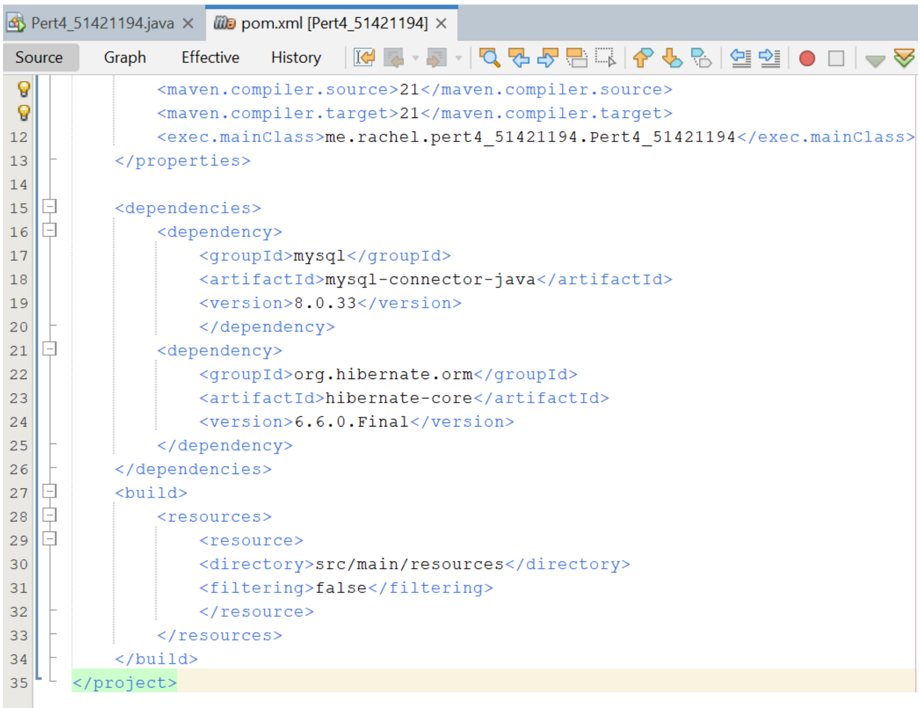
Nama : Rachel Amelia Lesmana

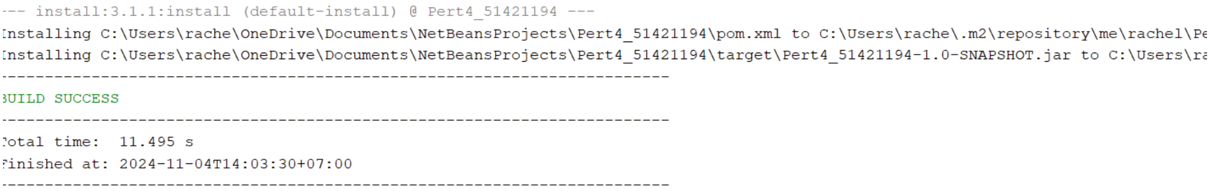
Kelas : 4IA14

NPM : 51421194

Pom

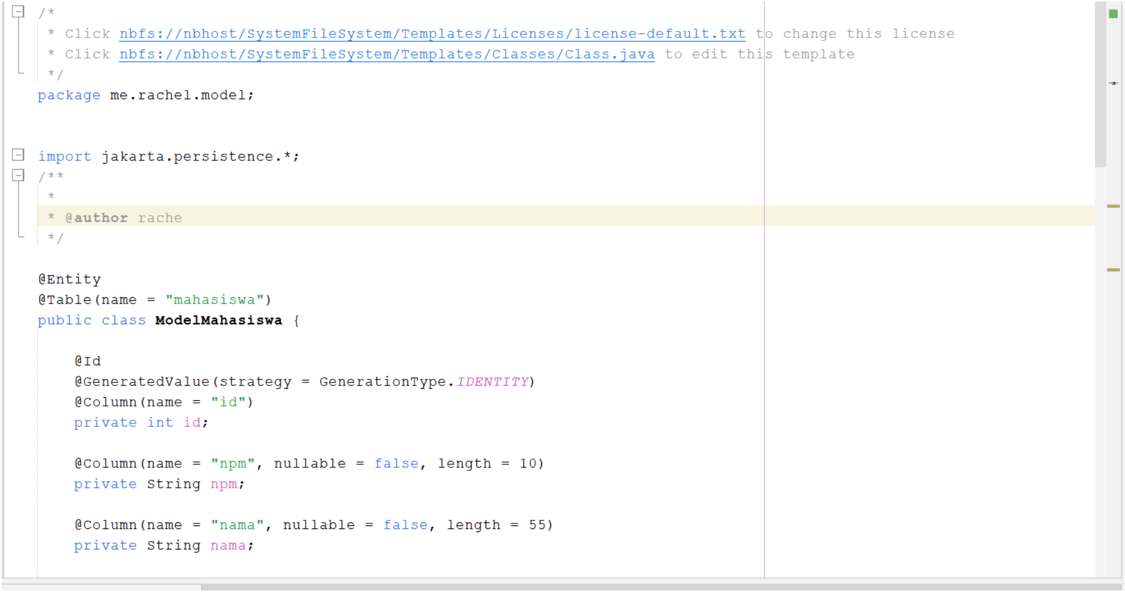


```
        <maven.compiler.source>21</maven.compiler.source>
        <maven.compiler.target>21</maven.compiler.target>
        <exec.mainClass>me.rachel.pert4_51421194.Pert4_51421194</exec.mainClass>
    </properties>

    <dependencies>
        <dependency>
            <groupId>mysql</groupId>
            <artifactId>mysql-connector-java</artifactId>
            <version>8.0.33</version>
            </dependency>
        <dependency>
            <groupId>org.hibernate.orm</groupId>
            <artifactId>hibernate-core</artifactId>
            <version>6.6.0.Final</version>
        </dependency>
    </dependencies>
    <build>
        <resources>
            <resource>
            <directory>src/main/resources</directory>
            <filtering>false</filtering>
            </resource>
        </resources>
    </build>
</project>
```

