PERTEMUAN

Nama : Teuku Ardhi Firmansyah Al Ghozali

NPM : 51421472

Kelas : 4IA28

PERTEMUAN 5

OUTPUT TAMBAH DATA

```
Menu:
1. Tampilkan semua mahasiswa
2. Tambah mahasiswa baru
3. Cek koneksi database
4. Keluar
Pilih opsi: 2
Masukkan NPM : 51421472
Masukkan Nama : Teuku
Masukkan Semester : 7
Masukkan IPK : 4
Hibernate: insert into mahasiswa (ipk,nama,npm,semester) values (?,?,?,?)
Mahasiswa berhasil ditambahkan.

Menu:
1. Tampilkan semua mahasiswa
2. Tambah mahasiswa baru
3. Cek koneksi database
4. Keluar
Pilih opsi: 2
Masukkan NPM : 51421231
Masukkan Nama : Raihan
Masukkan Semester : 9
Masukkan IPK : 4
Hibernate: insert into mahasiswa (ipk,nama,npm,semester) values (?,?,?,?)
Mahasiswa berhasil ditambahkan.
```

```
Menu:
1. Tampilkan semua mahasiswa
2. Tambah mahasiswa baru
3. Cek koneksi database
4. Keluar
Pilih opsi: 2
Masukkan NPM : 5121123
Masukkan Nama : Adit
Masukkan Semester : 7
Masukkan IPK : 4
Hibernate: insert into mahasiswa (ipk,nama,npm,semester) values (?,?,?,?)
Mahasiswa berhasil ditambahkan.

Menu:
1. Tampilkan semua mahasiswa
2. Tambah mahasiswa baru
3. Cek koneksi database
4. Keluar
Pilih opsi: 1
Hibernate: select mm1_0.id,mm1_0.ipk,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0
Mahasiswa{id=1, nama='Teuku', nama='51421472', nama='7', jurusan='4.0'}
Mahasiswa{id=2, nama='Raihan', nama='51421231', nama='9', jurusan='4.0'}
Mahasiswa{id=3, nama='Adit', nama='5121123', nama='7', jurusan='4.0'}
```

Pertemuan5SpringBootApplication

```java
package com.mahasiswa;
import com.mahasiswa.controller.MahasiswaController;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.CommandLineRunner;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

/**
 *
 * @author ASUS
 */
@SpringBootApplication
public class Pertemuan5SpringBootApplication implements CommandLineRunner{

    @Autowired
    private MahasiswaController mhsController;
    public static void main(String[] args) {
        SpringApplication.run(Pertemuan5SpringBootApplication.class, args);
    }

    @Override
    public void run(String... args) throws Exception {
        mhsController.tampilkanMenu();
    }

}
```

