

ACTIVITY PERTEMUAN 4

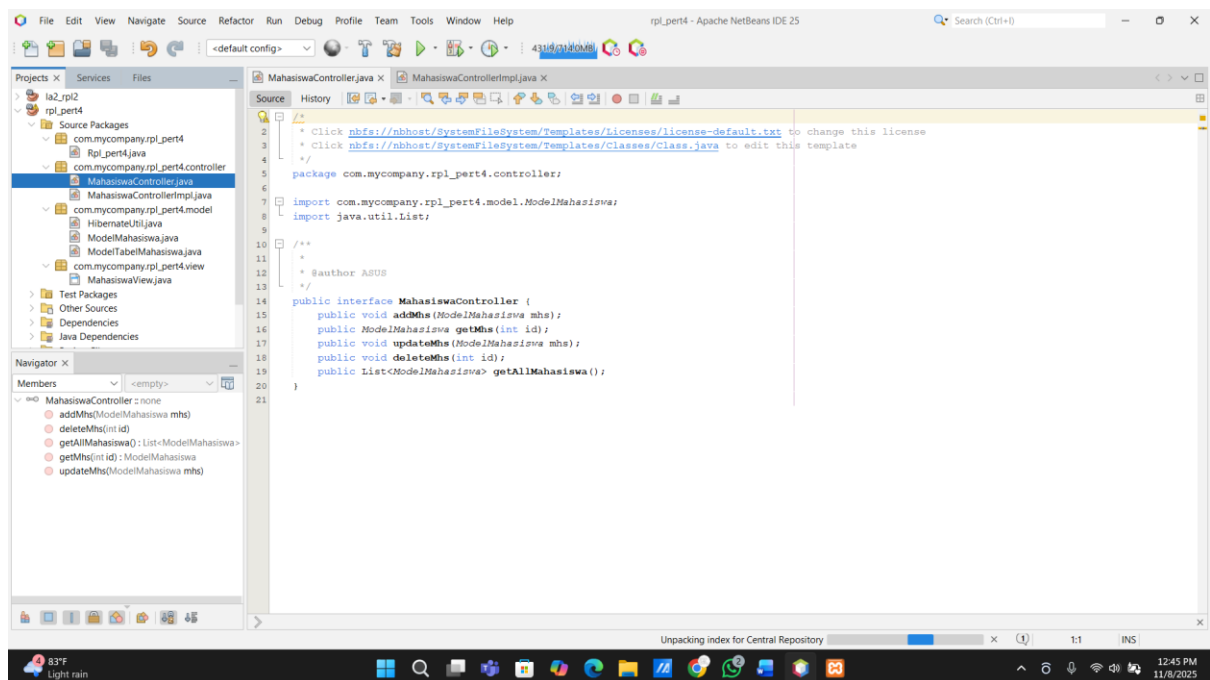
NAMA : Muhammad Rifqi Dwi Putra
NPM : 51422145
KELAS : 4IA08
MATERI : Konsep dasar object relational mapping (ORM) dan framework hibernate
MATA PRAKTIKUM : Rekayasa Perangkat Lunak 2

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

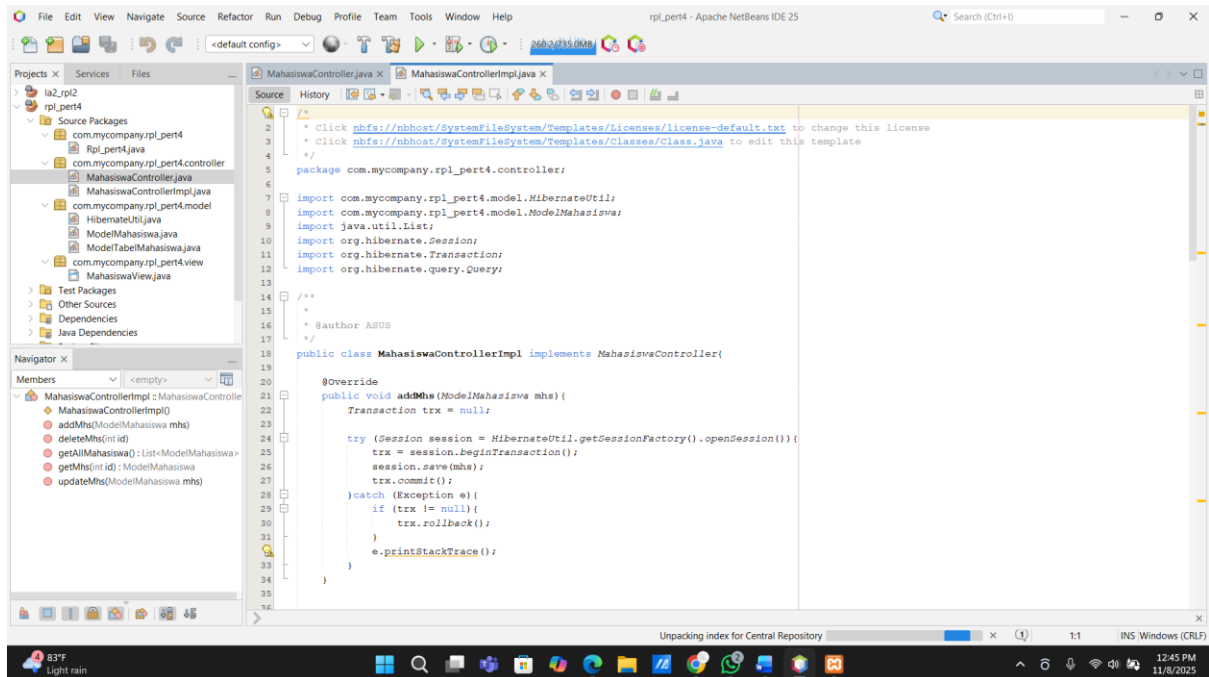
Screenshot kode dan output program

Source Code :