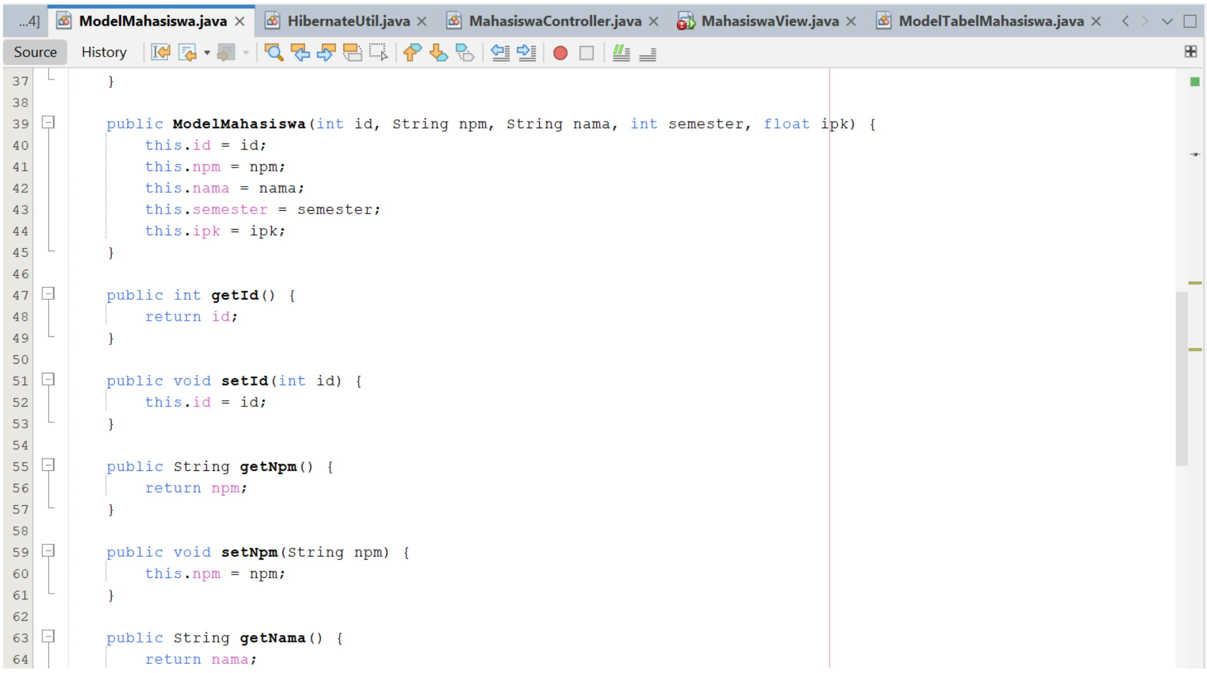


```
--- install:3.1.1:install (default-install) @ Pert4_51421194 ---
Installing C:\Users\rache\OneDrive\Documents\NetBeansProjects\Pert4_51421194\pom.xml to C:\Users\rache\.m2\repository\me\rachel\Pe
Installing C:\Users\rache\OneDrive\Documents\NetBeansProjects\Pert4_51421194\target\Pert4_51421194-1.0-SNAPSHOT.jar to C:\Users\ra
------------------------------------------------------------------------
BUILD SUCCESS
------------------------------------------------------------------------
Total time:  11.495 s
Finished at: 2024-11-04T14:03:30+07:00
------------------------------------------------------------------------
```