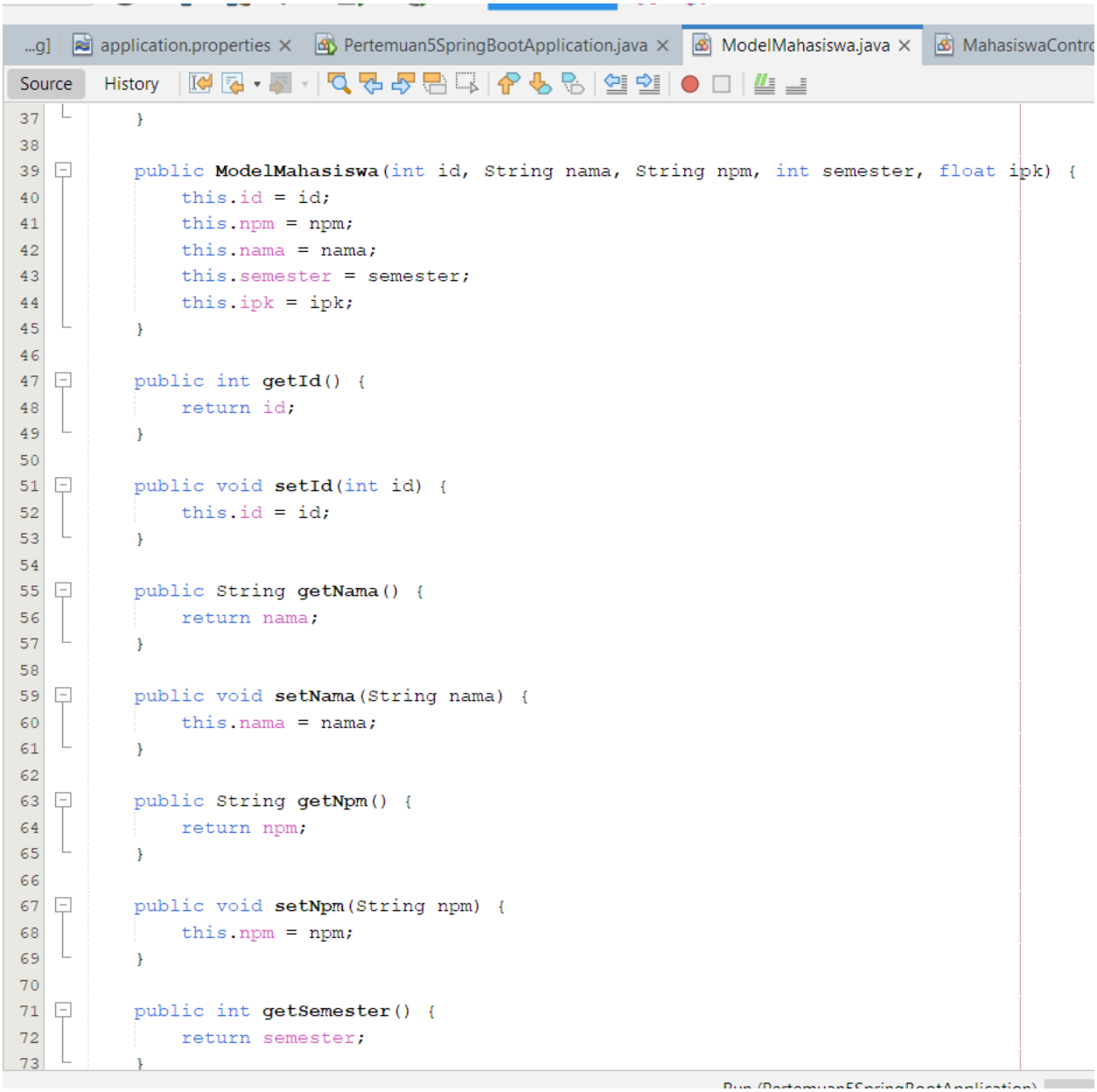
MahasiswaController

```java
package com.mahasiswa.controller;

import com.mahasiswa.model.ModelMahasiswa;
import com.mahasiswa.repository.MahasiswaRepository;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;

import java.util.List;
import java.util.Scanner;

@Controller
public class MahasiswaController {

    @Autowired
    private MahasiswaRepository mahasiswaRepository;

    public void tampilkanMenu() {
        Scanner scanner = new Scanner(System.in);
        int opsi;

        do {
            System.out.println("\nMenu:");
            System.out.println("1. Tampilkan semua mahasiswa");
            System.out.println("2. Tambah mahasiswa baru");
            System.out.println("3. Cek koneksi database");
            System.out.println("4. Keluar");
            System.out.print("Pilih opsi: ");
            opsi = scanner.nextInt();
            scanner.nextLine(); // menangkap newline

            switch (opsi) {
                case 1:
                    tampilkanSemuaMahasiswa();
                    break;
                case 2:
                    tambahMahasiswa(scanner);
                    break;
                case 3:
                    cekKoneksi();
```

```java
                    System.out.println("Opsi tidak valid, coba lagi.");
                }

            } while (opsi != 4);
        }

        private void tampilkanSemuaMahasiswa() {
            List<ModelMahasiswa> mahasiswaList = mahasiswaRepository.findAll();
            if (mahasiswaList.isEmpty()) {
                System.out.println("Tidak ada data mahasiswa.");
            } else {
                mahasiswaList.forEach(mahasiswa -> System.out.println(mahasiswa));
            }
        }

        private void tambahMahasiswa(Scanner scanner) {
            System.out.print("Masukkan NPM : ");
            String npm = scanner.nextLine();
            System.out.print("Masukkan Nama : ");
            String nama = scanner.nextLine();
            System.out.print("Masukkan Semester : ");
            int semester = scanner.nextInt();
            System.out.print("Masukkan IPK : ");
            float ipk = scanner.nextFloat();

            ModelMahasiswa mahasiswa = new ModelMahasiswa(0, npm, nama, semester, ipk);
            mahasiswaRepository.save(mahasiswa);
            System.out.println("Mahasiswa berhasil ditambahkan.");
        }

        private void cekKoneksi() {
            try {
                mahasiswaRepository.findAll();
                System.out.println("Koneksi ke database berhasil.");
            } catch (Exception e) {
                System.out.println("Gagal terhubung ke database.");
            }
        }
    }
```

