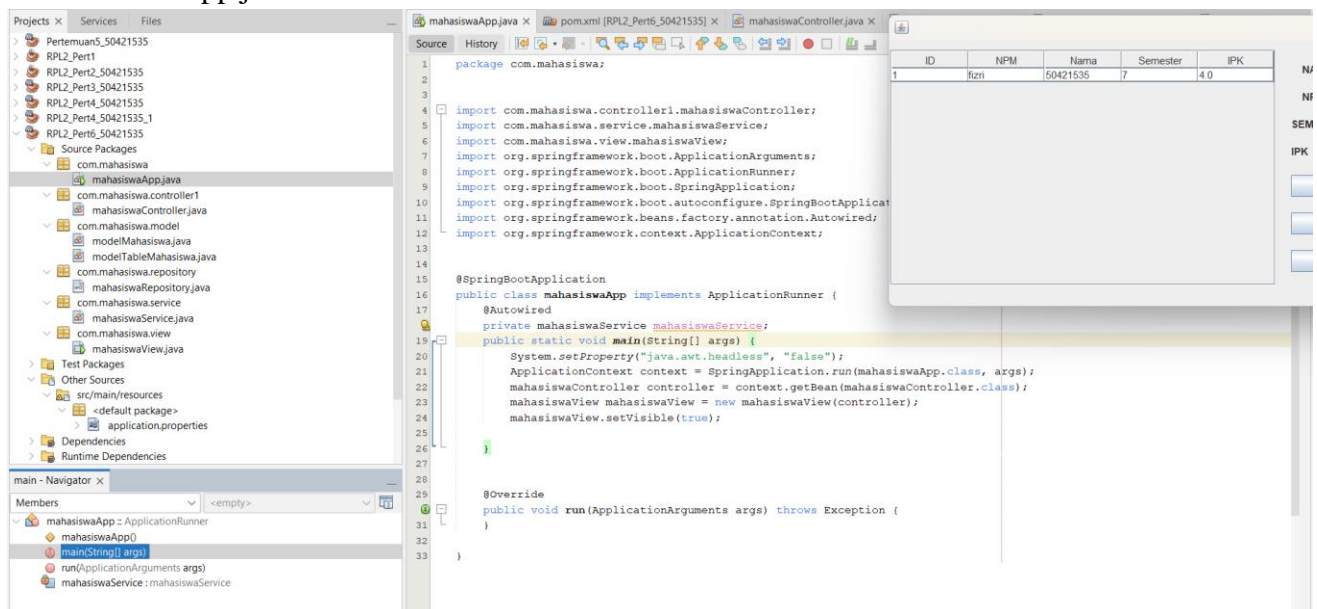
Soal

1. Screenshot hasil codingan kalian lengkap dengan window Projects, Navigator, dan Outputnya!

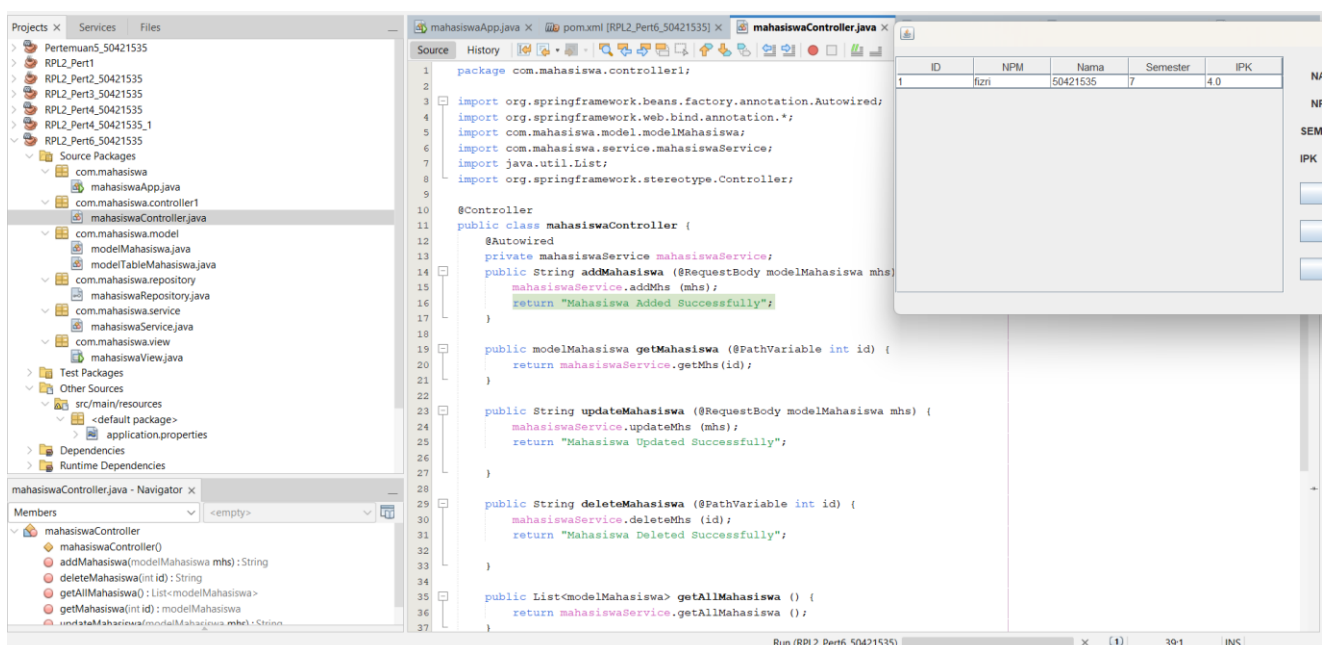
Jawab

1

A.mahasiswaApp.java



B.mahasiswaController



C.modelMahasiswa

The image displays a Java IDE environment with the following components:

- Project Explorer (Left):** Shows the project structure. The 'modelMahasiswa' package is selected, containing files like 'modelMahasiswa.java', 'modelTableMahasiswa.java', 'mahasiswaRepository.java', 'mahasiswaService.java', and 'mahasiswaView.java'.
- Source Editor (Middle):** Displays the code for 'modelMahasiswa.java'. The code includes package imports, JPA annotations, and getter/setter methods.

```
1 package com.mahasiswa.model;
2 import jakarta.persistence.*;
3
4 @Entity
5 @Table(name = "mahasiswa")
6
7 public class modelMahasiswa {
8     @Id
9     @GeneratedValue(strategy = GenerationType.IDENTITY)
10    @Column(name = "id")
11    private int id;
12    @Column(name = "npm", nullable = false, length = 8)
13    private String npm;
14    @Column(name = "nama", nullable = false, length = 50)
15    private String nama;
16    @Column(name = "semester")
17    private int semester;
18    @Column(name = "ipk")
19    private float ipk;
20
21    public modelMahasiswa() {
22    }
23
24    public modelMahasiswa(int id, String npm, String nama, int semester, float ipk) {
25        this.id = id;
26        this.npm = npm;
27        this.nama = nama;
28        this.semester = semester;
29        this.ipk = ipk;
30    }
31
32    public int getId() {
33        return id;
34    }
35
36    public void setId(int id) {
37
```
- Table View (Right):** Shows a table with the following data:

ID	NPM	Nama	Semester	IPK
1	fizr	50421535	7	4.0

D.modelTableMahasiswa

The screenshot displays the implementation of the `modelTableMahasiswa` class in an IDE. The code is organized into several methods that implement the `AbstractTableModel` interface.

```
package com.mahasiswa.model;

import javax.swing.table.AbstractTableModel;
import java.util.List;

public class modelTableMahasiswa extends AbstractTableModel {
    private List<modelMahasiswa> mahasiswaList;
    private String[] columnNames = {"ID", "NPM", "Nama", "Semester", "IPK"};

    public modelTableMahasiswa(List<modelMahasiswa> mahasiswaList) {
        this.mahasiswaList = mahasiswaList;
    }

    @Override
    public int getRowCount() {
        return mahasiswaList.size();
    }

    @Override
    public int getColumnCount() {
        return columnNames.length;
    }

    @Override
    public Object getValueAt(int rowIndex, int columnIndex) {
        modelMahasiswa mahasiswa = mahasiswaList.get(rowIndex);
        switch (columnIndex) {
            case 0:
                return mahasiswa.getId();
            case 1:
                return mahasiswa.getNpm();
            case 2:
                return mahasiswa.getNama();
            case 3:
                return mahasiswa.getSemester();
            case 4:
                return mahasiswa.getIpk();
        }
    }

    @Override
    public String getColumnName(int column) {
        return columnNames[column];
    }

    @Override
    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return false;
    }

    public void setMahasiswaList(List<modelMahasiswa> mahasiswaList) {
        this.mahasiswaList = mahasiswaList;
        fireTableDataChanged();
    }
}
```

The table view on the right shows the data for a student named Ifzri:

ID	NPM	Nama	Semester	IPK
1	Ifzri	50421535	7	4.0

E.mahasiswaRepository

The screenshot shows an IDE with the following components:

- Project Explorer:** Displays a project structure with packages like `com.mahasiswa`, `com.mahasiswa.model`, `com.mahasiswa.repository`, `com.mahasiswa.service`, `com.mahasiswa.view`, `Test Packages`, `Other Sources`, `Dependencies`, and `Runtime Dependencies`.
- Source Editor:** Shows the `modelMahasiswa.java` file with the following code:

```
1 package com.mahasiswa.repository;
2
3
4 import com.mahasiswa.model.modelMahasiswa;
5 import org.springframework.data.jpa.repository.JpaRepository;
6 import org.springframework.stereotype.Repository;
7
8
9 @Repository
10 public interface mahasiswaRepository extends JpaRepository<modelMahasiswa, Integer> {
11
12
13 }
```
- Navigator:** Shows the `mahasiswaRepository.java` file with the following code:

```
mahasiswaRepository : none : JpaRepository<modelMahasiswa, Integer>
```
- Modal Form:** A dialog box with a table and input fields. The table has columns: ID, NPM, Nama, Semester, and IPK. The input fields are labeled NAMA, NPM, SEMESTER, and IPK. There are buttons for Save, Refresh, and Delete.

F.mahasiswaService

The screenshot shows an IDE with the following components:

- Project Explorer:** Displays a project structure with packages like `com.mahasiswa`, `com.mahasiswa.model`, `com.mahasiswa.repository`, `com.mahasiswa.service`, `com.mahasiswa.view`, `Test Packages`, `Other Sources`, `Dependencies`, and `Runtime Dependencies`.
- Source Editor:** Shows the `mahasiswaService.java` file with the following code:

```
1 package com.mahasiswa.service;
2
3 import com.mahasiswa.model.modelMahasiswa;
4 import com.mahasiswa.repository.mahasiswaRepository;
5 import java.util.List;
6 import org.springframework.beans.factory.annotation.Autowired;
7 import org.springframework.stereotype.Service;
8
9 @Service
10 public class mahasiswaService {
11     @Autowired
12     private mahasiswaRepository repository;
13
14     public void addMhs(modelMahasiswa mhs) {
15         repository.save(mhs);
16     }
17
18     public modelMahasiswa getMhs(int id) {
19         return repository.findById(id).orElse(null);
20     }
21
22     public void updateMhs(modelMahasiswa mhs) {
23         repository.save(mhs);
24     }
25
26     public void deleteMhs(int id) {
27         repository.deleteById(id);
28     }
29
30     public List<modelMahasiswa> getAllMahasiswa() {
31         return repository.findAll();
32     }
33 }
```
- Navigator:** Shows the `mahasiswaService.java` file with the following code:

```
mahasiswaService : none : JpaRepository<modelMahasiswa, Integer>
```
- Modal Form:** A dialog box with a table and input fields. The table has columns: ID, NPM, Nama, Semester, and IPK. The input fields are labeled NAMA, NPM, SEMESTER, and IPK. There are buttons for Save, Refresh, and Delete.

