

## ACTIVITY PERTEMUAN 6

**NAMA** : Muhammad Rifqi Dwi Putra

**NPM** : 51422145

**KELAS** : 4IA08

**MATERI** : Implementasi AOP dan Dependency Injection Pada Project Spring dan Hibernate

**MATA PRAKTIKUM** : Rekayasa Perangkat Lunak 2

(Screenshoot langkah-langkah sesuai video pembelajaran dan jelaskan dengan ringkas)

### 1. Screenshot Kode dan Output Program!

Kode :

Pom.xml

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
3   <modelVersion>4.0.0</modelVersion>
4   <groupId>com.mycompany</groupId>
5   <artifactId>rifqi_pert6</artifactId>
6   <version>1.0-SNAPSHOT</version>
7   <packaging>jar</packaging>
8   <properties>
9     <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
10    <maven.compiler.release>22</maven.compiler.release>
11    <exec.mainClass>com.mycompany.rifqi_pert6.MahasiswaApp</exec.mainClass>
12  </properties>
13  <parent>
14    <groupId>org.springframework.boot</groupId>
15    <artifactId>spring-boot-starter-parent</artifactId>
16    <version>3.3.3</version>
17    <relativePath></relativePath>
18  </parent>
19  <dependencies>
20    <!-- Hibernate + Spring Data JPA -->
21    <dependency>
22      <groupId>org.springframework.boot</groupId>
23      <artifactId>spring-boot-starter-data-jpa</artifactId>
24    </dependency>
25
26    <!-- MySQL Connector -->
27    <dependency>
28      <groupId>mysql</groupId>
29      <artifactId>mysql-connector-java</artifactId>
30      <version>8.0.33</version>
31    </dependency>
32
33    <!-- Spring Boot Web dependency (for MVC if needed) -->
34    <dependency>
35      <groupId>org.springframework.boot</groupId>
36      <artifactId>spring-boot-starter-web</artifactId>
37    </dependency>
38
39    <!-- Testing dependencies -->
40    <dependency>
41      <groupId>org.springframework.boot</groupId>
42      <artifactId>spring-boot-starter-test</artifactId>
43      <scope>test</scope>
44    </dependency>
45  </dependencies>
46
47  <build>
48    <plugins>
49      <plugin>
50        <groupId>org.springframework.boot</groupId>
51        <artifactId>spring-boot-maven-plugin</artifactId>
52      </plugin>
53    </plugins>
54  </build>
55
56
57 </project>
```

## MahasiswaController.java

```
Output - Run (MahasiswaApp) x MahasiswaApp.java x pom.xml [rifqi_pert6] x MahasiswaController.java x ModelMahasiswa.java x ModelTabelMahasiswa.java x MahasiswaRepository.java x MahasiswaService.java x Mahasis... < > v
Source History
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package com.mycompany.rifqi_pert6.controller;

import com.mycompany.rifqi_pert6.model.ModelMahasiswa;
import com.mycompany.rifqi_pert6.service.MahasiswaService;
import java.util.List;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestBody;

/**
 *
 * @author mrfiq
 */
@Controller
public class MahasiswaController {
    @Autowired
    private MahasiswaService mahasiswaService;

    public String addMahasiswa(@RequestBody ModelMahasiswa mhs) {
        mahasiswaService.addMhs(mhs);
        return "Mahasiswa added successfully";
    }

    public ModelMahasiswa getMahasiswa(@PathVariable int id) {
        return mahasiswaService.getMhs(id);
    }

    public String updateMahasiswa(@RequestBody ModelMahasiswa mhs) {
        mahasiswaService.updateMhs(mhs);
        return "Mahasiswa updated successfully";
    }

    public String deleteMahasiswa(@PathVariable int id) {
        mahasiswaService.deleteMhs(id);
        return "Mahasiswa deleted successfully";
    }

    public List<ModelMahasiswa> getAllMahasiswa() {
        return mahasiswaService.getAllMahasiswa();
    }
}
```

## ModelMahasiswa.java

```
Output - Run (MahasiswaApp) x MahasiswaApp.java x pom.xml [rifqi_pert6] x MahasiswaController.java x ModelMahasiswa.java x ModelTabelMahasiswa.java x MahasiswaRepository.java x MahasiswaService.java x Mahasis... < > v
Source History
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package com.mycompany.rifqi_pert6.model;

import jakarta.persistence.*;

/**
 *
 * @author mrfiq
 */
@Entity
@Table(name="mahasiswa")
public class ModelMahasiswa {
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Column(name="id")
    private int id;