ModelMahasiswa

```java
/*
 * 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.mahasiswa.model;

import jakarta.persistence.Entity;
import jakarta.persistence.*;

/**
 *
 * @author Teuku
 */

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

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

    @Column(name = "npm", nullable = false, length = 8)
    private String npm;

    @Column(name = "semester")
    private int semester;

    @Column(name = "ipk")
    private float ipk;

    public ModelMahasiswa() {

    }
```




```
37      }
38
39      public ModelMahasiswa(int id, String nama, String npm, int semester, float ipk) {
40          this.id = id;
41          this.npm = npm;
42          this.nama = nama;
43          this.semester = semester;
44          this.ipk = ipk;
45      }
46
47      public int getId() {
48          return id;
49      }
50
51      public void setId(int id) {
52          this.id = id;
53      }
54
55      public String getNama() {
56          return nama;
57      }
58
59      public void setNama(String nama) {
60          this.nama = nama;
61      }
62
63      public String getNpm() {
64          return npm;
65      }
66
67      public void setNpm(String npm) {
68          this.npm = npm;
69      }
70
71      public int getSemester() {
72          return semester;
73      }
```

```
    }

    public int getSemester() {
        return semester;
    }

    public void setSemester(int semester) {
        this.semester = semester;
    }

    public float getIpk() {
        return ipk;
    }

    public void setIpk(float ipk) {
        this.ipk = ipk;
    }

    @Override
    public String toString() {
        return "Mahasiswa{"
                + "id=" + id
                + ", nama='" + npm + '\''
                + ", nama='" + nama + '\''
                + ", nama='" + semester + '\''
                + ", jurusan='" + ipk + '\''
                + '}';
    }
}
```

PERTEMUAN

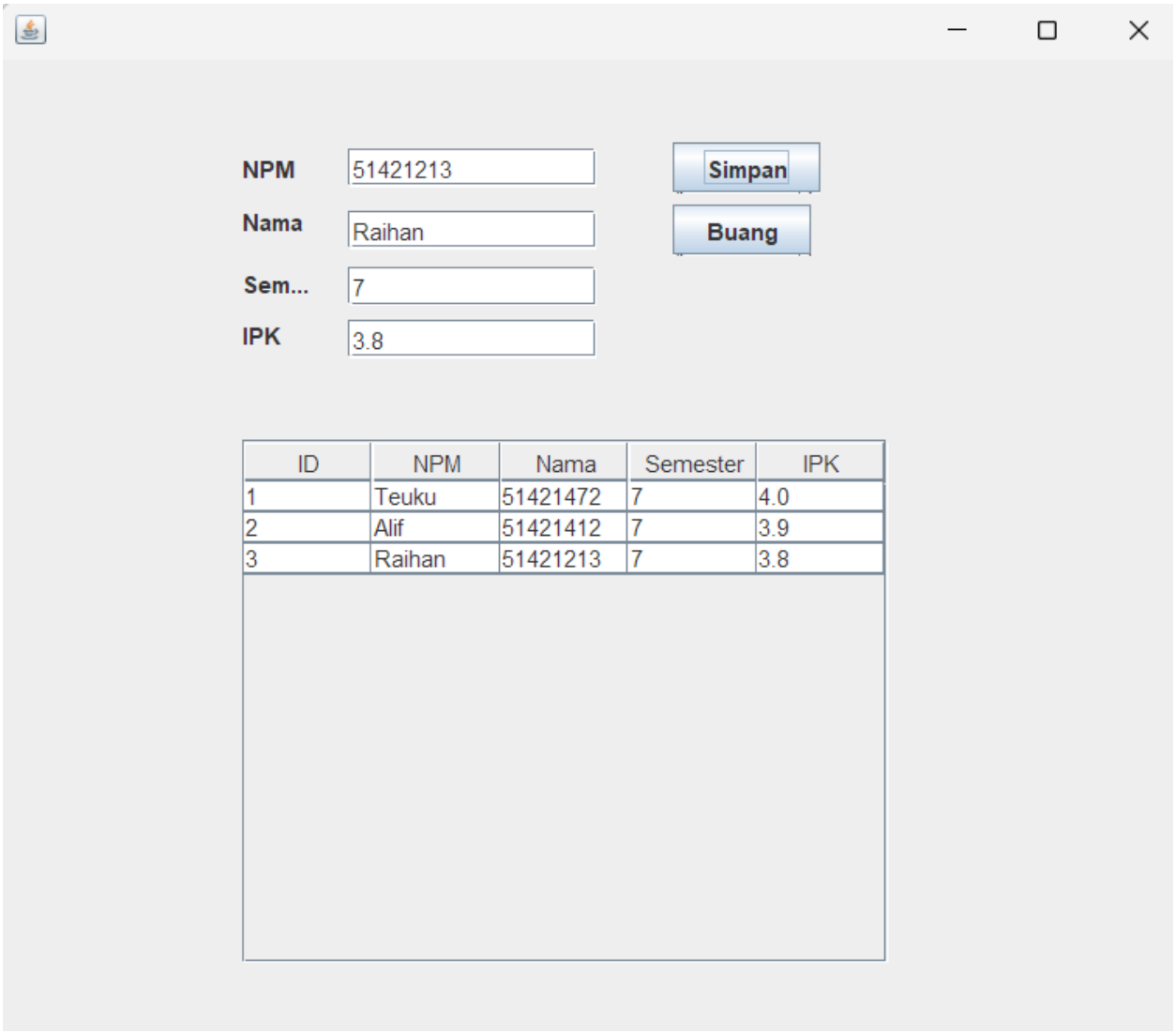
Nama : Teuku Ardhi Firmansyah Al Ghozali

