Object-Relational Mapping

Praktikum Pemrograman II - 08

Wanda Gusdya

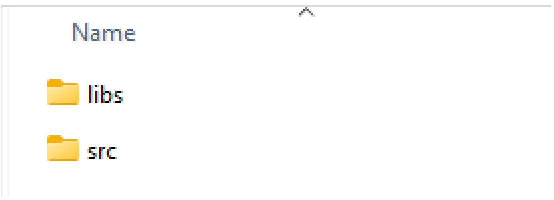
2024

Object Relational Mapping

Dalam Java Standard Edition (Java SE), terdapat beberapa library Object-Relational Mapping (ORM) yang dapat digunakan untuk mengelola interaksi antara aplikasi Java dan database relasional. Salah satu library yang populer adalah MyBatis. MyBatis adalah framework ORM yang lebih fleksibel dibandingkan beberapa ORM lain karena memungkinkan kontrol lebih besar terhadap query SQL. MyBatis sangat cocok jika kita membutuhkan ORM tetapi tetap ingin menulis query SQL secara manual.

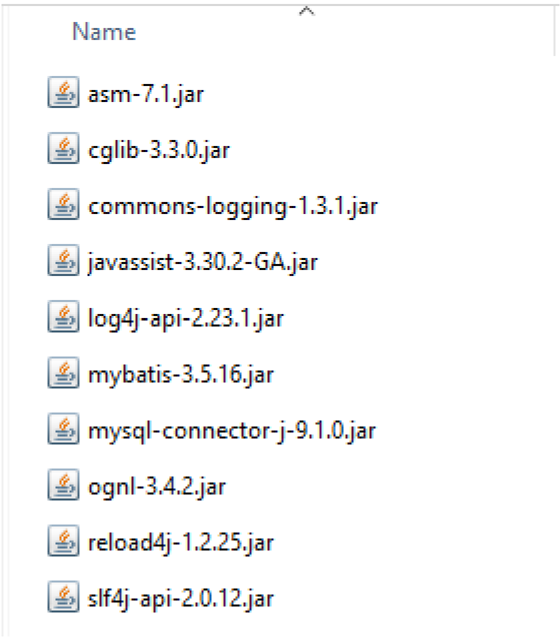
Latihan 1

Kita masih akan menggunakan basisdata pp2_membership yang telah kita buat di modul 7. Pertama-tama buat folder/project baru, misal dengan nama MembershipORM. Kemudian buat 2 folder dengan nama src dan libs. Extract jar yang disediakan ke dalam folder libs, sehingga struktur folder akan terlihat sebagai berikut.