    @Column(name="npm", nullable=false, length=10)
    private String npm;

    @Column(name="nama", nullable=false, length=50)
    private String nama;

    @Column(name="semester", nullable=false)
    private int semester;

    @Column(name="ipk", nullable=false)
    private float ipk;

    public ModelMahasiswa() {
    }

    public ModelMahasiswa(int id, String npm, String nama, int semester, float ipk) {
        this.id = id;
        this.npm = npm;
        this.nama = nama;
        this.semester = semester;
        this.ipk = ipk;
    }

    public int getId() {
        return id;
    }
}
```

## ModelTabelMahasiswa.java

```
Source History Output - Run (MahasiswaApp) MahasiswaApp.java pom.xml [rifqi.pert6] MahasiswaController.java ModelMahasiswa.java ModelTabelMahasiswa.java MahasiswaRepository.java MahasiswaService.java Mahasiswa...

1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
4  */
5  package com.mycompany.rifqi_pert6.model;
6
7  import java.util.List;
8  import javax.swing.table.AbstractTableModel;
9
10
11  /**
12   *
13   * @author mrifqi
14   */
15  public class ModelTabelMahasiswa extends AbstractTableModel {
16      private List<ModelMahasiswa> mahasiswaList;
17      private String[] columnNames = {"ID", "NPM", "Nama", "Semester", "IPK"};
18
19      public ModelTabelMahasiswa(List<ModelMahasiswa> mahasiswaList) {
20          this.mahasiswaList = mahasiswaList;
21      }
22
23      @Override
24      public int getRowCount() {
25          return mahasiswaList.size();
26      }
27
28      @Override
29      public int getColumnCount() {
30          return columnNames.length;
31      }
32
33      @Override
34      public Object getValueAt(int rowIndex, int columnIndex) {
35          ModelMahasiswa mahasiswa = mahasiswaList.get(rowIndex);
36          switch (columnIndex) {
37              case 0:
38                  return mahasiswa.getId();
39              case 1:
40                  return mahasiswa.getNpm();
41              case 2:
42                  return mahasiswa.getName();
43              case 3:
44                  return mahasiswa.getSemester();
45              case 4:
46                  return mahasiswa.getIpk();
47              default:
48                  return null;
49          }
50      }
51
52      @Override
53      public String getColumnName(int column) {
54          return columnNames[column]; // Mengatur nama kolom
55      }
56
57      @Override
58      public boolean isCellEditable(int rowIndex, int columnIndex) {
59          return false; // Semua sel tidak dapat diedit
60      }
61
62      // Method untuk menambahkan atau memodifikasi data, jika dibutuhkan
63      public void setMahasiswaList(List<ModelMahasiswa> mahasiswaList) {
64          this.mahasiswaList = mahasiswaList;
65          fireTableDataChanged(); // Memberitahu JTable bahwa data telah berubah
66      }
67  }
```

## MahasiswaRepository.java

```

1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Interface.java to edit this template
4   */
5   package com.mycompany.rifqi_pert6.repository;
6
7   import com.mycompany.rifqi_pert6.model.ModelMahasiswa;
8   import org.springframework.data.jpa.repository.JpaRepository;
9   import org.springframework.stereotype.Repository;
10
11  /**
12   *
13   * @author mrfiq
14   */
15
16  @Repository
17  public interface MahasiswaRepository extends JpaRepository<ModelMahasiswa, Integer> {
18
19  }
20