NPM : 51421472

Kelas : 4IA28

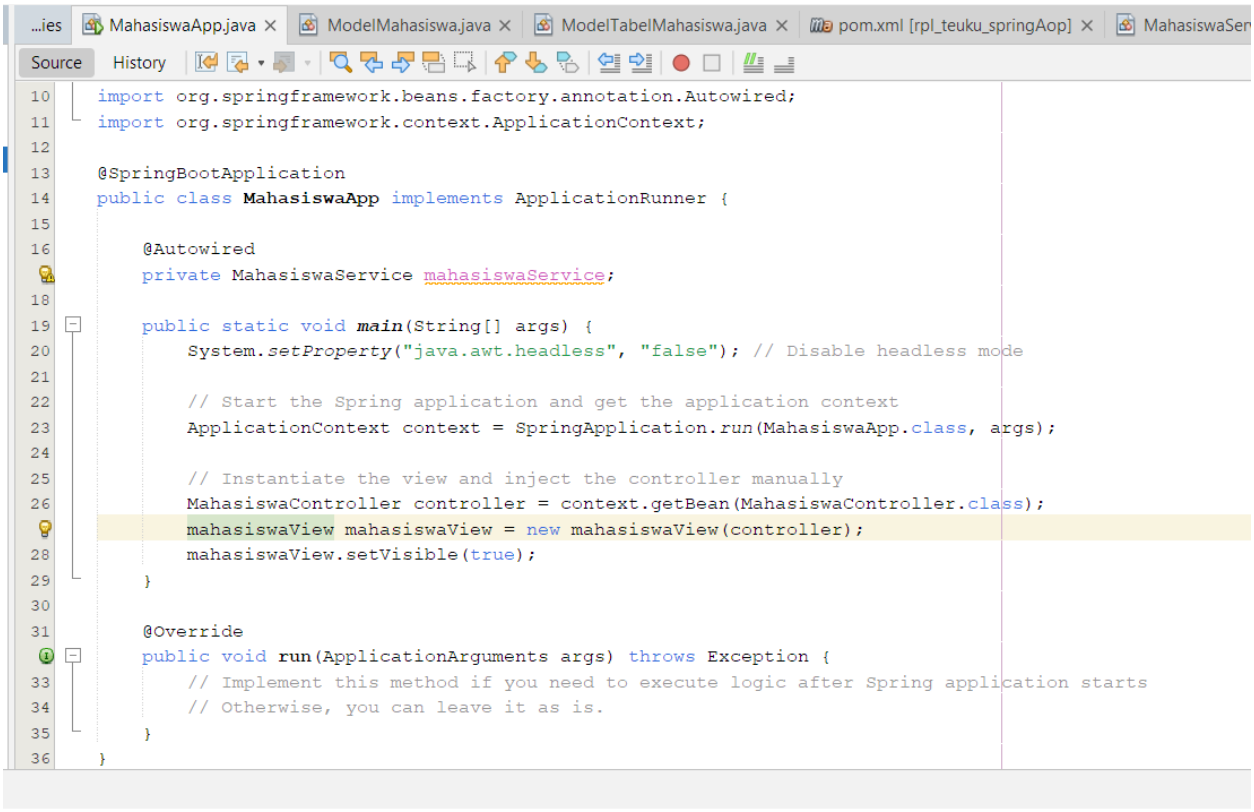
PERTEMUAN 6

OUTPUT 3 DATA



CODE

MahasiswaApp

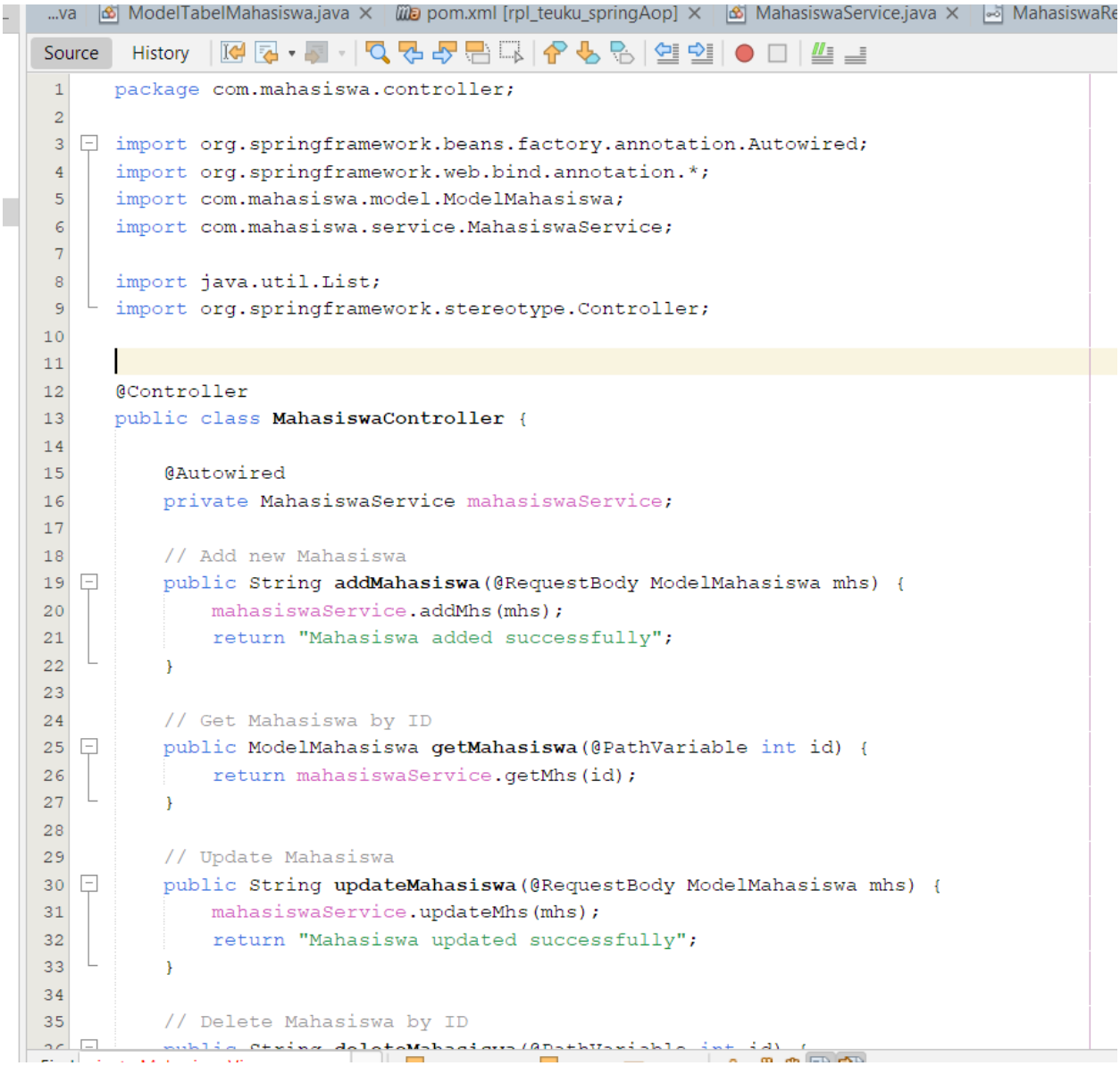


```
...ies | MahasiswaApp.java | ModelMahasiswa.java | ModelTabelMahasiswa.java | pom.xml [rpl_teuku_springAop] | MahasiswaSer
Source  History
```

```java
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.context.ApplicationContext;

@SpringBootApplication
public class MahasiswaApp implements ApplicationRunner {

    @Autowired
    private MahasiswaService mahasiswaService;

    public static void main(String[] args) {
        System.setProperty("java.awt.headless", "false"); // Disable headless mode

        // Start the Spring application and get the application context
        ApplicationContext context = SpringApplication.run(MahasiswaApp.class, args);

        // Instantiate the view and inject the controller manually
        MahasiswaController controller = context.getBean(MahasiswaController.class);
        mahasiswaView mahasiswaView = new mahasiswaView(controller);
        mahasiswaView.setVisible(true);
    }

    @Override
    public void run(ApplicationArguments args) throws Exception {
        // Implement this method if you need to execute logic after Spring application starts
        // Otherwise, you can leave it as is.
    }
}
```