Model Mahasiswa

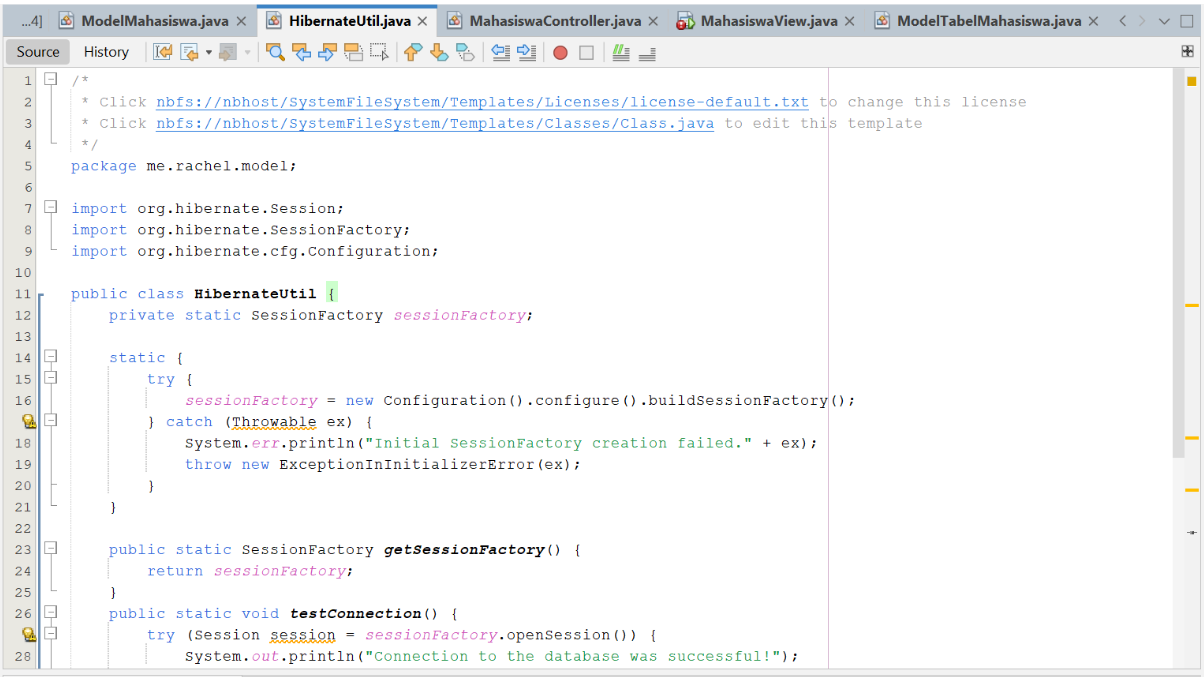


```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package me.rachel.model;


import jakarta.persistence.*;
/**
 *
 * @author rache
 */

@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 = 55)
    private String nama;
```



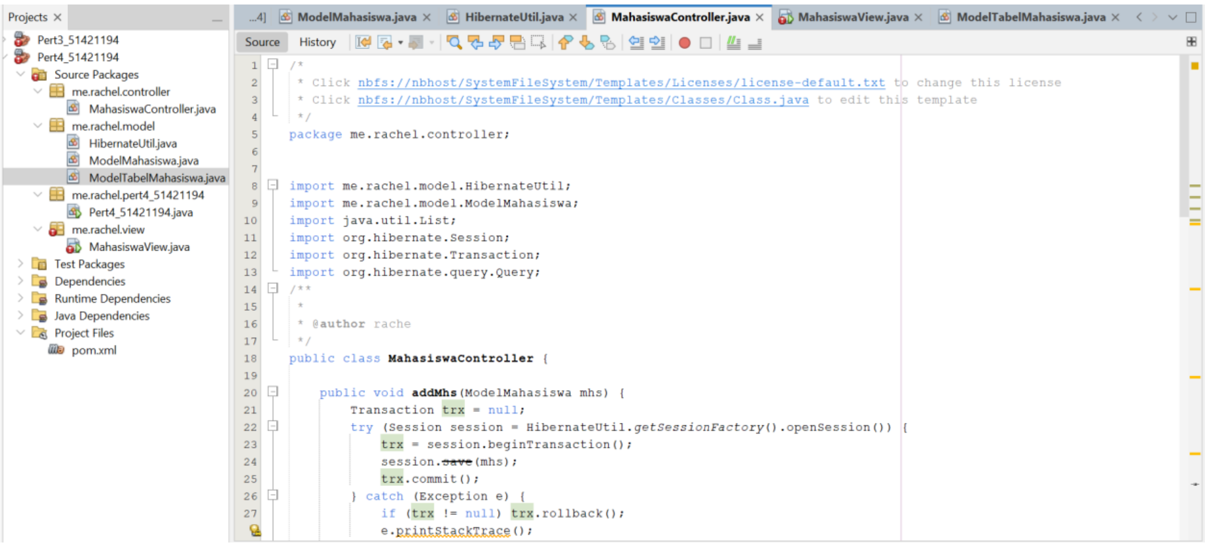
```java
    }

    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;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getNpm() {
        return npm;
    }

    public void setNpm(String npm) {
        this.npm = npm;
    }

    public String getNama() {
        return nama;
```