Name
libs
src

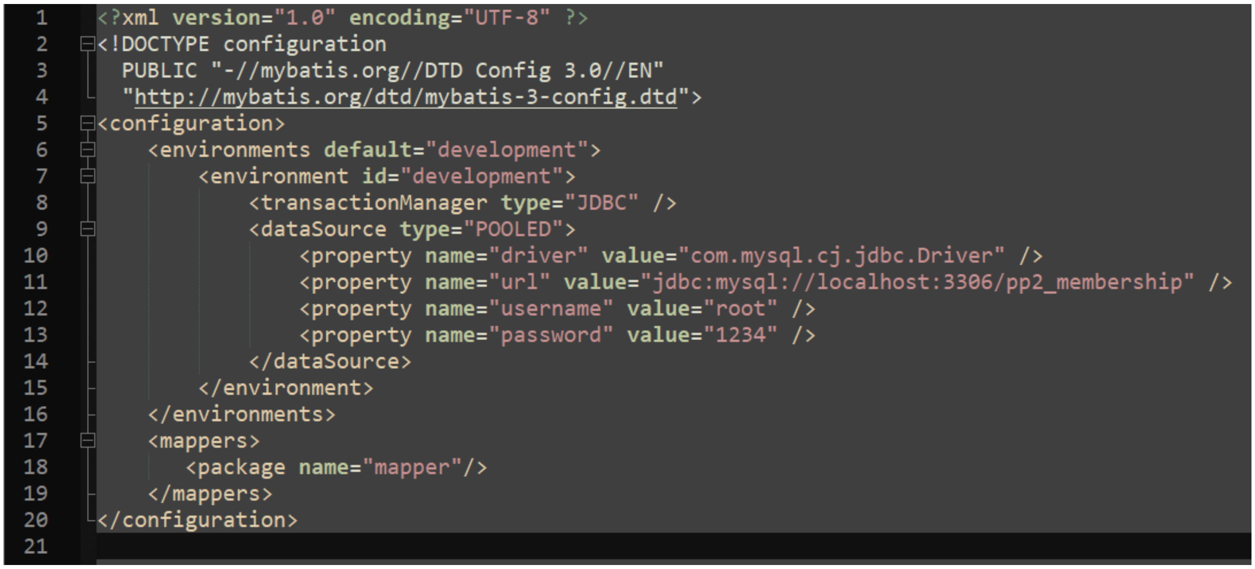
Gambar 1 Struktur Luar Proyek



Name
asm-7.1.jar
cglib-3.3.0.jar
commons-logging-1.3.1.jar
javassist-3.30.2-GA.jar
log4j-api-2.23.1.jar
mybatis-3.5.16.jar
mysql-connector-j-9.1.0.jar
ognl-3.4.2.jar
reload4j-1.2.25.jar
slf4j-api-2.0.12.jar

Gambar 2 Isi Folder Libs

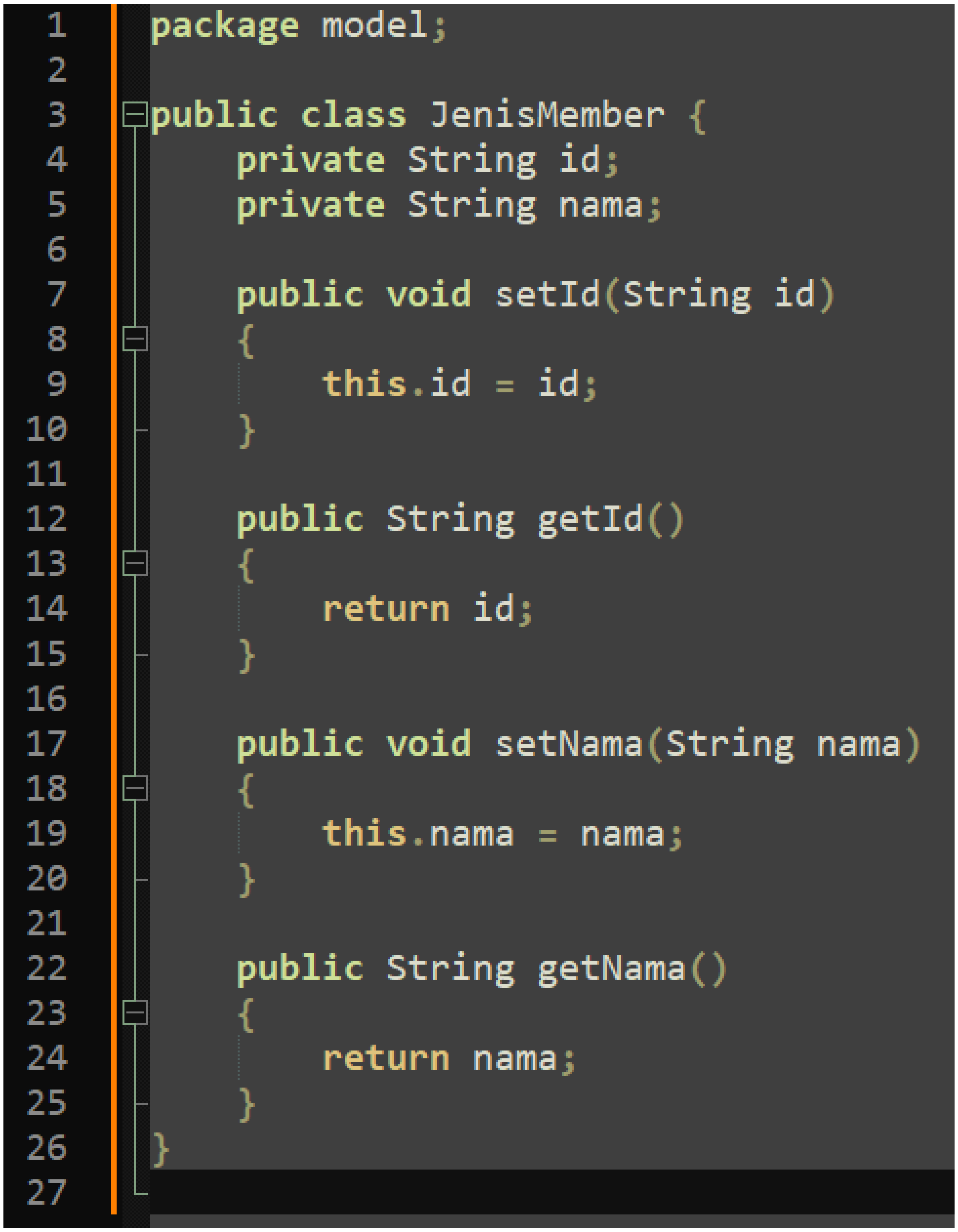
Kemudian, buat file dengan nama mybatis-config.xml dengan isi sebagai berikut.