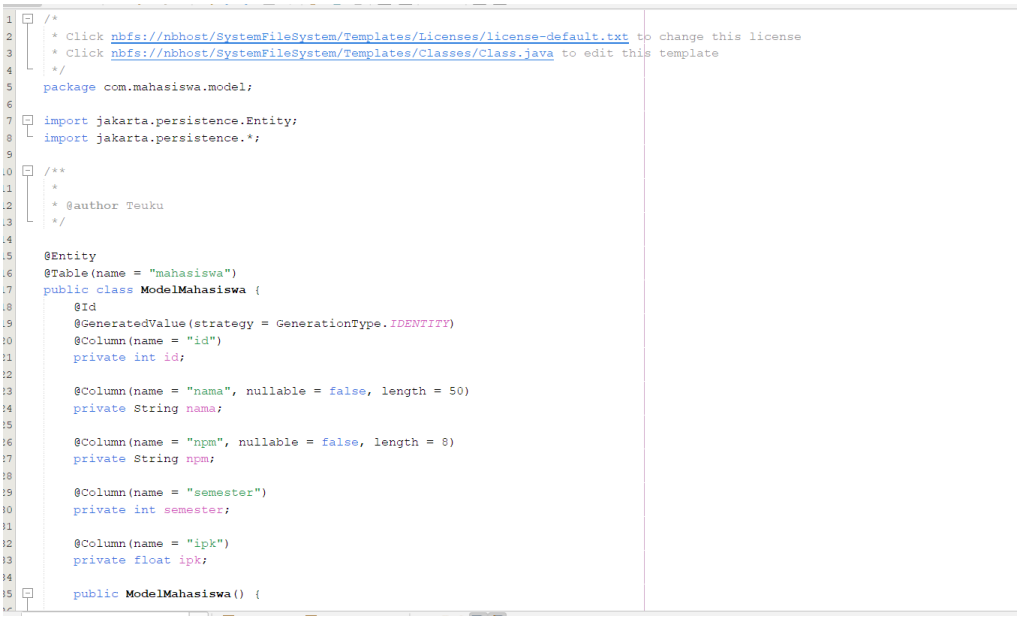
```
k,mm1_0.nama,mm1_0.npm,mm1_0.semester from mahasiswa mm1_0
```