Hibernate Util

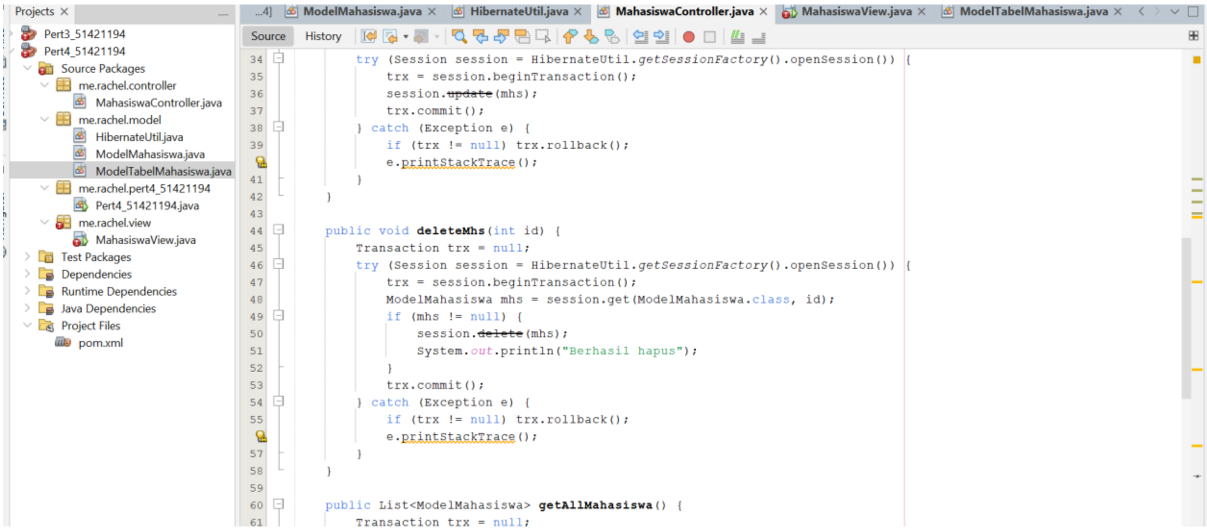


```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package me.rachel.model;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class HibernateUtil {
    private static SessionFactory sessionFactory;

    static {
        try {
            sessionFactory = new Configuration().configure().buildSessionFactory();
        } catch (Throwable ex) {
            System.err.println("Initial SessionFactory creation failed." + ex);
            throw new ExceptionInInitializerError(ex);
        }
    }

    public static SessionFactory getSessionFactory() {
        return sessionFactory;
    }
    public static void testConnection() {
        try (Session session = sessionFactory.openSession()) {
            System.out.println("Connection to the database was successful!");
```

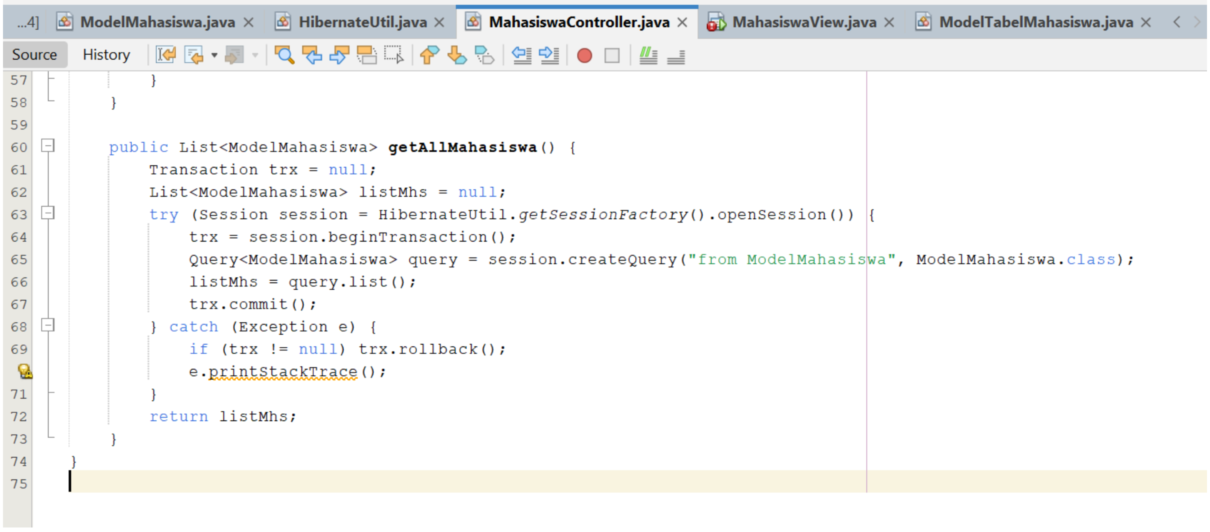
Mahasiswa Controller



```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 me.rachel.controller;


import me.rachel.model.HibernateUtil;
import me.rachel.model.ModelMahasiswa;
import java.util.List;
import org.hibernate.Session;
import org.hibernate.Transaction;
import org.hibernate.query.Query;
/**
 *
 * @author rache
 */
public class MahasiswaController {

    public void addMhs(ModelMahasiswa mhs) {
        Transaction trx = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            trx = session.beginTransaction();
            session.~~save~~(mhs);
            trx.commit();
        } catch (Exception e) {
            if (trx != null) trx.rollback();
            e.printStackTrace();
```