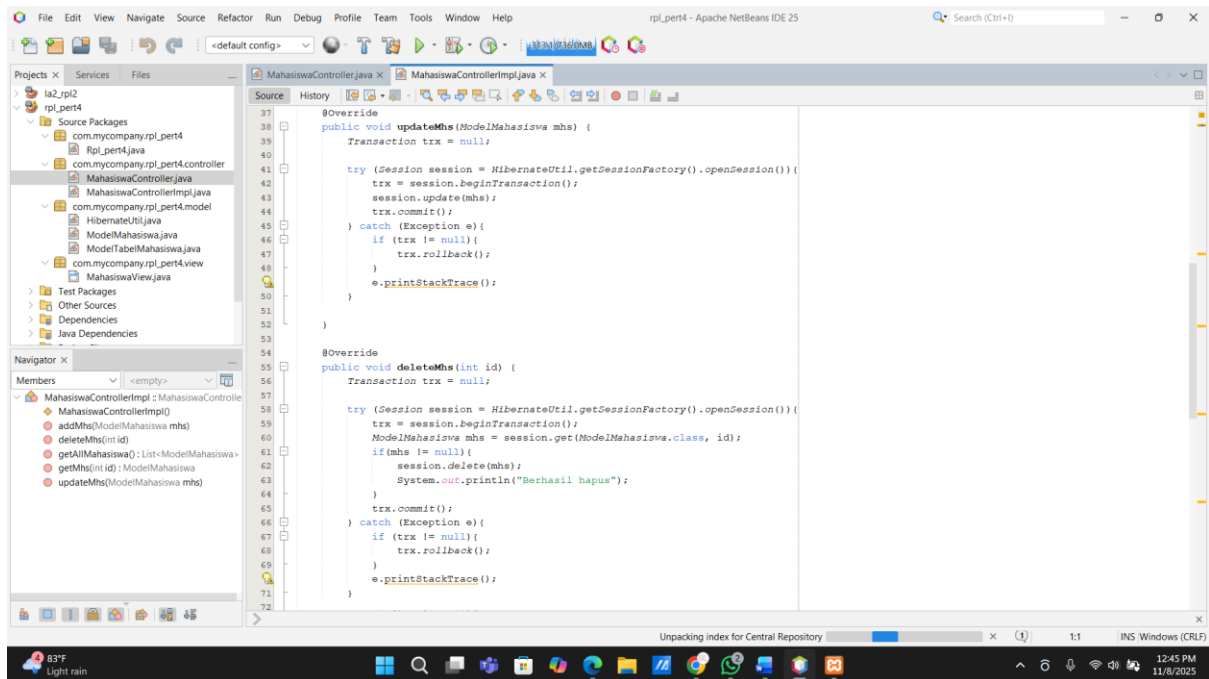
File MahasiswaController.java



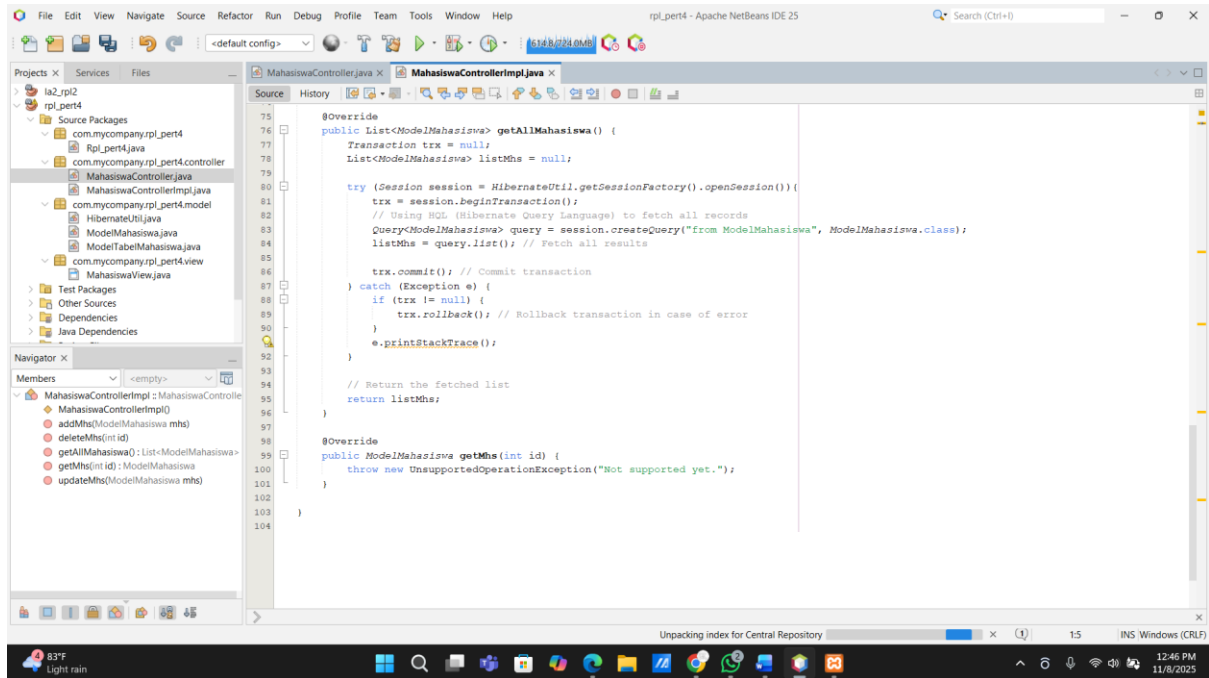
File MahasiswaControllermpl.java



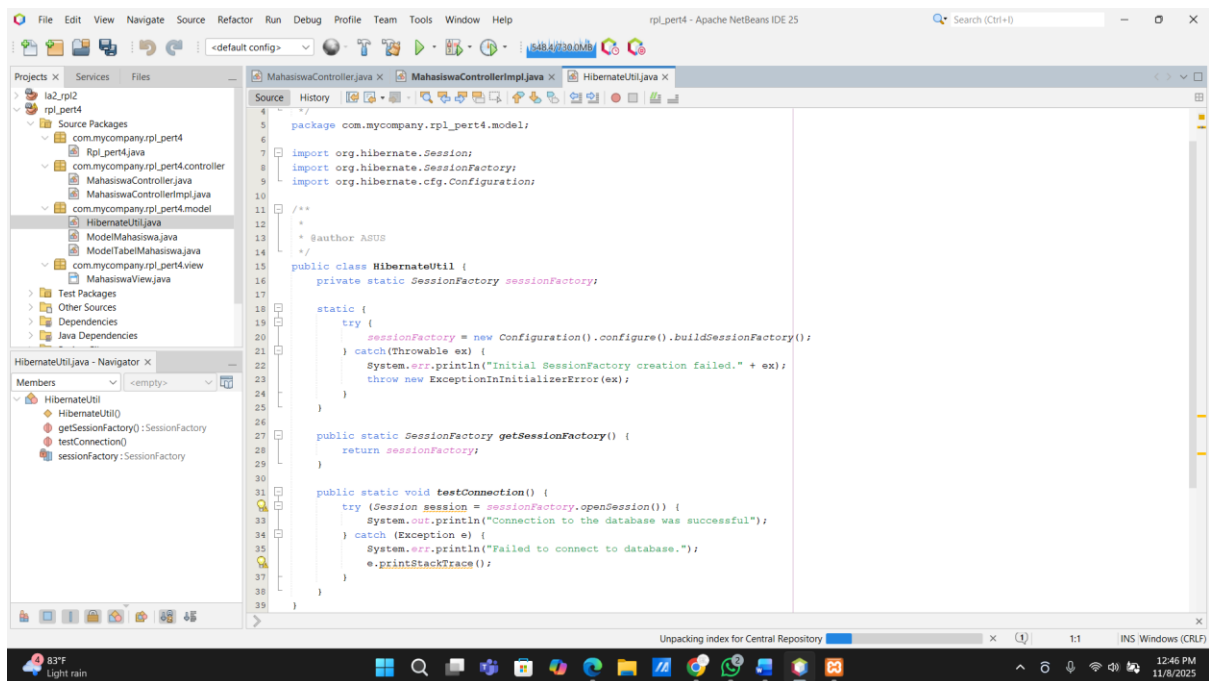
```
1  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
3
4  package com.mycompany.rpl_pert4.controller;
5
6
7  import com.mycompany.rpl_pert4.model.HibernateUtil;
8  import com.mycompany.rpl_pert4.model.ModelMahasiswa;
9  import java.util.List;
10 import org.hibernate.Session;
11 import org.hibernate.Transaction;
12 import org.hibernate.query.Query;
13
14 /**
15  *
16  * @author ASUS
17  */
18 public class MahasiswaControllerImpl implements MahasiswaController{
19
20     @Override
21     public void addMhs(ModelMahasiswa mhs) {
22         Transaction trx = null;
23
24         try {
25             Session session = HibernateUtil.getSessionFactory().openSession();
26             trx = session.beginTransaction();
27             session.save(mhs);
28             trx.commit();
29         } catch (Exception e) {
30             if (trx != null) {
31                 trx.rollback();
32             }
33             e.printStackTrace();
34         }
35     }
36 }
```