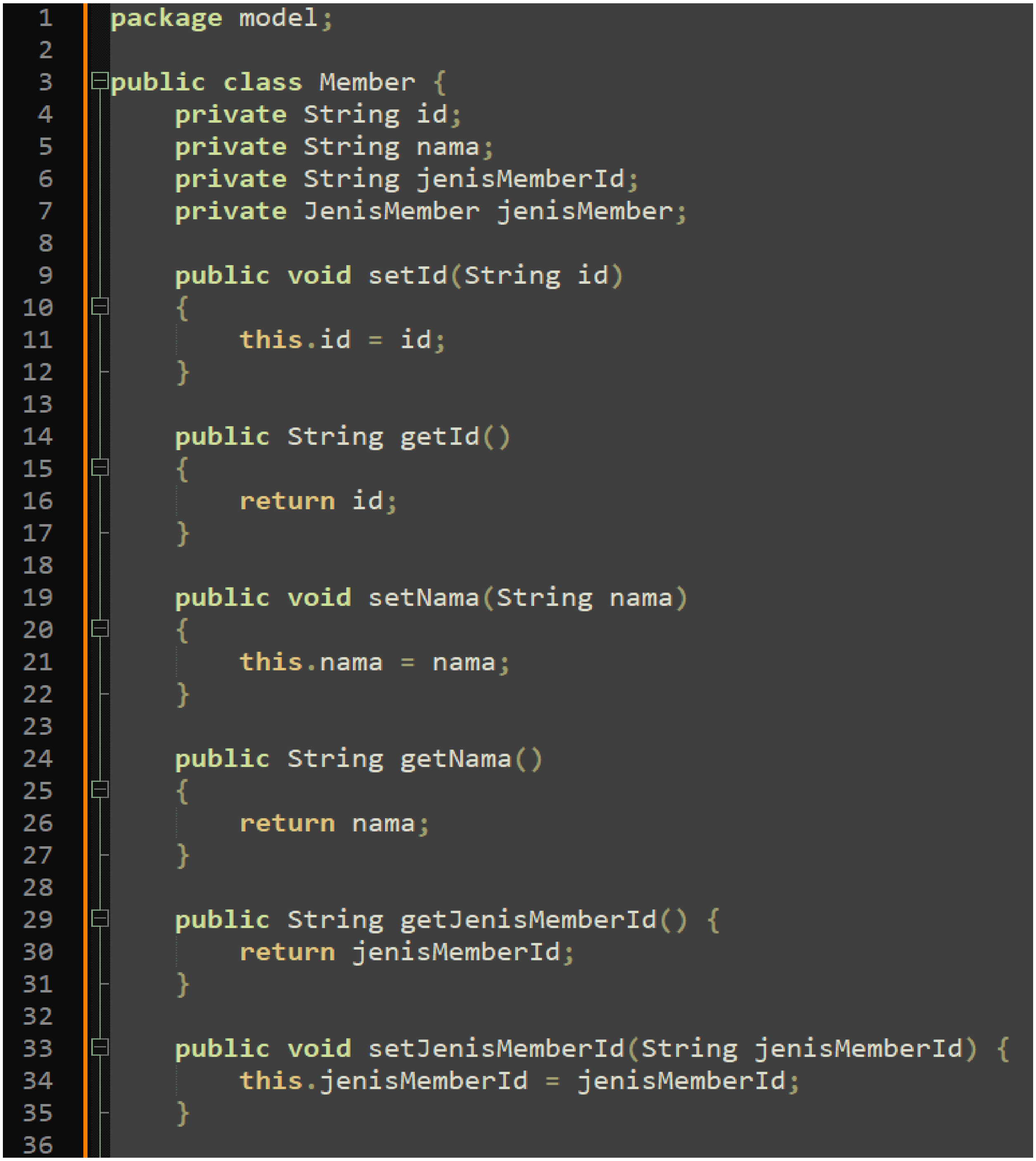
```xml
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE configuration
  PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
  "http://mybatis.org/dtd/mybatis-3-config.dtd">
<configuration>
    <environments default="development">
        <environment id="development">
            <transactionManager type="JDBC" />
            <dataSource type="POOLED">
                <property name="driver" value="com.mysql.cj.jdbc.Driver" />
                <property name="url" value="jdbc:mysql://localhost:3306/pp2_membership" />
                <property name="username" value="root" />
                <property name="password" value="1234" />
            </dataSource>
        </environment>
    </environments>
    <mappers>
       <package name="mapper"/>
    </mappers>
</configuration>

