

ACTIVITY PERTEMUAN 6

NAMA : Muhammad Rifqi Dwi Putra
NPM : 51422145
KELAS : 4IA08
MATERI : Implementasi AOP dan Dependecy Injection Pada Project Spring dan Hibernate

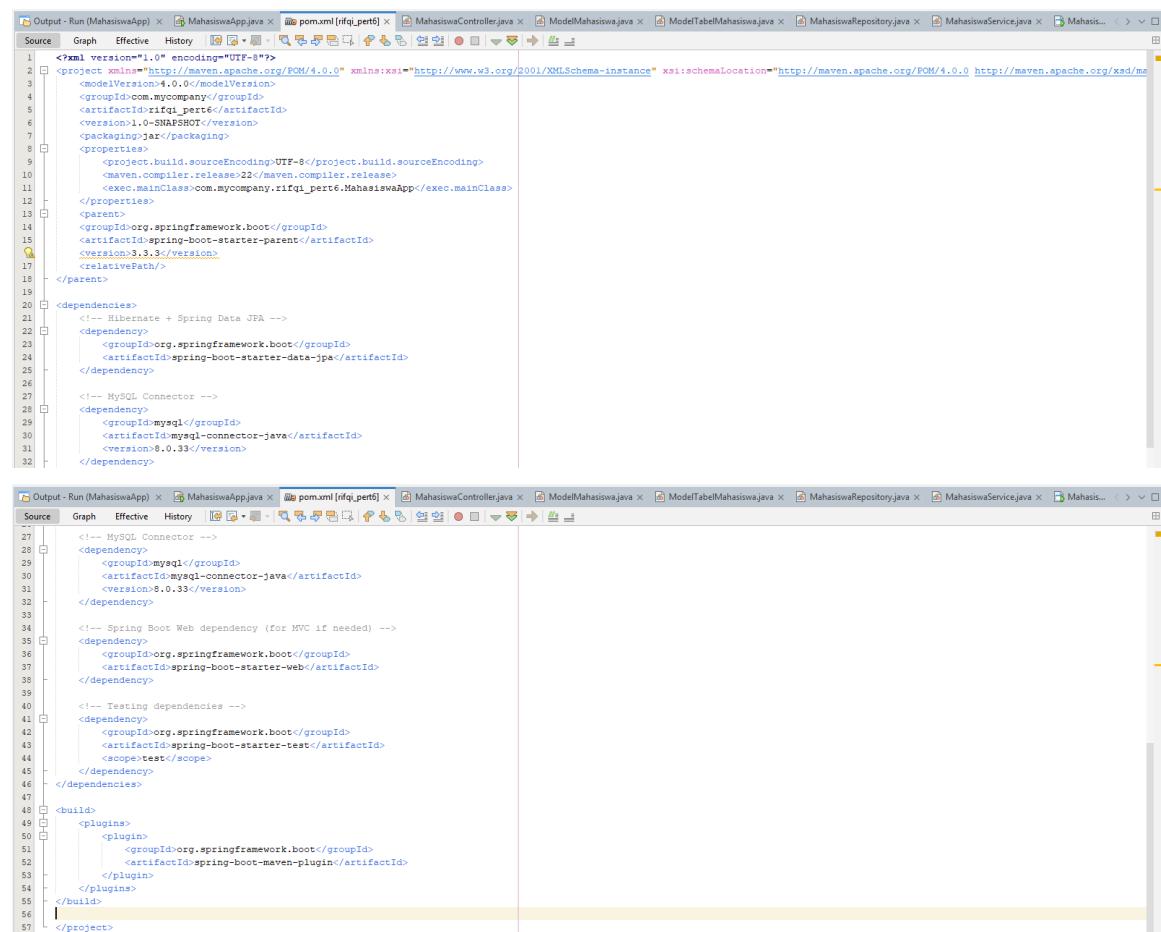
MATA PRAKTIKUM : Rekayasa Perangkat Lunak 2

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

1. Screenshot Kode dan Output Program!

Kode :

Pom.xml



```
<?xml version="1.0" encoding="UTF-8"?>
<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">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.mycompany</groupId>
  <artifactId>rifqi_pert6</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.release>22</maven.compiler.release>
    <exec.mainClass>com.mycompany.rifqi_pert6.MahasiswaApp</exec.mainClass>
  </properties>
  <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>
      <version>3.3.3</version>
    </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>
```