G.mahasiswaView

The screenshot displays the development environment for a Java Swing application named G.mahasiswaView. The interface is divided into several panels:

- Project Explorer:** Shows the project structure, including source packages (com.mahasiswa), test packages, and other sources. The main package is com.mahasiswa, which contains files like mahasiswaApp.java, mahasiswaController1.java, modelMahasiswa.java, modelTableMahasiswa.java, mahasiswaRepository.java, mahasiswaService.java, and mahasiswaView.java.
- Source Editor:** Displays the Java code for the mahasiswaView class. The code includes imports for various Java Swing and utility classes, and defines the mahasiswaView class extending JFrame. It includes methods for initializing components and loading data from a table model.
- Design View:** Shows a visual representation of the user interface. It features a window titled "Title 1" through "Title 4". On the right side, there are input fields for "NAMA", "NPM", "SEMESTER", and "IPK", each with a corresponding "Save", "Refresh", or "Delete" button.
- Palette:** Lists various Swing components and containers available for use in the design view, such as Panel, Split Pane, Tool Bar, Internal Frame, Label, Toggle Button, Radio Button, Combo Box, Text Field, Progress Bar, Password Field, Separator, Editor Pane, Table, Button, Check Box, Button Group, List, Text Area, Slider, Formatted Field, Spinner, Text Pane, and Tree.

The Source Editor shows the following code:

```
package com.mahasiswa.view;

import com.mahasiswa.controller1.mahasiswaController;
import com.mahasiswa.model.modelMahasiswa;
import com.mahasiswa.model.modelTableMahasiswa;

import java.util.List;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class mahasiswaView extends javax.swing.JFrame {
    private mahasiswaController controller;

    public mahasiswaView(mahasiswaController controller) {
        this.controller = controller;
        initComponents();
        loadMahasiswaTable();
    }

    private mahasiswaView() {
        throw new UnsupportedOperationException("Not Supported Yet.");
    }

    public void loadMahasiswaTable() {
        List<modelMahasiswa> listMahasiswa = controller.getAllMahasiswa();
        modelTableMahasiswa tableModel = new modelTableMahasiswa(listMahasiswa);
        dataTable.setModel(tableModel);
    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
}
```

ID	NPM	Nama	Semester	IPK
1	fizri	50421535	7	4.0

H.pom.xml

The screenshot displays an IDE with the following components:

- Project Explorer:** Shows the project structure for RPL2_Pert6_50421535, including source packages (com.mahasiswa), test packages, and other sources.
- Navigator:** Shows the POM model for the project, including the Model Version (4.0.0), GroupId (com.mahasiswa), ArtifactId (RPL2_Pert6_50421535), Packaging (jar), and Name (spring-boot-starter-parent).
- Source Editor:** Displays the pom.xml file with the following content:

```
<parent>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-parent</artifactId>
  <version>3.3.3</version>
  <relativePath/>
</parent>

<dependencies>
  <!-- Hibernate + Spring Data JPA -->
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-data-jpa</artifactId>
  </dependency>

  <!-- MySQL Connector -->
  <dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.33</version>
  </dependency>

  <!-- Spring Boot Web dependency (for MVC if needed) -->
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>

  <!-- Testing dependencies -->
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-test</artifactId>
    <scope>test</scope>
  </dependency>
</dependencies>

<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
</project>
```
- Table:** A table with 5 columns: ID, NPM, Nama, Semester, and IPK. It contains one row of data: 1, fuzri, 50421535, 7, 4.0.