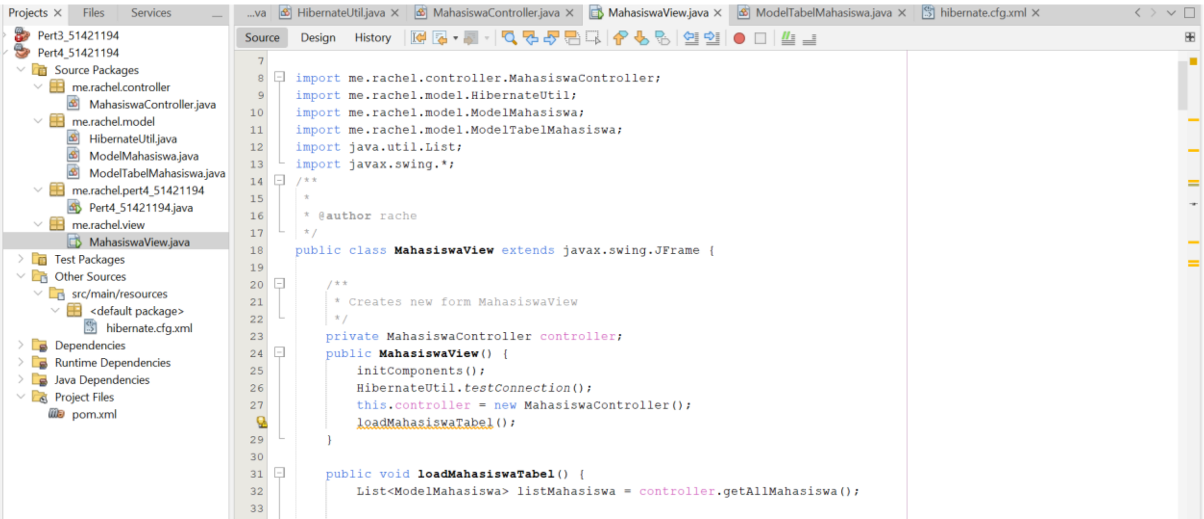


```java
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            trx = session.beginTransaction();
            session.~~update~~(mhs);
            trx.commit();
        } catch (Exception e) {
            if (trx != null) trx.rollback();
            e.printStackTrace();
        }
    }

    public void deleteMhs(int id) {
        Transaction trx = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            trx = session.beginTransaction();
            ModelMahasiswa mhs = session.get(ModelMahasiswa.class, id);
            if (mhs != null) {
                session.~~delete~~(mhs);
                System.out.println("Berhasil hapus");
            }
            trx.commit();
        } catch (Exception e) {
            if (trx != null) trx.rollback();
            e.printStackTrace();
        }
    }

    public List<ModelMahasiswa> getAllMahasiswa() {
        Transaction trx = null;
```