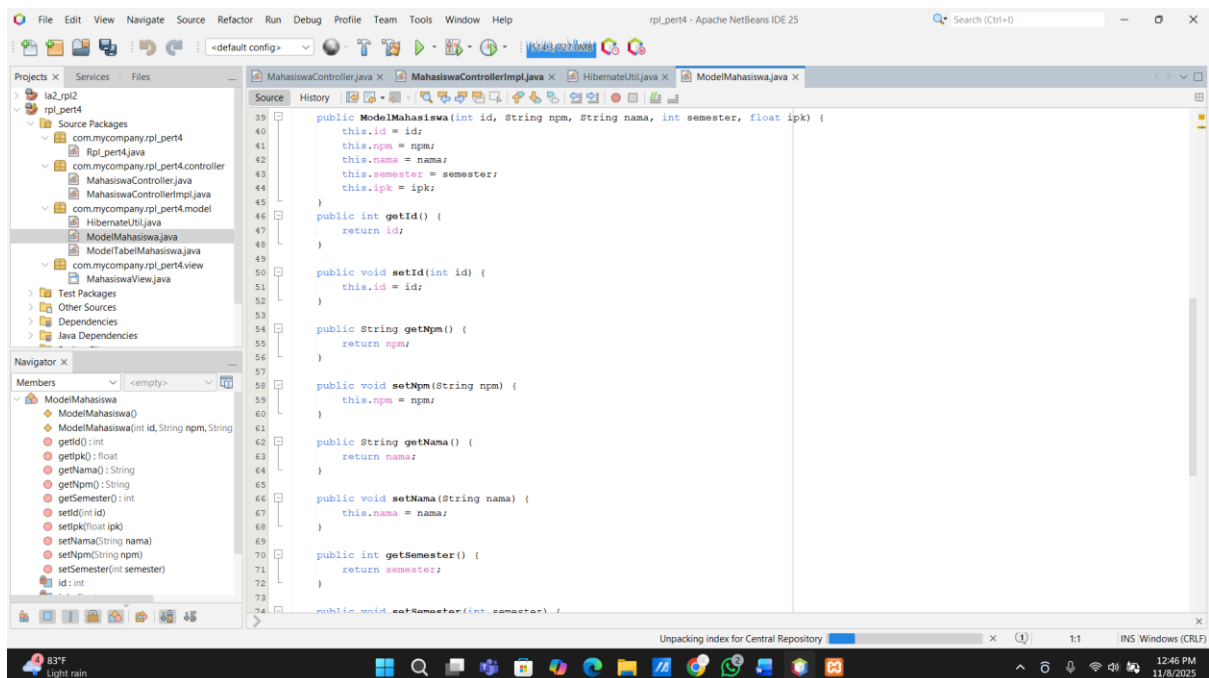
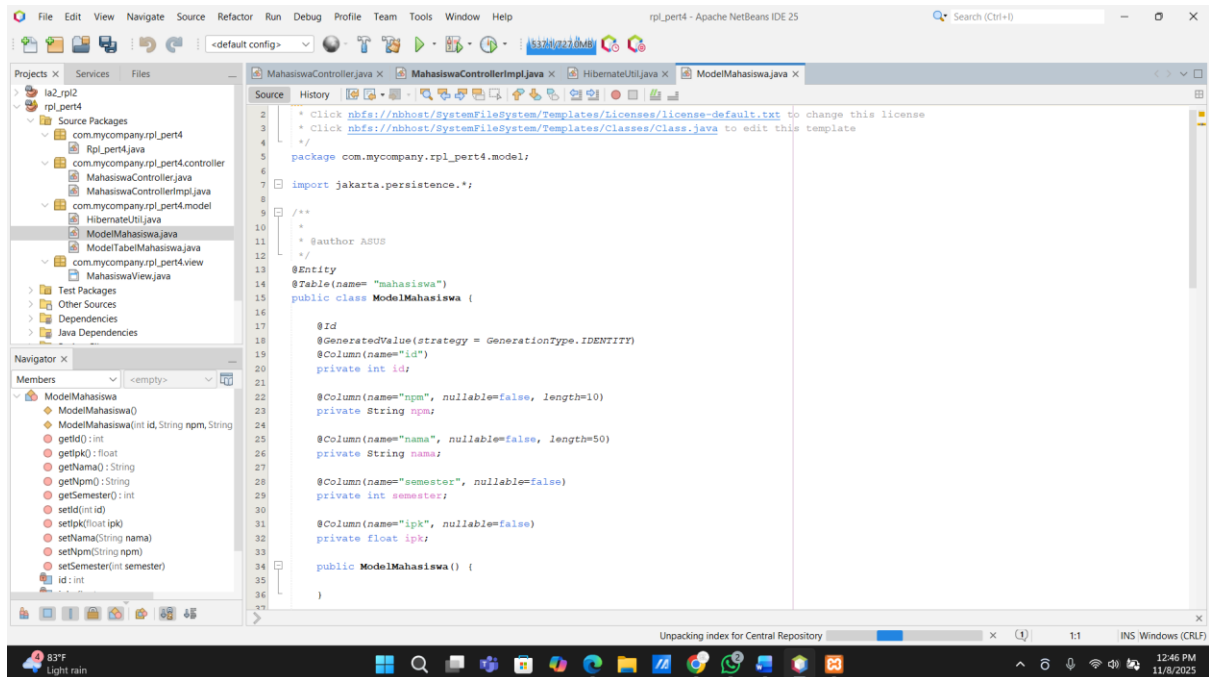
```
37
38 @Override
39 public void updateMhs(ModelMahasiswa mhs) {
40     Transaction trx = null;
41
42     try {
43         Session session = HibernateUtil.getSessionFactory().openSession();
44         trx = session.beginTransaction();
45         session.update(mhs);
46         trx.commit();
47     } catch (Exception e) {
48         if (trx != null) {
49             trx.rollback();
50         }
51         e.printStackTrace();
52     }
53 }
54
55 @Override
56 public void deleteMhs(int id) {
57     Transaction trx = null;
58
59     try {
60         Session session = HibernateUtil.getSessionFactory().openSession();
61         trx = session.beginTransaction();
62         ModelMahasiswa mhs = session.get(ModelMahasiswa.class, id);
63         if (mhs != null) {
64             session.delete(mhs);
65             System.out.println("Berhasil hapus");
66         }
67         trx.commit();
68     } catch (Exception e) {
69         if (trx != null) {
70             trx.rollback();
71         }
72         e.printStackTrace();
73     }
74 }
```