I. application.properties

The screenshot shows the NetBeans IDE with the `application.properties` file open. The configuration includes MySQL and Hibernate settings. A modal form for student data is also displayed.

application.properties

```
#Konfigurasi MySQL, Hibernate
spring.datasource.url=jdbc:mysql://localhost:3306/peraturan6_db?useSSL=false&serverTimezone=UTC
spring.datasource.username=root
spring.datasource.password=
spring.datasource.driver-class-name=com.mysql.jdbc.Driver

#Hibernate settings
spring.jpa.hibernate.ddl-auto=update
spring.jpa.show-sql=true
```

Student Data Form

ID	NPM	Nama	Semester	IPK
1	fzri	50421535	7	4.0

Form fields: NAMA, NPM, SEMESTER, IPK. Buttons: Save, Refresh, Delete.

J. Output

The screenshot shows the NetBeans IDE with the application output window open. The output displays the startup logs of the application, including database initialization and Hibernate settings. A modal form for student data is also displayed.

Application Output

```
2024-11-19T11:19:33.971+07:00 INFO 1720 --- [main] .s.d.r. Hapus Mahasiswa
2024-11-19T11:19:34.039+07:00 INFO 1720 --- [main] .s.d.r.
2024-11-19T11:19:34.630+07:00 INFO 1720 --- [main] o.s.b.
2024-11-19T11:19:34.644+07:00 INFO 1720 --- [main] o.apac
2024-11-19T11:19:34.644+07:00 INFO 1720 --- [main] o.apac
2024-11-19T11:19:34.697+07:00 INFO 1720 --- [main] o.a.c.
2024-11-19T11:19:34.698+07:00 INFO 1720 --- [main] w.s.c.S
2024-11-19T11:19:34.871+07:00 INFO 1720 --- [main] o.hibernate.jpa.internal.util.LogHelper : HHH000204: Processing Persiste
2024-11-19T11:19:34.939+07:00 INFO 1720 --- [main] org.hibernate.Version : HHH000412: Hibernate ORM core
2024-11-19T11:19:34.976+07:00 INFO 1720 --- [main] o.h.c.internal.RegionFactoryInitiator : HHH000026: Second-level cache
2024-11-19T11:19:35.262+07:00 INFO 1720 --- [main] o.s.o.j.p.SpringPersistenceUnitInfo : No LoadTimeWeaver setup; ignor
2024-11-19T11:19:35.289+07:00 INFO 1720 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Starting...
2024-11-19T11:19:35.289+07:00 WARN 1720 --- [main] com.zaxxer.hikari.util.DriverDataSource : Registered driver with driverC
2024-11-19T11:19:35.440+07:00 INFO 1720 --- [main] com.zaxxer.hikari.pool.HikariPool : HikariPool-1 - Added connectio
2024-11-19T11:19:35.442+07:00 INFO 1720 --- [main] com.zaxxer.hikari.HikariDataSource : HikariPool-1 - Start completed
2024-11-19T11:19:35.499+07:00 INFO 1720 --- [main] org.hibernate.dialect.Dialect : HHH000511: The 5.5.5 version f
2024-11-19T11:19:36.189+07:00 INFO 1720 --- [main] o.h.e.t.j.p.i.JtaPlatformInitiator : HHH000489: No JTA platform ava
2024-11-19T11:19:36.229+07:00 INFO 1720 --- [main] j.LocalContainerEntityManagerFactoryBean : Initialized JPA EntityManagerF
2024-11-19T11:19:36.486+07:00 WARN 1720 --- [main] JpaBaseConfiguration$JpaWebConfiguration : spring.jpa.open-in-view is enal
2024-11-19T11:19:36.826+07:00 INFO 1720 --- [main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8080 (h
2024-11-19T11:19:36.835+07:00 INFO 1720 --- [main] com.mahasiswa.mahasiswaApp : Started mahasiswaApp in 4.015
Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0
```

Student Data Form

ID	NPM	Nama	Semester	IPK
1	fzri	50421535	7	4.0

Form fields: NAMA, NPM, SEMESTER, IPK. Buttons: Save, Refresh, Delete.