MahasiswaController.java

```
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_per6.controller;
6
7  import com.mycompany.rifqi_per6.model.ModelMahasiswa;
8  import com.mycompany.rifqi_per6.service.MahasiswaService;
9  import java.util.List;
10 import org.springframework.beans.factory.annotation.Autowired;
11 import org.springframework.stereotype.Controller;
12 import org.springframework.web.bind.annotation.PathVariable;
13 import org.springframework.web.bind.annotation.RequestBody;
14
15 /**
16  *
17  * @author mrifq
18  */
19
20 @Controller
21 public class MahasiswaController {
22     @Autowired
23     private MahasiswaService mahasiswaService;
24
25     public String addMahasiswa(@RequestBody ModelMahasiswa mhs) {
26         mahasiswaService.addMhs(mhs);
27         return "Mahasiswa added successfully";
28     }
29
30     public ModelMahasiswa getMahasiswa(@PathVariable int id) {
31         return mahasiswaService.getMhs(id);
32     }
33
34     public String updateMahasiswa(@RequestBody ModelMahasiswa mhs){
35         mahasiswaService.updateMhs(mhs);
36         return "Mahasiswa updated successfully";
37     }
38
39     public String deleteMahasiswa(@PathVariable int id) {
40         mahasiswaService.deleteMhs(id);
41         return "Mahasiswa deleted successfully";
42     }
43
44     public List<ModelMahasiswa> getAllMahasiswa() {
45         return mahasiswaService.getAllMahasiswa();
46     }
47 }
```

ModelMahasiswa.java

```
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_per6.model;
6
7  import jakarta.persistence.*;
8
9 /**
10  *
11  * @author mrifq
12  */
13
14 @Entity
15 @Table(name="mahasiswa")
16 public class ModelMahasiswa {
17     @Id
18     @GeneratedValue(strategy = GenerationType.IDENTITY)
19     @Column(name="id")
20     private int id;
21
22     @Column(name="npm", nullable=false, length=10)
23     private String npm;
24
25     @Column(name="nama", nullable=false, length=50)
26     private String nama;
27
28     @Column(name="semester", nullable=false)
29     private int semester;
30
31     @Column(name="ipk", nullable=false)
32     private float ipk;
33
34     public ModelMahasiswa() {
35     }
36
37     public ModelMahasiswa(int id, String npm, String nama, int semester, float ipk) {
38         this.id = id;
39         this.npm = npm;
40         this.nama = nama;
41         this.semester = semester;
42         this.ipk = ipk;
43     }
44     public int getId() {
45         return id;
46     }
47 }
```

ModelTabelMahasiswa.java

```
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 mrifq
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.getNama();
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 mrifq
14 */
15 @Repository
16 public interface MahasiswaRepository extends JpaRepository<ModelMahasiswa, Integer> {
17 }
```

MahasiswaService.java

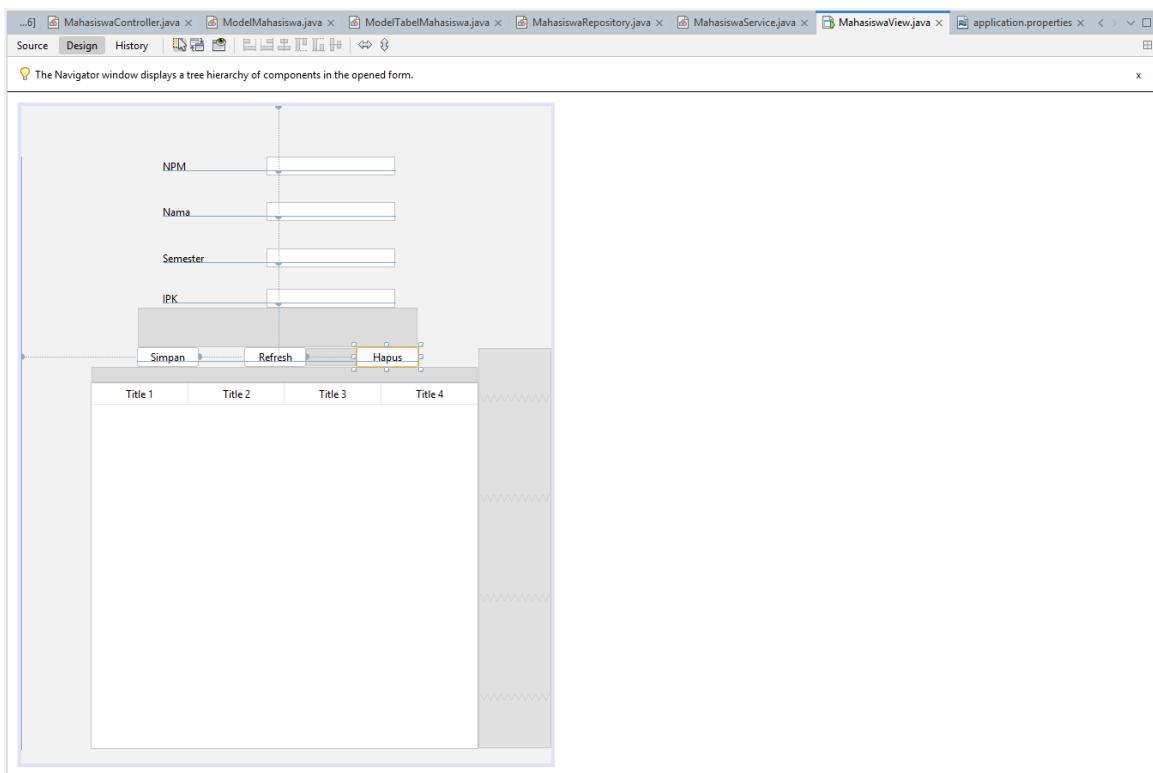
```
Output - Run (MahasiswaApp) × MahasiswaApp.java × pom.xml [rifqi_pert6] × MahasiswaController.java × ModelMahasiswa.java × ModelTabelMahasiswa.java × MahasiswaRepository.java × MahasiswaService.java × Mahasiswa... ×
```