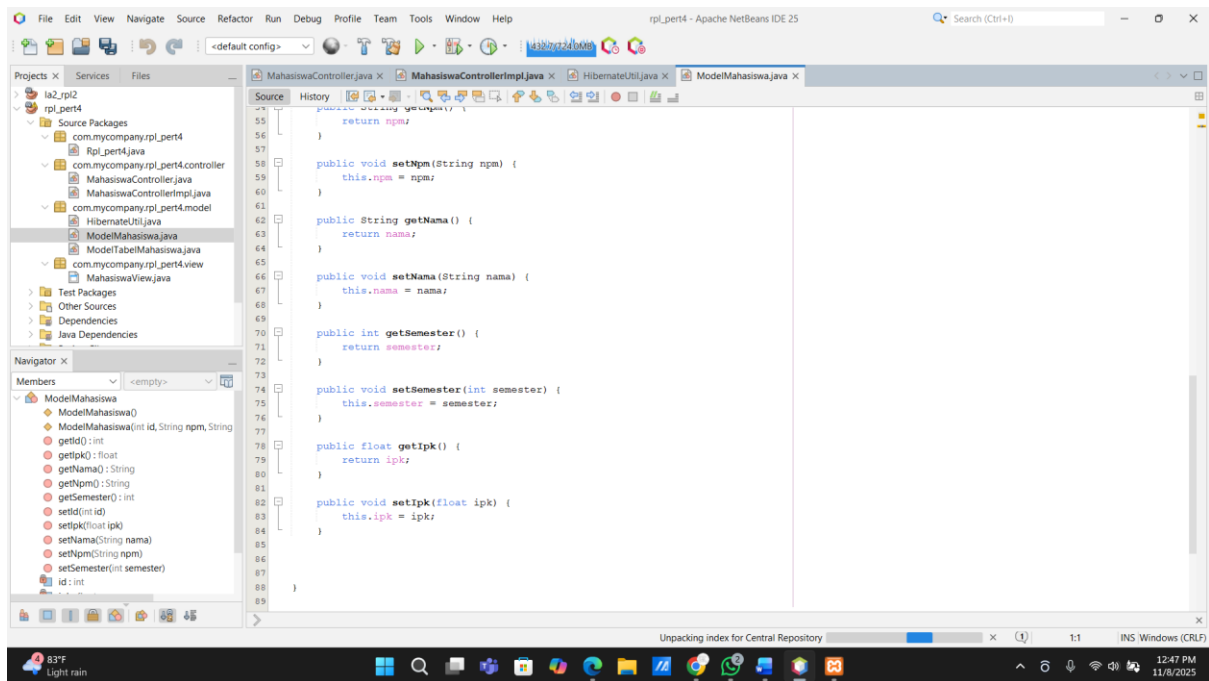


File HibernateUtil.java

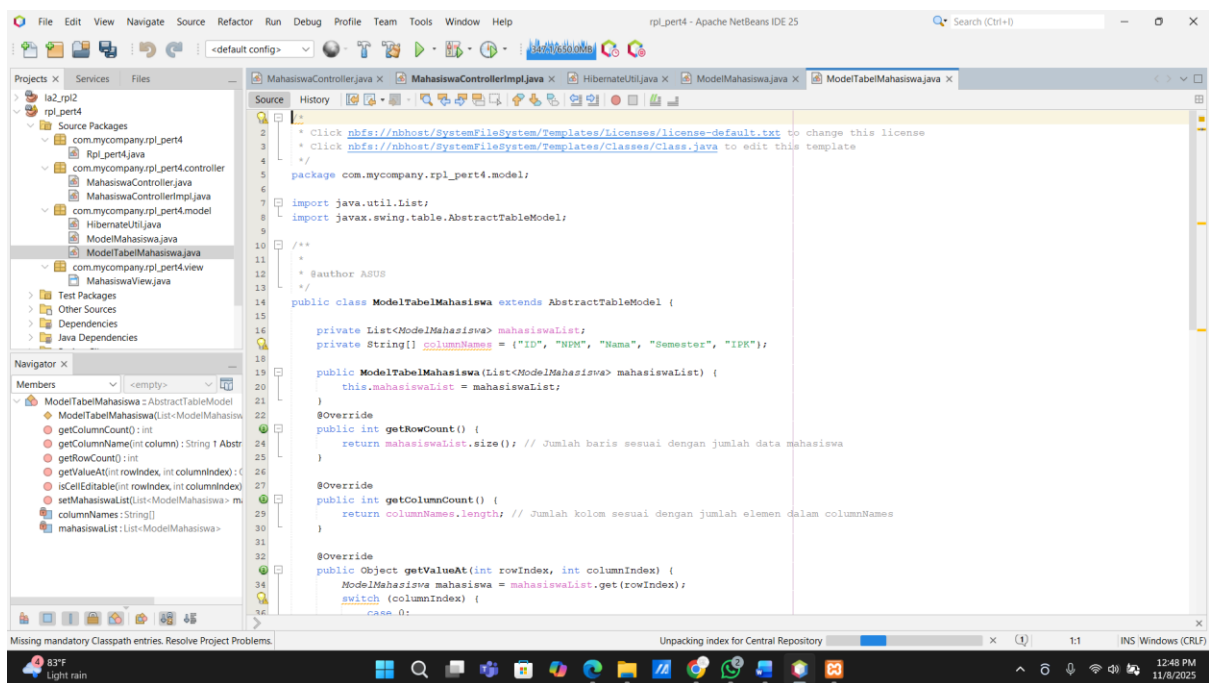