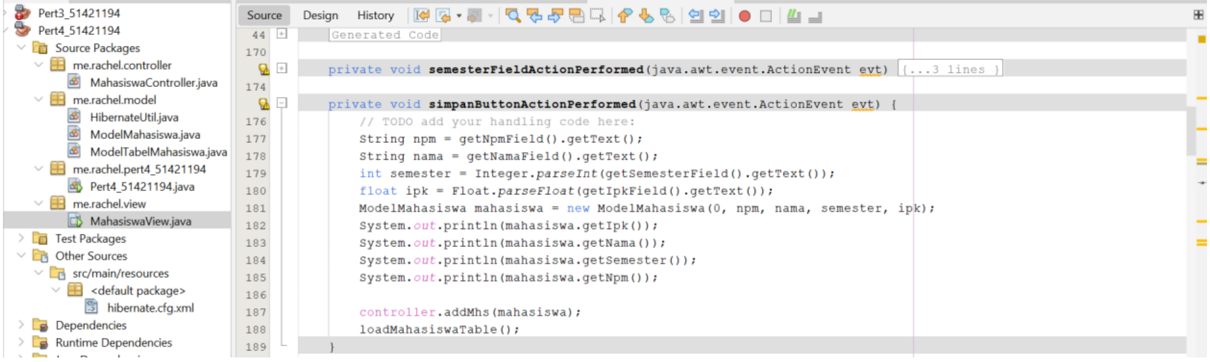


```java
        }
    }

    public List<ModelMahasiswa> getAllMahasiswa() {
        Transaction trx = null;
        List<ModelMahasiswa> listMhs = null;
        try (Session session = HibernateUtil.getSessionFactory().openSession()) {
            trx = session.beginTransaction();
            Query<ModelMahasiswa> query = session.createQuery("from ModelMahasiswa", ModelMahasiswa.class);
            listMhs = query.list();
            trx.commit();
        } catch (Exception e) {
            if (trx != null) trx.rollback();
            e.printStackTrace();
        }
        return listMhs;
    }
}
```

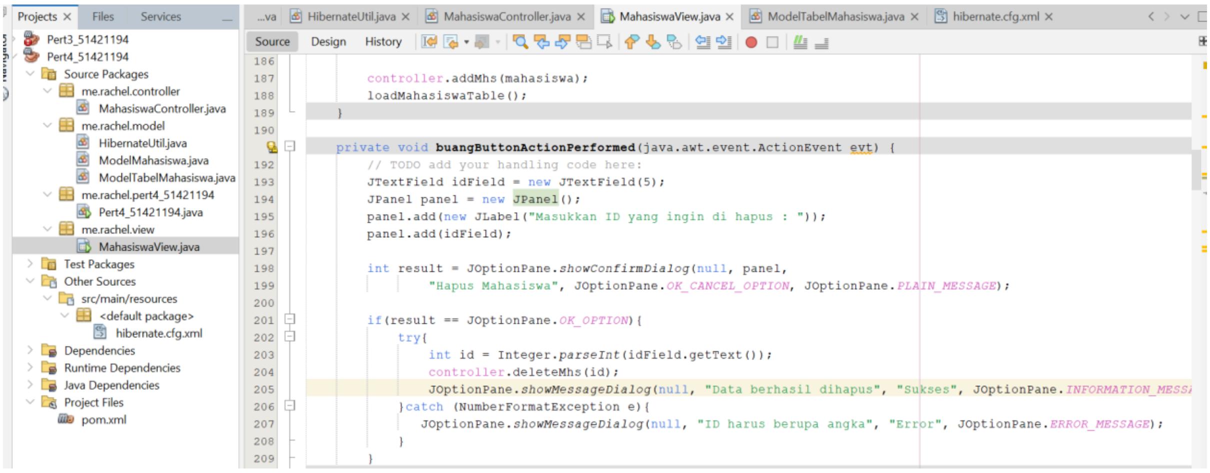
Mahasiswa View



```java

import me.rachel.controller.MahasiswaController;
import me.rachel.model.HibernateUtil;
import me.rachel.model.ModelMahasiswa;
import me.rachel.model.ModelTabelMahasiswa;
import java.util.List;
import javax.swing.*;
/**
 *
 * @author rache
 */
public class MahasiswaView extends javax.swing.JFrame {

    /**
     * Creates new form MahasiswaView
     */
    private MahasiswaController controller;
    public MahasiswaView() {
        initComponents();
        HibernateUtil.testConnection();
        this.controller = new MahasiswaController();
        loadMahasiswaTabel();
    }

    public void loadMahasiswaTabel() {
        List<ModelMahasiswa> listMahasiswa = controller.getAllMahasiswa();