```

Sesuaikan properti dataSource dengan konfigurasi lokal anda.

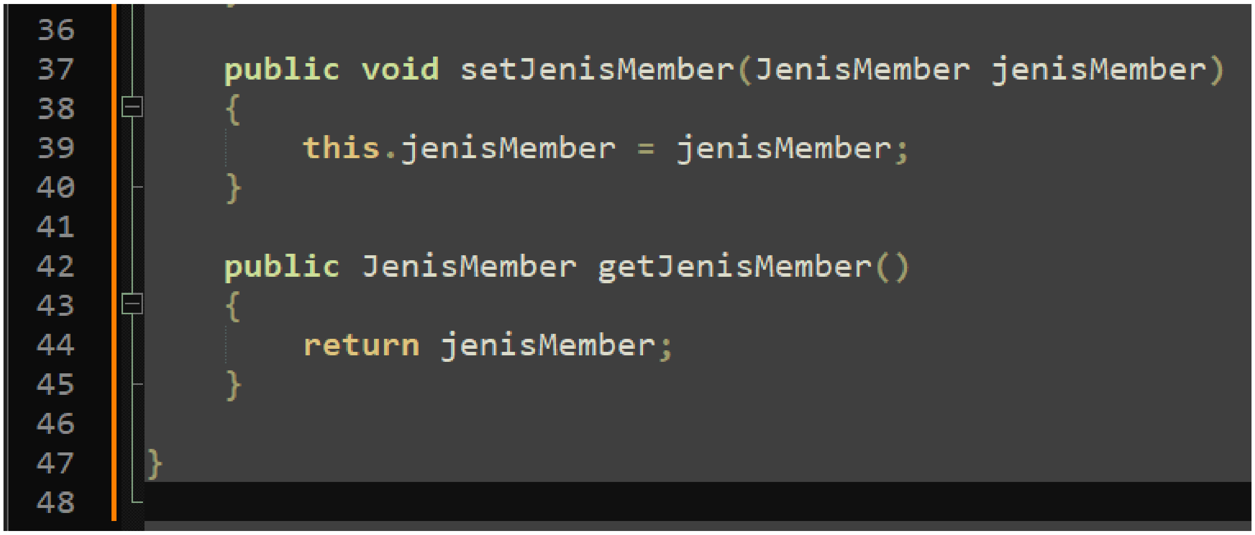
Selanjutnya, buat folder/package model lalu buat dua kelas yaitu Member dan JenisMember dengan kode sebagai berikut.



```java
package model;

public class JenisMember {
    private String id;
    private String nama;

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

    public String getId()
    {
        return id;
    }

    public void setNama(String nama)
    {
        this.nama = nama;
    }

    public String getNama()
    {
        return nama;
    }
}