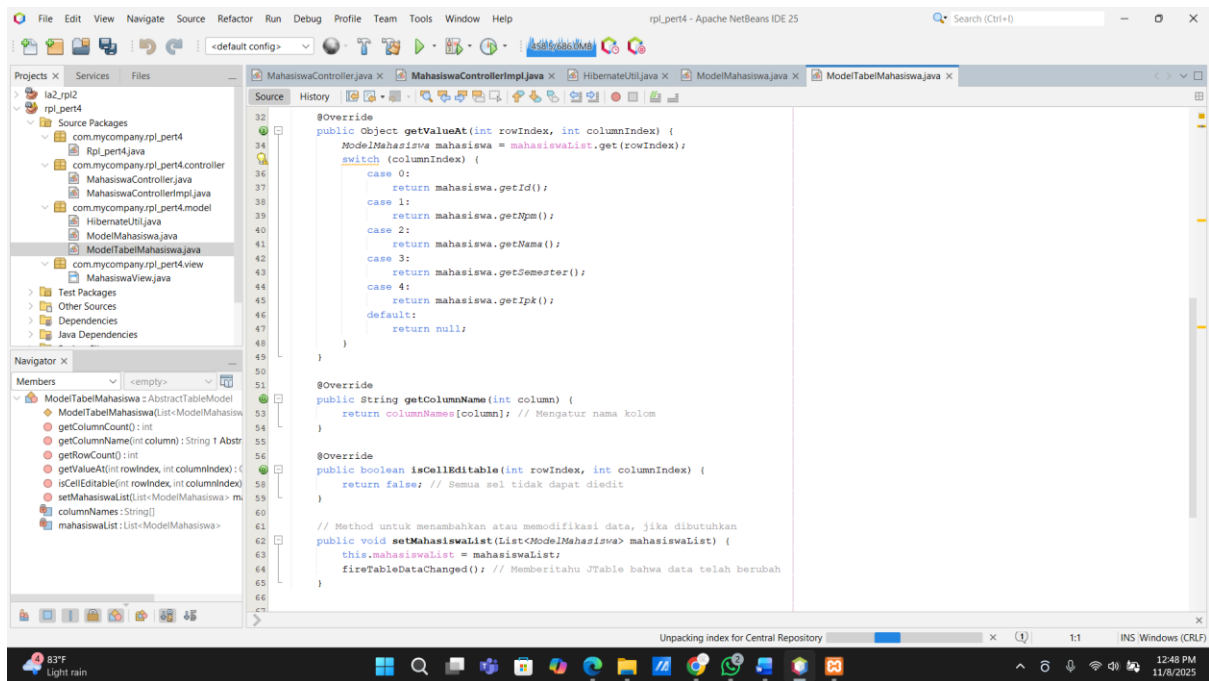
File ModelMahasiswa.java



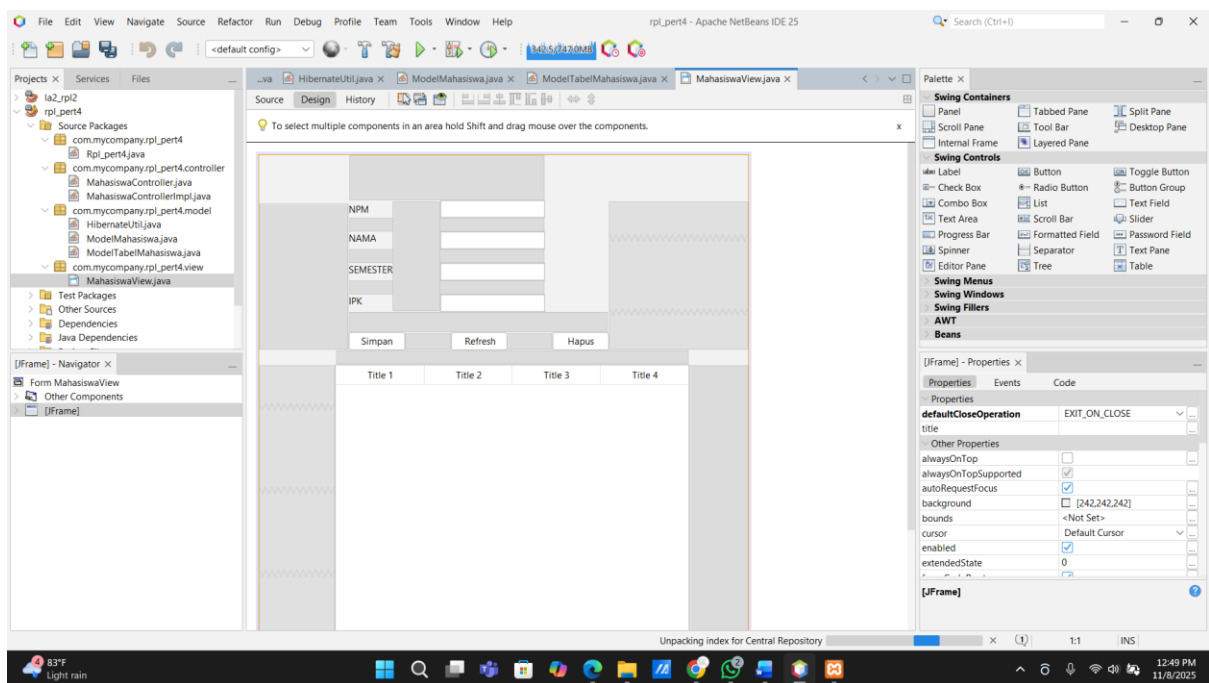


File ModelTabelMahasiswa.java