ID	NPM	Nama	Semester	IPK
2	fizri	50421535	7	3.6

NAMA

NPM

SEMESTER

IPK

Save

Refresh

Delete

RPL2_Pert6_50421535

delTableMahasiswa.java

ate

c:mysql://localhost:330

e=root

d=

class-name=com.mysql.jc

auto=update

2024-11-19T11:19:34.939+07:00

INFO 1720 --- [

main] org.hibernate.Version

:

2024-11-19T11:19:34.976+07:00

INFO 1720 --- [

main] o.h.c.internal.RegionFactoryInitiator

:

2024-11-19T11:19:35.262+07:00

INFO 1720 --- [

main] o.s.o.j.p.SpringPersistenceUnitInfo

:

2024-11-19T11:19:35.285+07:00

INFO 1720 --- [

main] com.zaxxer.hikari.HikariDataSource

:

2024-11-19T11:19:35.289+07:00

WARN 1720 --- [

main] c.zaxxer.hikari.util.DriverDataSource

:

2024-11-19T11:19:35.440+07:00

INFO 1720 --- [

main] com.zaxxer.hikari.pool.HikariPool

:

2024-11-19T11:19:35.442+07:00

INFO 1720 --- [

main] com.zaxxer.hikari.HikariDataSource

:

2024-11-19T11:19:35.499+07:00

WARN 1720 --- [

main] org.hibernate.dialect.Dialect

:

2024-11-19T11:19:36.189+07:00

INFO 1720 --- [

main] o.h.e.t.j.p.i.JtaPlatformInitiator

:

2024-11-19T11:19:36.229+07:00

INFO 1720 --- [

main] j.LocalContainerEntityManagerFactoryBean

:

2024-11-19T11:19:36.486+07:00

WARN 1720 --- [

main] JpaBaseConfiguration\$JpaWebConfiguration

:

2024-11-19T11:19:36.826+07:00

INFO 1720 --- [

main] o.s.b.w.embedded.tomcat.TomcatWebServer

:

2024-11-19T11:19:36.835+07:00

INFO 1720 --- [

main] com.mahasiswa.mahasiswaApp

:

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0 where mm1_0.

Hibernate: delete from mahasiswa where id=?

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

3.6

50421535

7

fizri

Hibernate: insert into mahasiswa (ipk,nama,npm,semester) values (?, ?, ?, ?)

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

ID	NPM	Nama	Semester	IPK
2	fizri	50421535	7	3.6

NAMA

NPM

SEMESTER

IPK

Save

Refresh

Delete

RPL2_Pert

delTableMahasiswa.java

ate

c:mysql://local

e=root

d=

class-name=com.

auto=update

2024-11-19T11:19:35.289+07:00

WARN 1720 --- [

main] c.zaxxer.hikari.util.DriverDataSou

:

2024-11-19T11:19:35.440+07:00

INFO 1720 --- [

main] com.zaxxer.hikari.pool.HikariPool

:

2024-11-19T11:19:35.442+07:00

INFO 1720 --- [

main] com.zaxxer.hikari.HikariDataSource

:

2024-11-19T11:19:35.499+07:00

WARN 1720 --- [

main] org.hibernate.dialect.Dialect

:

2024-11-19T11:19:36.189+07:00

INFO 1720 --- [

main] o.h.e.t.j.p.i.JtaPlatformInitiator

:

2024-11-19T11:19:36.229+07:00

INFO 1720 --- [

main] j.LocalContainerEntityManagerFactor

:

2024-11-19T11:19:36.486+07:00

WARN 1720 --- [

main] JpaBaseConfiguration\$JpaWebConfigur

:

2024-11-19T11:19:36.826+07:00

INFO 1720 --- [

main] o.s.b.w.embedded.tomcat.TomcatWebSe

:

2024-11-19T11:19:36.835+07:00

INFO 1720 --- [

main] com.mahasiswa.mahasiswaApp

:

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0 wher

Hibernate: delete from mahasiswa where id=?

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

3.6

50421535

7

fizri

Hibernate: insert into mahasiswa (ipk,nama,npm,semester) values (?, ?, ?, ?)

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0

Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0