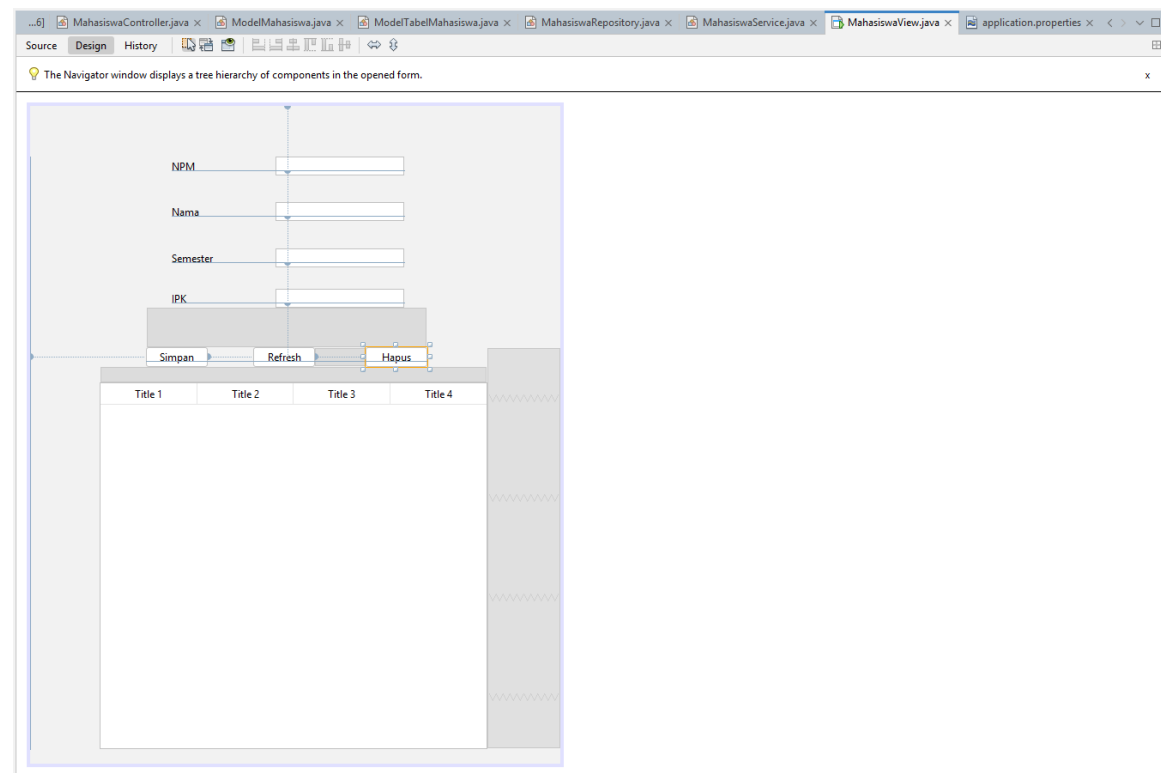
```

## MahasiswaService.java

```
Source History
1  /**
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
4   */
5   package com.mycompany.rifqi_pert6.service;
6
7   import com.mycompany.rifqi_pert6.model.ModelMahasiswa;
8   import com.mycompany.rifqi_pert6.repository.MahasiswaRepository;
9   import java.util.List;
10  import org.springframework.beans.factory.annotation.Autowired;
11  import org.springframework.stereotype.Service;
12
13  /**
14   *
15   * @author mrifqi
16   */
17
18  @Service
19  public class MahasiswaService {
20      @Autowired
21      private MahasiswaRepository repository;
22
23      public void addMhs(ModelMahasiswa mhs) {
24          repository.save(mhs);
25      }
26
27      public ModelMahasiswa getMhs(int id) {
28          return repository.findById(id).orElse(null);
29      }
30
31      public void updateMhs(ModelMahasiswa mhs) {
32          repository.save(mhs);
33      }
34
35      public void deleteMhs(int id) {
36          repository.deleteById(id);
37      }
38
39      public List<ModelMahasiswa> getAllMahasiswa() {
40          return repository.findAll();
41      }
42  }
43
```

## MahasiswaView.java

Design :



## Source