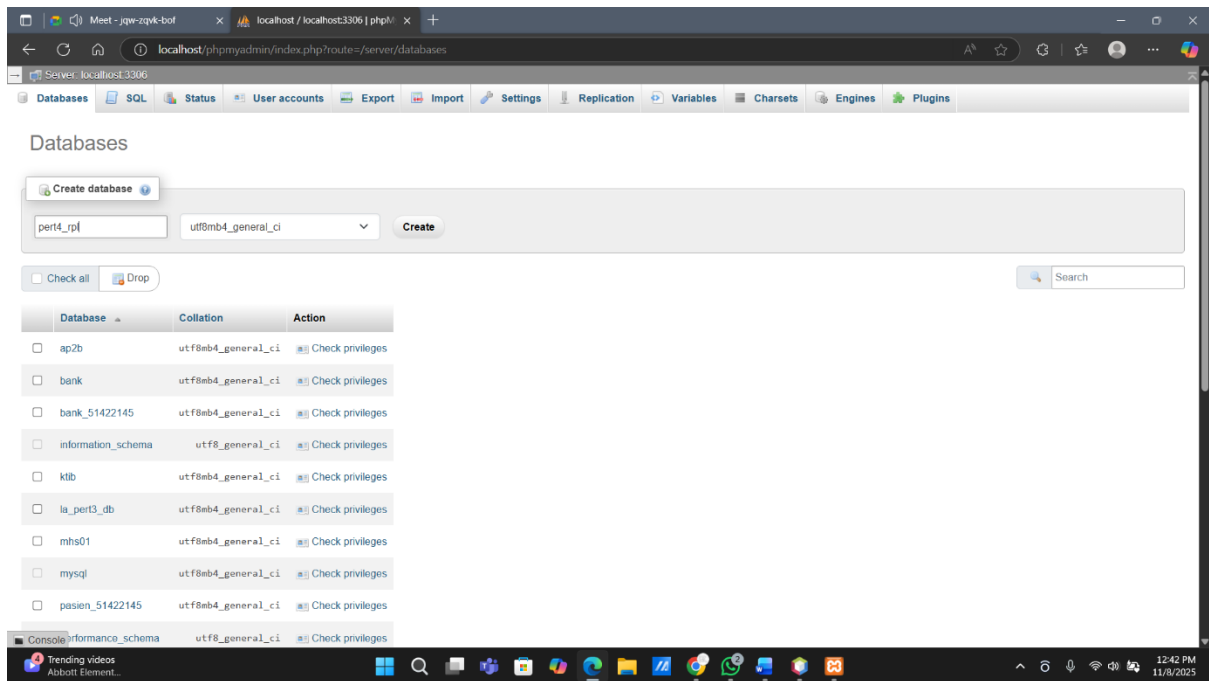




File MahasiswaView.java

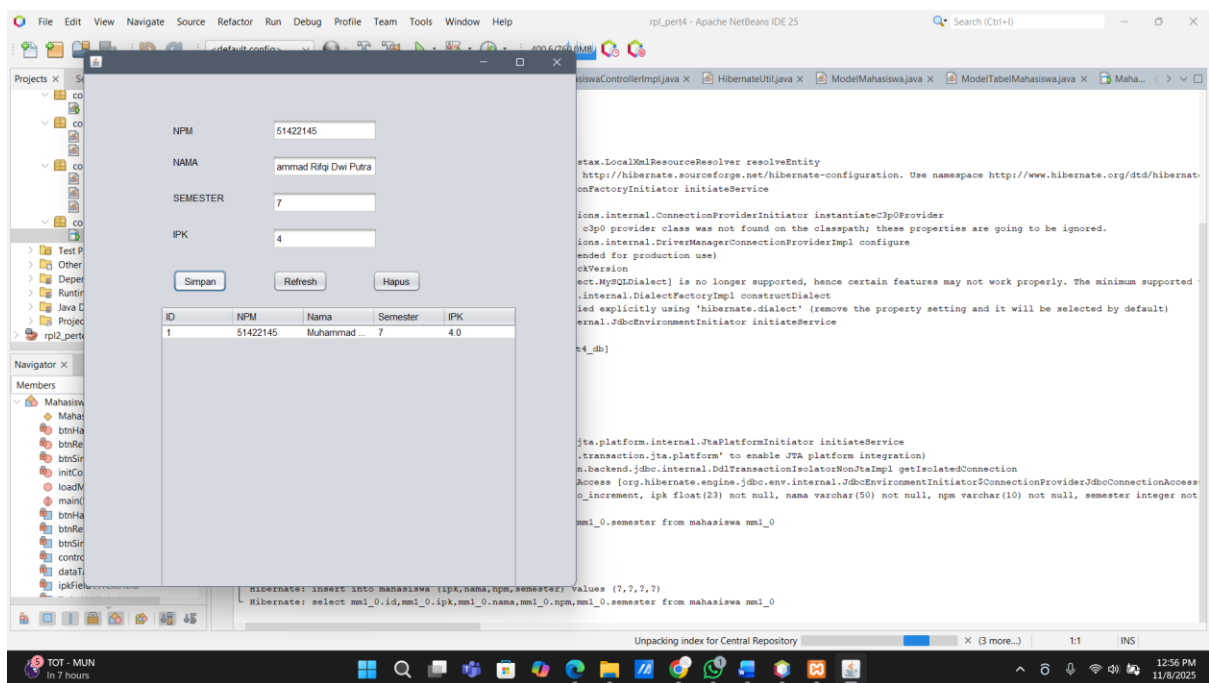