```

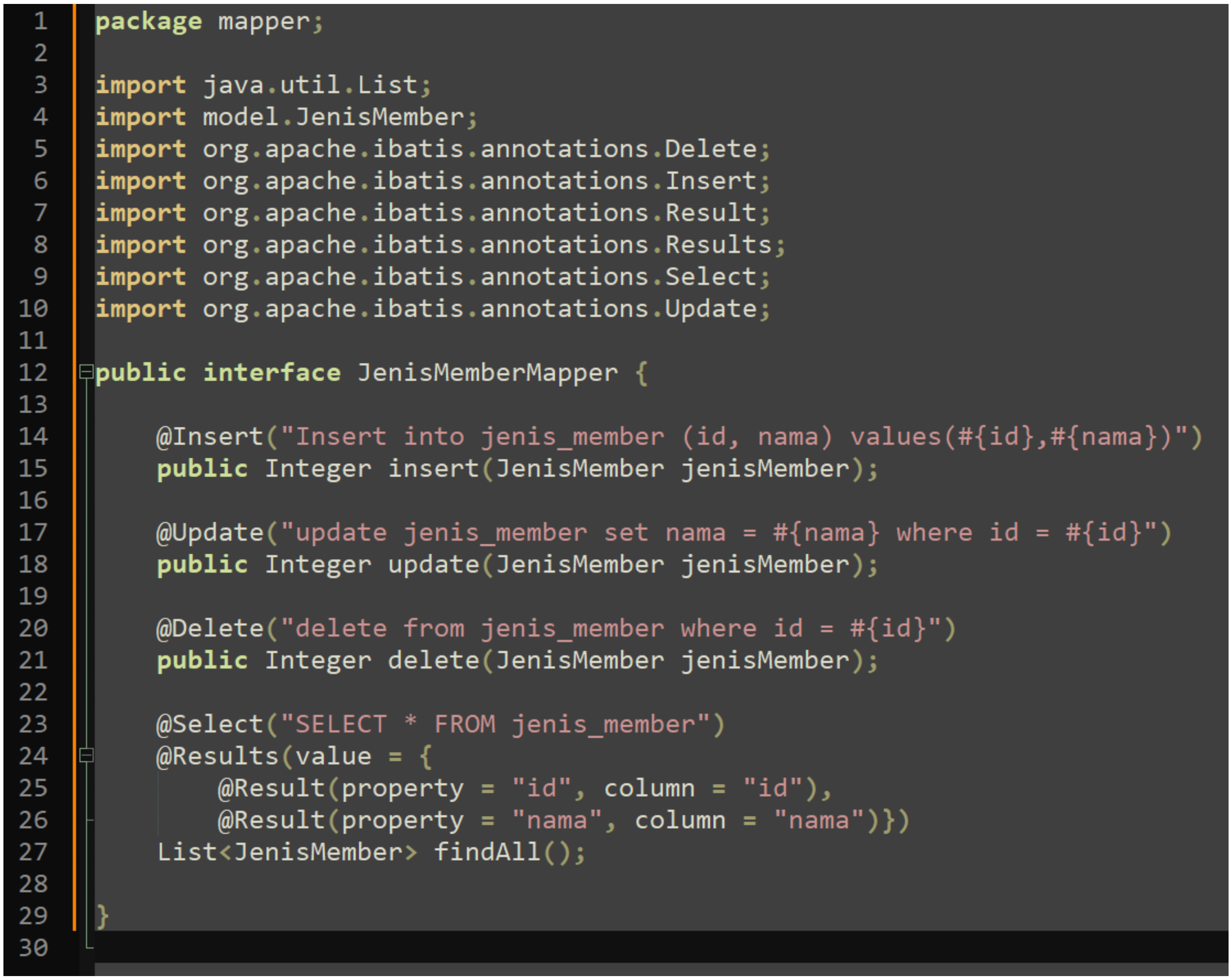


```java
package model;

public class Member {
    private String id;
    private String nama;
    private String jenisMemberId;
    private JenisMember jenisMember;

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

    public String getId()
    {
        return id;
    }

    public void setNama(String nama)
    {
        this.nama = nama;
    }

    public String getNama()
    {
        return nama;
    }

    public String getJenisMemberId() {
        return jenisMemberId;
    }

    public void setJenisMemberId(String jenisMemberId) {
        this.jenisMemberId = jenisMemberId;
    }