```

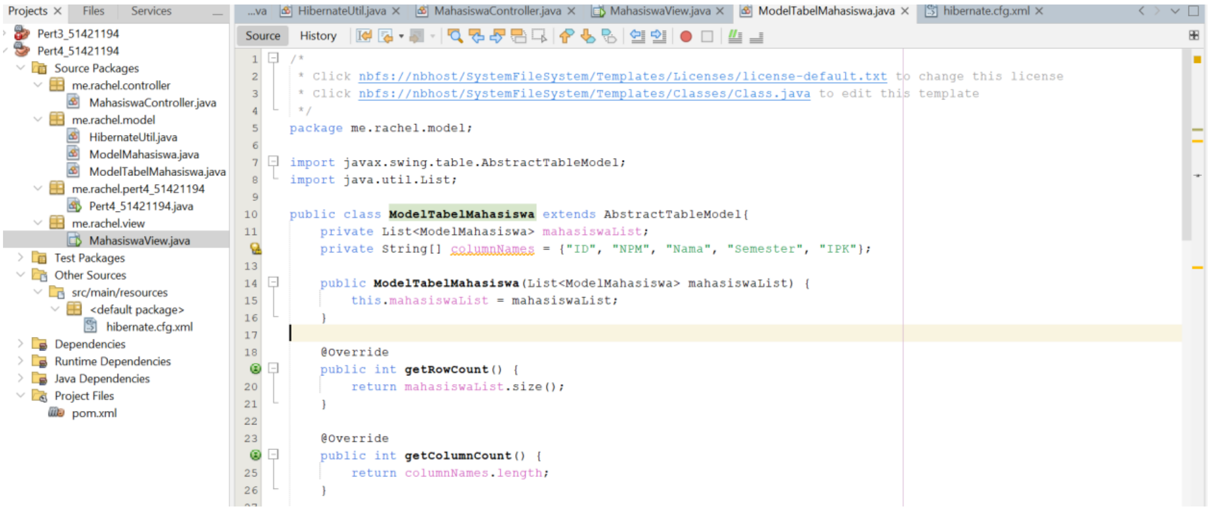


```java
    Generated Code

    private void semesterFieldActionPerformed(java.awt.event.ActionEvent evt) {...3 lines }

    private void simpanButtonActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        String npm = getNpmField().getText();
        String nama = getNamaField().getText();
        int semester = Integer.parseInt(getSemesterField().getText());
        float ipk = Float.parseFloat(getIpkField().getText());
        ModelMahasiswa mahasiswa = new ModelMahasiswa(0, npm, nama, semester, ipk);
        System.out.println(mahasiswa.getIpk());
        System.out.println(mahasiswa.getNama());
        System.out.println(mahasiswa.getSemester());
        System.out.println(mahasiswa.getNpm());

        controller.addMhs(mahasiswa);
        loadMahasiswaTable();
    }
```



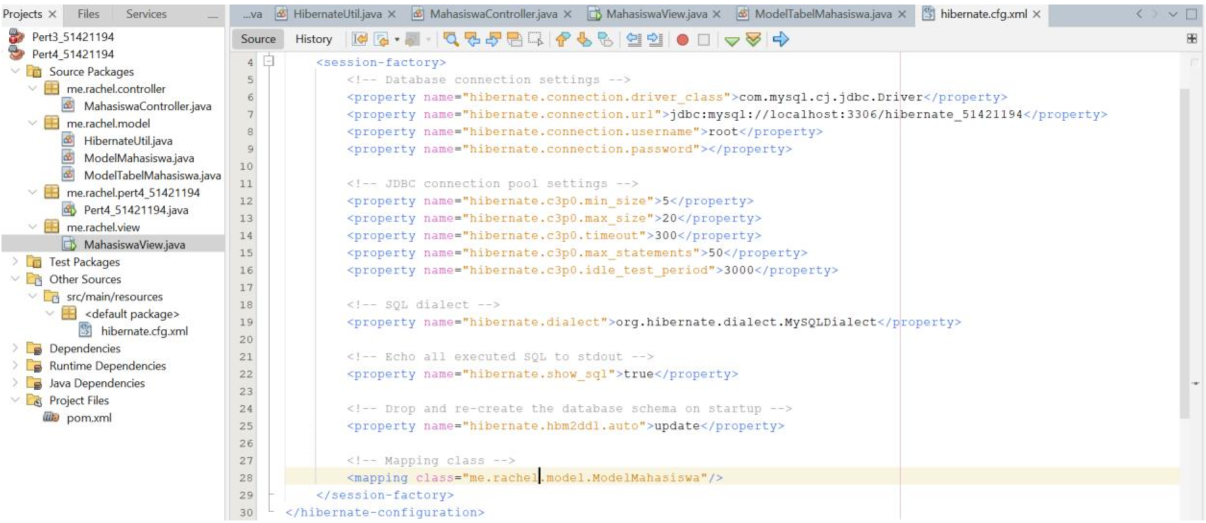
```java

        controller.addMhs(mahasiswa);
        loadMahasiswaTable();
    }

    private void buangButtonActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        JTextField idField = new JTextField(5);
        JPanel panel = new JPanel();
        panel.add(new JLabel("Masukkan ID yang ingin di hapus : "));
        panel.add(idField);

        int result = JOptionPane.showConfirmDialog(null, panel,
                "Hapus Mahasiswa", JOptionPane.OK_CANCEL_OPTION, JOptionPane.PLAIN_MESSAGE);

        if(result == JOptionPane.OK_OPTION){
            try{
                int id = Integer.parseInt(idField.getText());
                controller.deleteMhs(id);
                JOptionPane.showMessageDialog(null, "Data berhasil dihapus", "Sukses", JOptionPane.INFORMATION_MESS
            }catch (NumberFormatException e){
               JOptionPane.showMessageDialog(null, "ID harus berupa angka", "Error", JOptionPane.ERROR_MESSAGE);
            }
        }
```