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 mrifq
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 }
```

MahasiswaView.java

Design :



Source

```

1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
4  */
5  package com.mycompany.rifqi_per6.view;
6
7  import com.mycompany.rifqi_per6.controller.MahasiswaController;
8  import com.mycompany.rifqi_per6.model.ModelMahasiswa;
9  import com.mycompany.rifqi_per6.model.ModelTabelMahasiswa;
10 import java.util.List;
11 import javax.swing.JLabel;
12 import javax.swing.JOptionPane;
13 import javax.swing.JPanel;
14 import javax.swing.JTable;
15 import javax.swing.JTextField;
16
17 /**
18 * @author mrifq
19 */
20 public class MahasiswaView extends javax.swing.JFrame {
21
22     private MahasiswaController controller;
23
24     /**
25      * Creates new form MahasiswaView
26     */
27     public MahasiswaView(MahasiswaController controller) {
28         this.controller = controller;
29         initComponents();
30         loadMahasiswaTable();
31     }
32
33     private MahasiswaView() {
34         throw new UnsupportedOperationException("Not Supported yet.");
35     }
36
37     public void loadMahasiswaTable() {
38         List <ModelMahasiswa> listMahasiswa = controller.getAllMahasiswa();
39
40         ModelTabelMahasiswa tableModel = new ModelTabelMahasiswa(listMahasiswa);
41
42         dataTable.setModel(tableModel);
43     }
44
45 }

```

btnSimpan, btnRefresh, dan btnHapus

```

175     private void btnSaveActionPerformed(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.getId());
186         System.out.println(mahasiswa.getSemester());
187
188         controller.addMahasiswa(mahasiswa);
189         loadMahasiswaTable();
190     }
191
192     private void btnRefreshActionPerformed(java.awt.event.ActionEvent evt) {
193         // TODO add your handling code here:
194         loadMahasiswaTable();
195     }
196
197     private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {
198         // TODO add your handling code here:
199         JTextField idField = new JTextField(5);
200
201         // Membuat panel untuk menampung JTextField
202         JPanel panel = new JPanel();
203         panel.add(new JLabel("Masukkan ID yang ingin dihapus:"));
204         panel.add(idField);
205
206         // Menampilkan dialog box dengan JTextField, tombol OK, dan Cancel
207         int result = JOptionPane.showConfirmDialog(null, panel,
208             "Hapus Mahasiswa", JOptionPane.OK_CANCEL_OPTION, JOptionPane.PLAIN_MESSAGE);
209
210         // Jika tombol OK ditekan
211         if (result == JOptionPane.OK_OPTION) {
212             try {
213                 // Mengambil input ID dan memanggil metode deleteMhs
214                 int id = Integer.parseInt(idField.getText());
215                 controller.deleteMahasiswa(id);
216                 JOptionPane.showMessageDialog(null, "Data berhasil dihapus.", "Sukses", JOptionPane.INFORMATION_MESSAGE);
217             } catch (NumberFormatException e) {
218                 // Menangani error jika ID yang dimasukkan bukan angka
219                 JOptionPane.showMessageDialog(null, "ID harus berupa angka.", "Error", JOptionPane.ERROR_MESSAGE);
220             }
221         }
222     }

```

Application.properties