Database

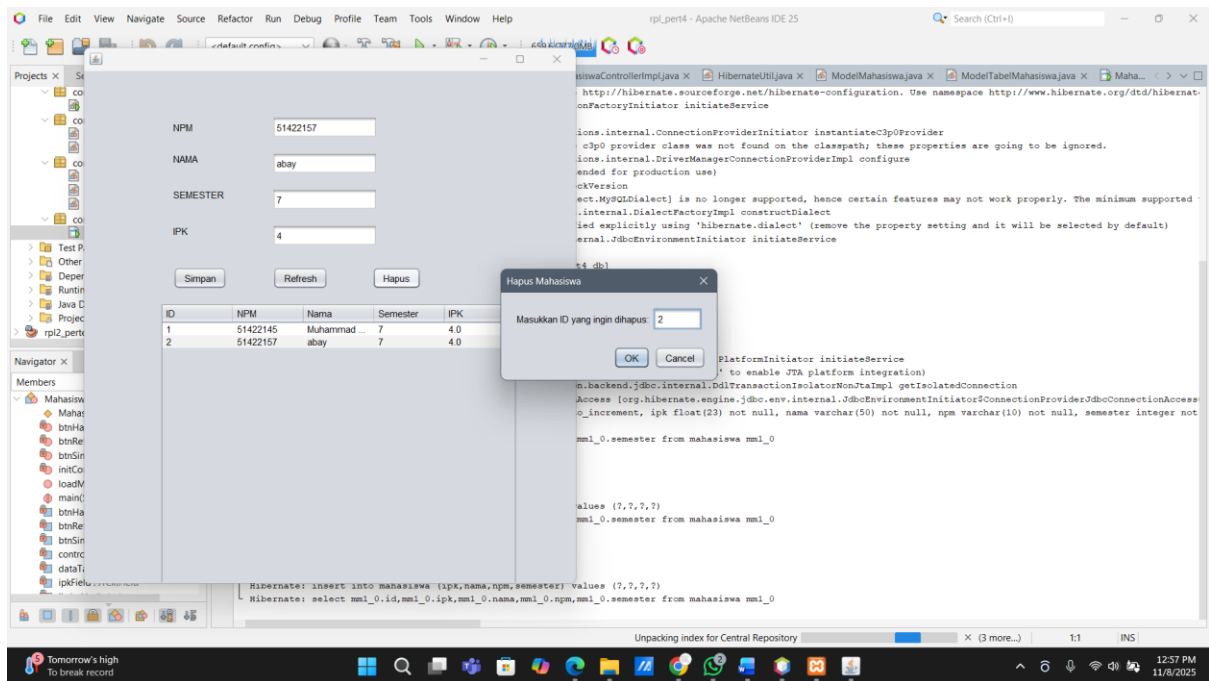


Output :

Tambah/Simpan data



Hapus data



Setelah di refresh