MahasiswaController



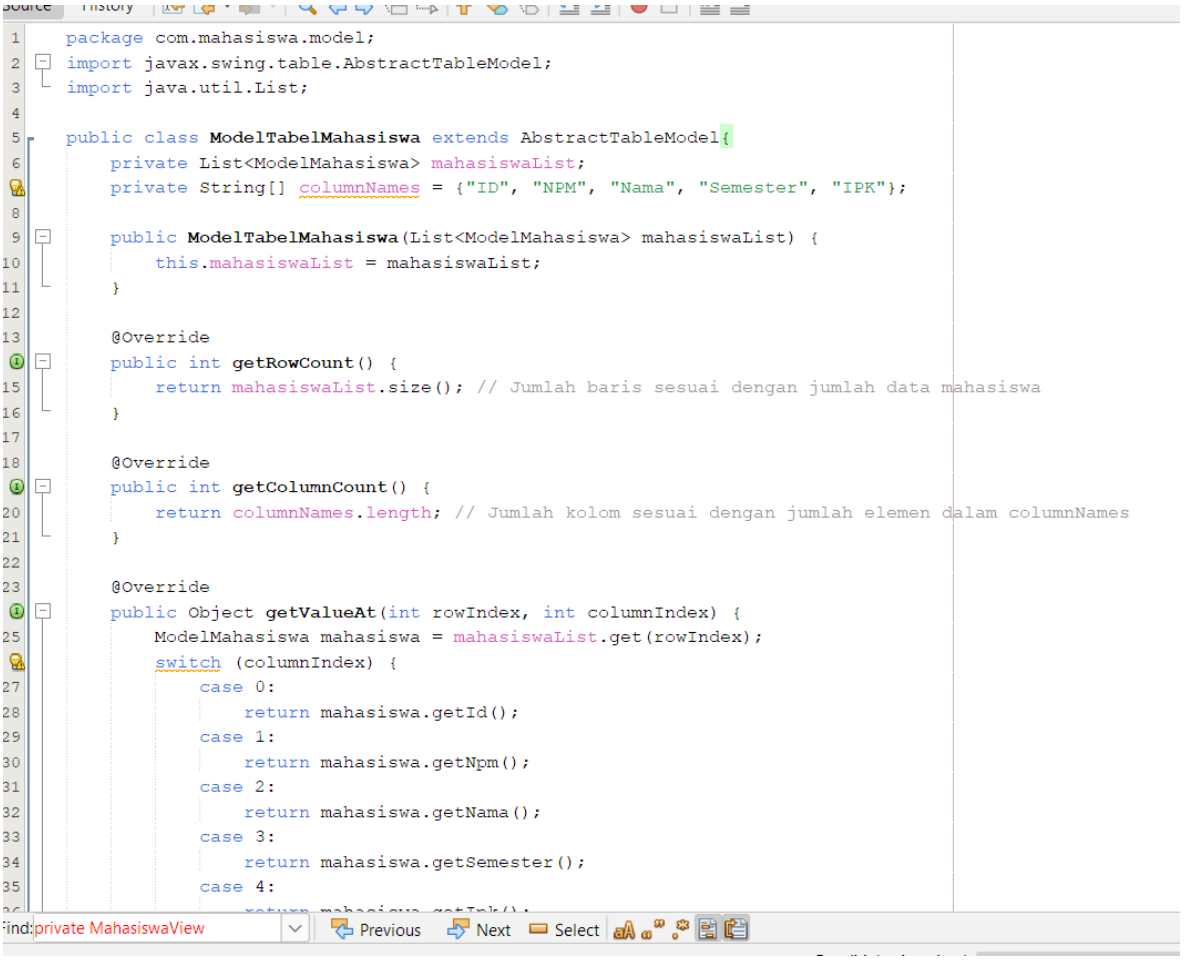

```java
package com.mahasiswa.controller;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.*;
import com.mahasiswa.model.ModelMahasiswa;
import com.mahasiswa.service.MahasiswaService;

import java.util.List;
import org.springframework.stereotype.Controller;


@Controller
public class MahasiswaController {

    @Autowired
    private MahasiswaService mahasiswaService;

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

    // Get Mahasiswa by ID
    public ModelMahasiswa getMahasiswa(@PathVariable int id) {
        return mahasiswaService.getMhs(id);
    }

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

    // Delete Mahasiswa by ID
```

ModelMahasiswa



```java
/*
 * 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.mahasiswa.model;

import jakarta.persistence.Entity;
import jakarta.persistence.*;

/**
 *
 * @author Teuku
 */

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

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

    @Column(name = "npm", nullable = false, length = 8)
    private String npm;

    @Column(name = "semester")
    private int semester;

    @Column(name = "ipk")
    private float ipk;

    public ModelMahasiswa() {
```