```
Source Design History
1 2
3  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
4  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
5  */
6  package com.mycompany.rifqi_pert6.view;
7
8  import com.mycompany.rifqi_pert6.controller.MahasiswaController;
9  import com.mycompany.rifqi_pert6.model.ModelMahasiswa;
10 import com.mycompany.rifqi_pert6.model.ModelTabelMahasiswa;
11 import java.util.List;
12 import javax.swing.JLabel;
13 import javax.swing.JOptionPane;
14 import javax.swing.JPanel;
15 import javax.swing.JTextField;
16
17 /**
18  *
19  * @author mrfiq
20  */
21 public class MahasiswaView extends javax.swing.JFrame {
22
23     private MahasiswaController controller;
24
25     /**
26      * Creates new form MahasiswaView
27      */
28     public MahasiswaView(MahasiswaController controller) {
29         this.controller = controller;
30         initComponents();
31         loadMahasiswaTable();
32     }
33
34     private MahasiswaView() {
35         throw new UnsupportedOperationException("Not Supported yet.");
36     }
37
38     public void loadMahasiswaTable() {
39         List<ModelMahasiswa> listMahasiswa = controller.getAllMahasiswa();
40         ModelTabelMahasiswa tableModel = new ModelTabelMahasiswa(listMahasiswa);
41         dataTable.setModel(tableModel);
42     }
43 }
```

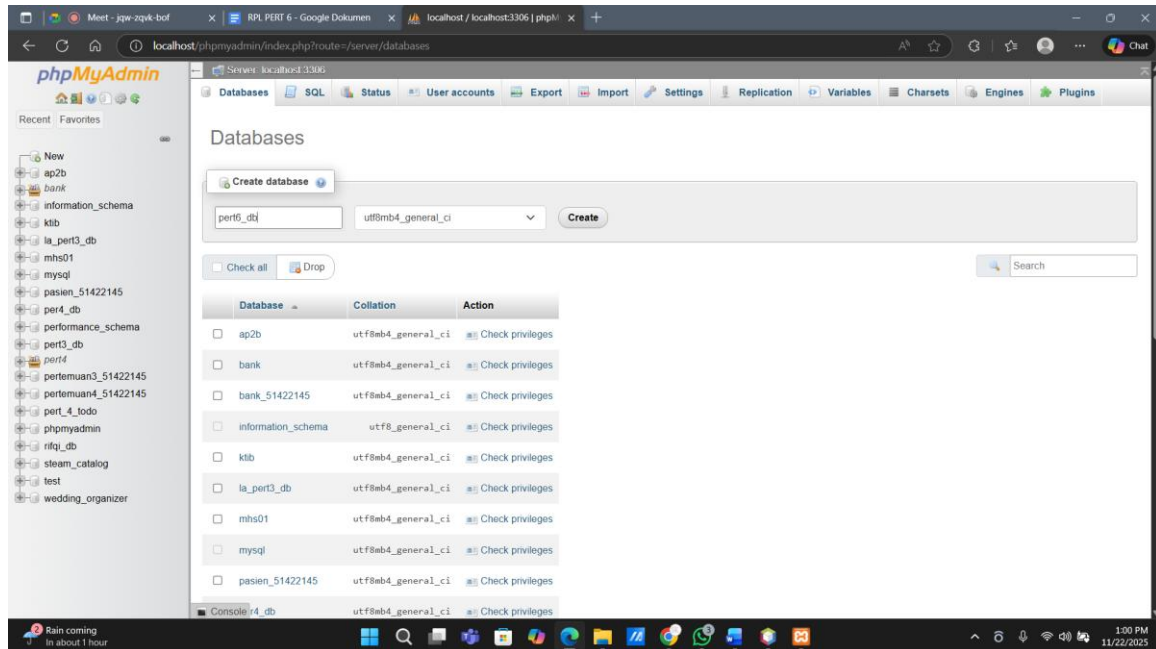
## btnSimpan, btnRefresh, dan btnHapus