```

1
2 # Konfigurasi MySQL Hibernate
3 spring.datasource.url=jdbc:mysql://localhost:3306/per6_db?useSSL=false&serverTimezone=UTC
4 spring.datasource.username=root
5 spring.datasource.password=
6 spring.datasource.driver-classname=com.mysql.cj.jdbc.Driver
7 server.error=404
8
9 # Hibernate settings
10 spring.jpa.hibernate.ddl-auto:update
11 spring.jpa.show-sql=true
12
13
14
15

```

Database

Rain coming
in about 1 hour

11/22/2025 10:00 PM

Output :

Build Pom.xml

```

rifqi_per6 - Apache NetBeans IDE 25
Search (Ctrl+F) — O X
Output: Build Info: rifqi_per6
rifqi_per6.java x pom.xml [rifqi_per6]
cd C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6; *JAVA_HOME=C:\Program Files\Java\jdk-22* cmd /c "%C:\Program Files\NetBeans-25\java\maven\bin\mvn.cmd" ^*-Dmaven
Scanning for projects...
--- < com.mycompany:rifqi_per6 >-----
Building rifqi_per6 1.0-SNAPSHOT
from pom.xml
-----[ jar ]-----
The artifact mysql:mysql-connector-java:jar@0.33 has been relocated to com.mysql:mysql-connector-j:jar@8.0.33: MySQL Connector/J artifacts moved to reverse-ZBDS compliant Maven 2+ coordinate.
resources:3.1.1:resources (default-resources) @ rifqi_per6 ---
skip non existing resourceDirectory C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6\src\main\resources
skip non existing resourceDirectory C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6\src\main\resources

compiler:1.1.0:compile (default-compile) @ rifqi_per6 ---
Recompiling the module because of changed source code.
Compiling 1 source file with javac (Debug parameters release 22) to target\classes
resources:3.1.1:testResources (default-testResources) @ rifqi_per6 ---
skip non existing resourceDirectory C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6\src\test\resources

compiler:3.13.0:testCompile (default-testCompile) @ rifqi_per6 ---
No sources to compile

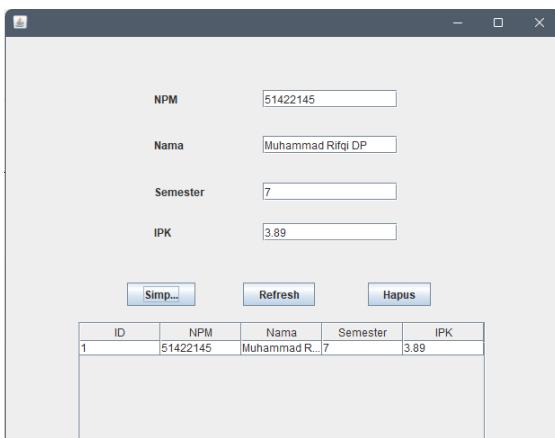
surefire:3.2.5:test (default-test) @ rifqi_per6 ---
No tests to run.

jar:3.4.2:jar (default-jar) @ rifqi_per6 ---
Building jar: C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6\target\rifqi_per6-1.0-SNAPSHOT.jar
spring-boot:3.2.3:repackage (repackage) @ rifqi_per6 ---
Replacing main artifact C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6\target\rifqi_per6-1.0-SNAPSHOT.jar with repackaged archive, adding nested dependencies in BOOT-INF.
The original artifact has been renamed to C:\Users\mrifq\Documents\NetBeansProjects\rifqi_per6\target\rifqi_per6-1.0-SNAPSHOT.jar.original
BUILD SUCCESS
Total time: 11.516 s
Finished at: 2025-11-22T12:09:10+07:00
-----
```

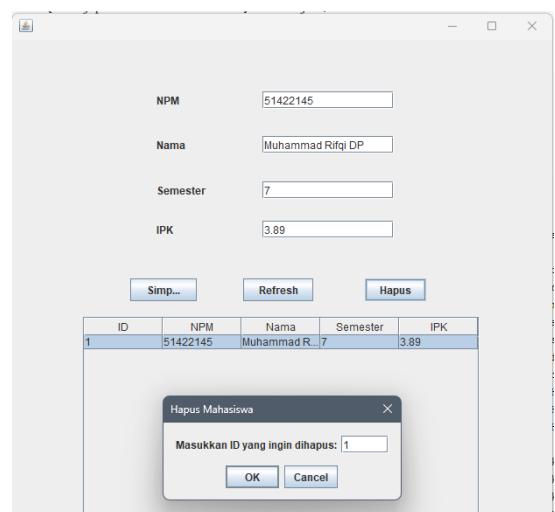
Rain coming
in about 1.5 hours

11/22/2025 12:09 PM

Simpan data



Hapus Data



Refresh Data