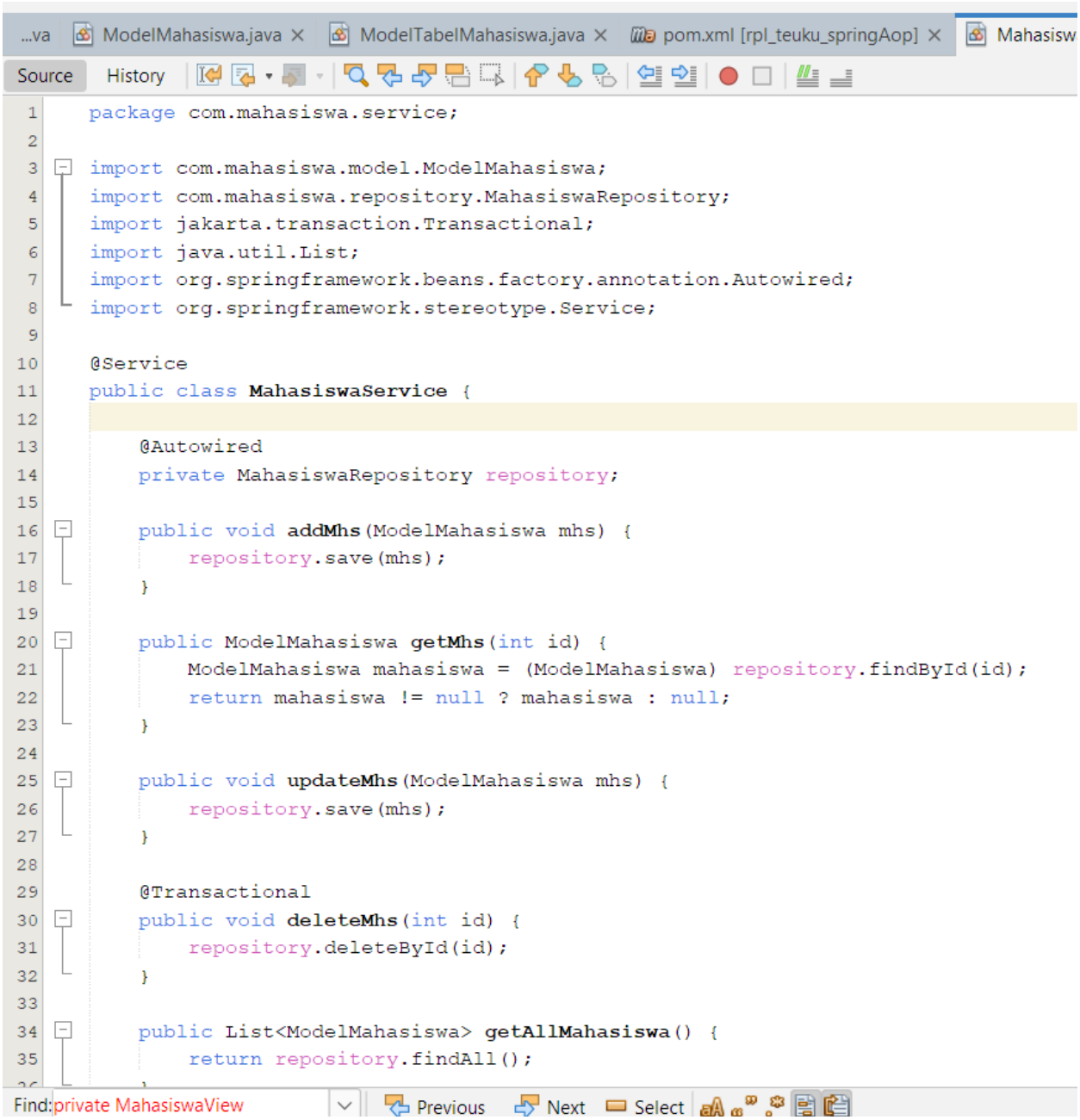
ModelTabelMahasiswa



```java
package com.mahasiswa.model;
import javax.swing.table.AbstractTableModel;
import java.util.List;

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

    public ModelTabelMahasiswa(List<ModelMahasiswa> mahasiswaList) {
        this.mahasiswaList = mahasiswaList;
    }

    @Override
    public int getRowCount() {
        return mahasiswaList.size(); // Jumlah baris sesuai dengan jumlah data mahasiswa
    }

    @Override
    public int getColumnCount() {
        return columnNames.length; // Jumlah kolom sesuai dengan jumlah elemen dalam columnNames
    }

    @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:
```

Find: private MahasiswaView Previous Next Select

MahasiswaService



MahasiswaView