```
Source Design History
175 private void btnSimpanActionPerformed(java.awt.event.ActionEvent evt) {
176     // TODO add your handling code here:
177     String npm = npmField.getText();
178     String nama = namaField.getText();
179     int semester = Integer.parseInt(semesterField.getText());
180     float ipk = Float.parseFloat(ipkField.getText());
181
182     ModelMahasiswa mahasiswa = new ModelMahasiswa(0, npm, nama, semester, ipk);
183     System.out.println(mahasiswa.getIpk());
184     System.out.println(mahasiswa.getNama());
185     System.out.println(mahasiswa.getRpm());
186     System.out.println(mahasiswa.getSemester());
187
188     controller.addMahasiswa(mahasiswa);
189     loadMahasiswaTable();
190 }
191
192
193 private void btnRefreshActionPerformed(java.awt.event.ActionEvent evt) {
194     // TODO add your handling code here:
195     loadMahasiswaTable();
196 }
197
198 private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {
199     // TODO add your handling code here:
200     JTextField idField = new JTextField(5);
201
202     // Membuat panel untuk menampung JTextField
203     JPanel panel = new JPanel();
204     panel.add(new JLabel("Masukkan ID yang ingin dihapus:"));
205     panel.add(idField);
206
207     // Menampilkan dialog box dengan JTextField, tombol OK, dan Cancel
208     int result = JOptionPane.showConfirmDialog(null, panel,
209         "Hapus Mahasiswa", JOptionPane.OK_CANCEL_OPTION, JOptionPane.PLAIN_MESSAGE);
210
211     // Jika tombol OK ditekan
212     if (result == JOptionPane.OK_OPTION) {
213         try {
214             // Mengambil input ID dan memanggil metode deleteMhs
215             int id = Integer.parseInt(idField.getText());
216             controller.deleteMahasiswa(id);
217             JOptionPane.showMessageDialog(null, "Data berhasil dihapus.", "Sukses", JOptionPane.INFORMATION_MESSAGE);
218         } catch (NumberFormatException e) {
219             // Menangani error jika ID yang dimasukkan bukan angka
220             JOptionPane.showMessageDialog(null, "ID harus berupa angka.", "Error", JOptionPane.ERROR_MESSAGE);
221         }
222     }
223 }
```

## Application.properties

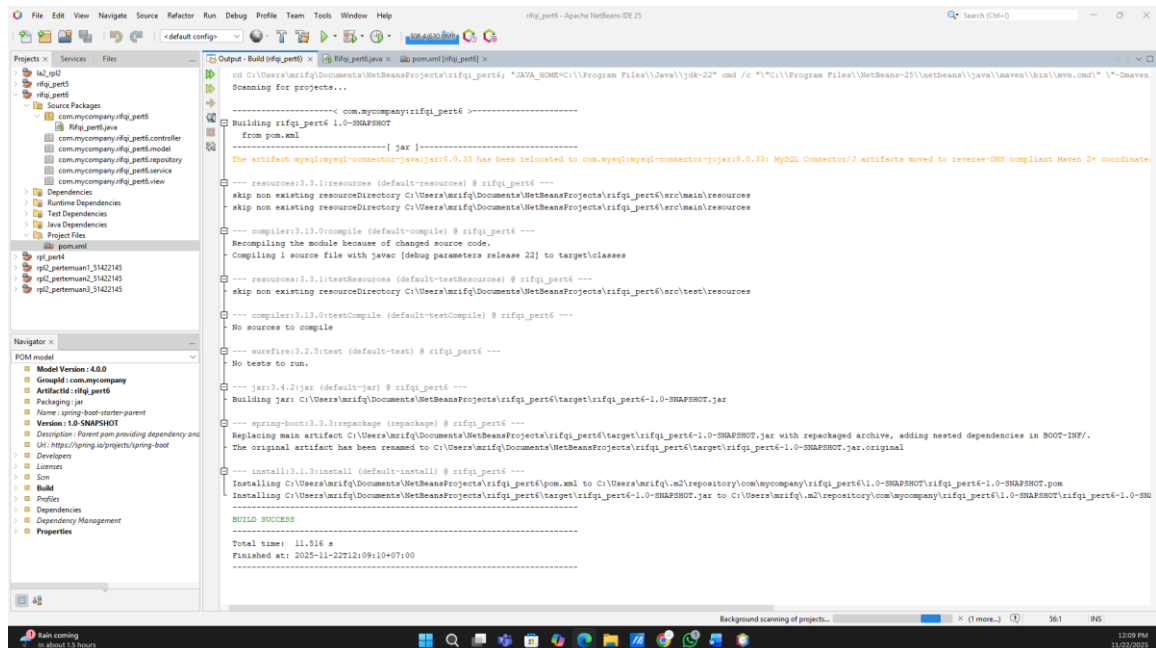
```
Source History
1
2 # Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3 # Click nbfs://nbhost/SystemFileSystem/Templates/Other/properties.properties to edit this template
4
5
6 # Konfigurasi MySQL Hibernate
7 spring.datasource.url=jdbc:mysql://localhost:3306/pert6_db?useSSL=false&serverTimezone=UTC
8 spring.datasource.username=root
9 spring.datasource.password=
10 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
11 server.port=8080
12
13 # Hibernate settings
14 spring.jpa.hibernate.ddl-auto=update
15 spring.jpa.show-sql=true
16
```

## Database



Output :

## Build Pom.xml



Simpan data

NPM

51422145

Nama

Muhammad Rifqi DP

Semester

7

IPK

3.89

Simp...

Refresh

Hapus

ID	NPM	Nama	Semester	IPK
1	51422145	Muhammad R...	7	3.89

Hapus Data

NPM

51422145

Nama

Muhammad Rifqi DP

Semester

7

IPK

3.89

Simp...

Refresh

Hapus

ID	NPM	Nama	Semester	IPK
1	51422145	Muhammad R...	7	3.89

Hapus Mahasiswa

Masukkan ID yang ingin dihapus: 1

OK

Cancel

Refresh Data

NPM

51422145

Nama

Muhammad Rifqi DP

Semester

7

IPK

3.89

Simp...

Refresh

Hapus

ID	NPM	Nama	Semester	IPK
----	-----	------	----------	-----