Model Tabel Mahasiswa



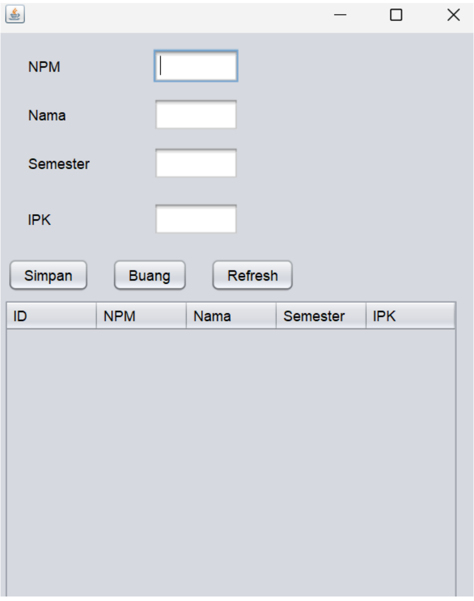
```java
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
 */
package me.rachel.model;

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();
    }

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

Hibernate.cfg



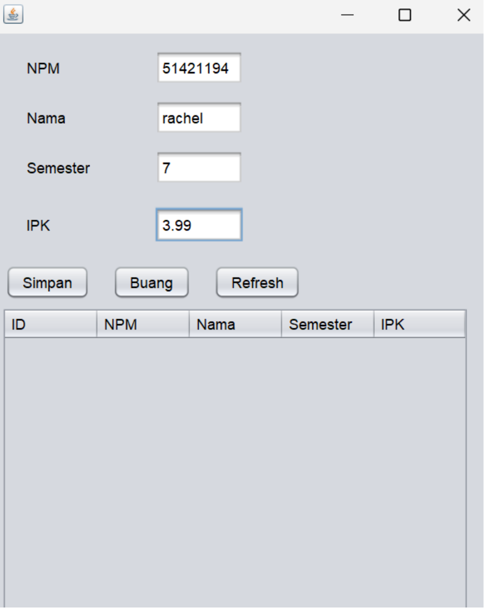
```xml
    <session-factory>
        <!-- Database connection settings -->
        <property name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/hibernate_51421194</property>
        <property name="hibernate.connection.username">root</property>
        <property name="hibernate.connection.password"></property>

        <!-- JDBC connection pool settings -->
        <property name="hibernate.c3p0.min_size">5</property>
        <property name="hibernate.c3p0.max_size">20</property>
        <property name="hibernate.c3p0.timeout">300</property>
        <property name="hibernate.c3p0.max_statements">50</property>
        <property name="hibernate.c3p0.idle_test_period">3000</property>

        <!-- SQL dialect -->
        <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

        <!-- Echo all executed SQL to stdout -->
        <property name="hibernate.show_sql">true</property>

        <!-- Drop and re-create the database schema on startup -->
        <property name="hibernate.hbm2ddl.auto">update</property>

        <!-- Mapping class -->
        <mapping class="me.rachel.model.ModelMahasiswa"/>
    </session-factory>
</hibernate-configuration>
```

Output:



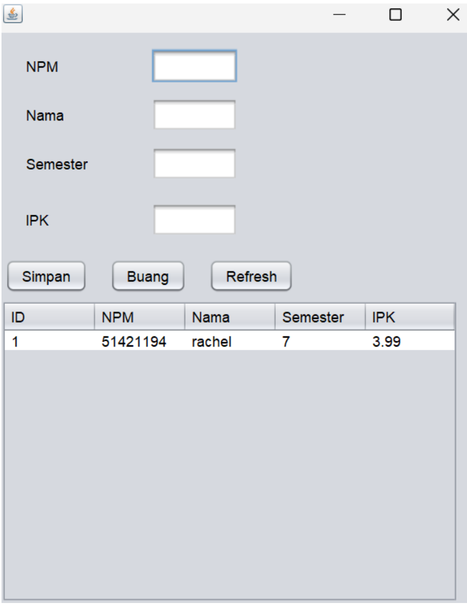
NPM
Nama
Semester
IPK

ID	NPM	Nama	Semester	IPK

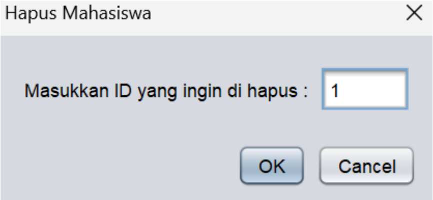


NPM 51421194
Nama rachel
Semester 7
IPK 3.99

ID	NPM	Nama	Semester	IPK



ID	NPM	Nama	Semester	IPK
1	51421194	rachel	7	3.99


Hapus Mahasiswa

Masukkan ID yang ingin di hapus : 1