```

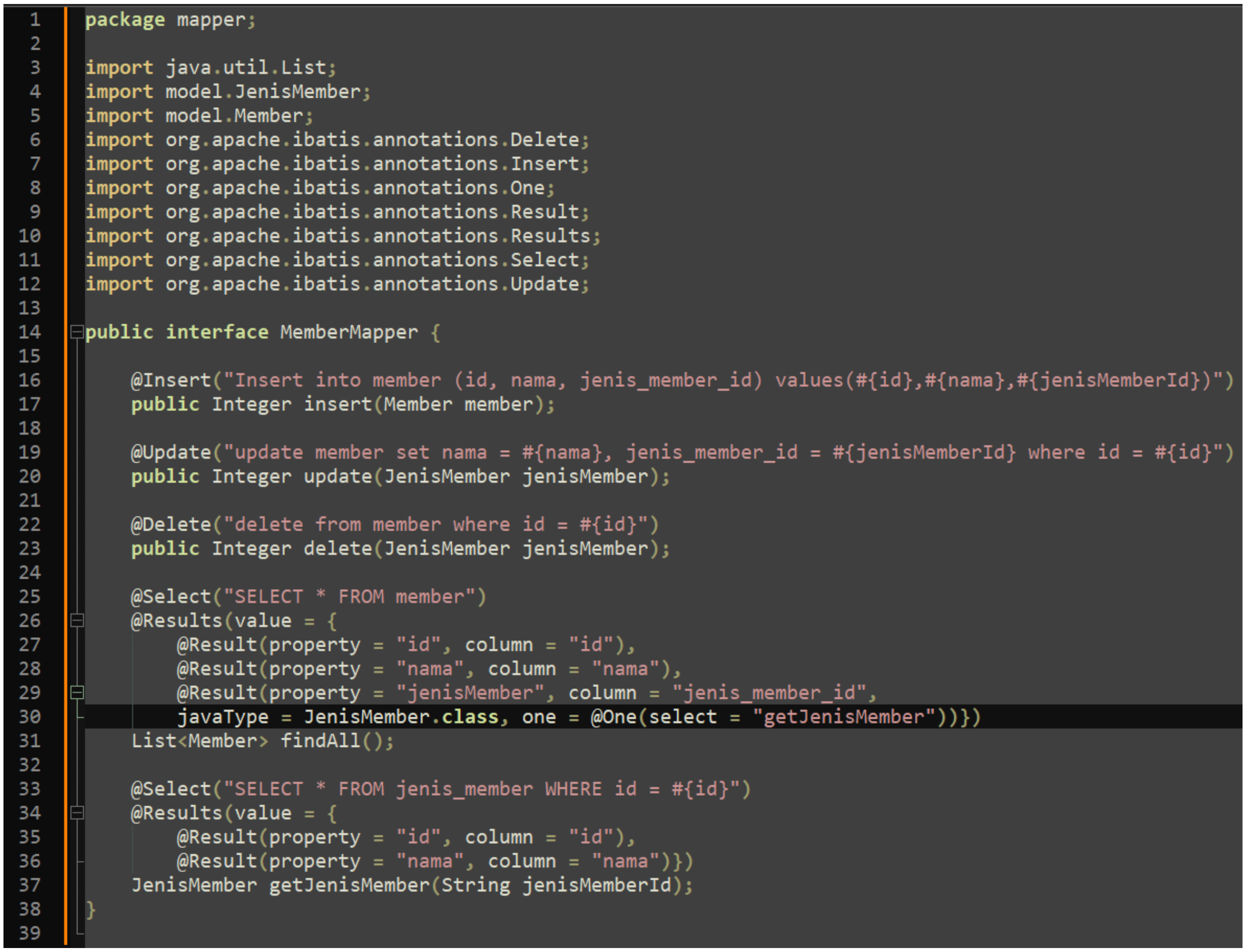


```java

    public void setJenisMember(JenisMember jenisMember)
    {
        this.jenisMember = jenisMember;
    }

    public JenisMember getJenisMember()
    {
        return jenisMember;
    }

}

```

Kemudian, buat package mapper, lalu buat dua kelas dengan nama MemberMapper dan JenisMemberMapper.

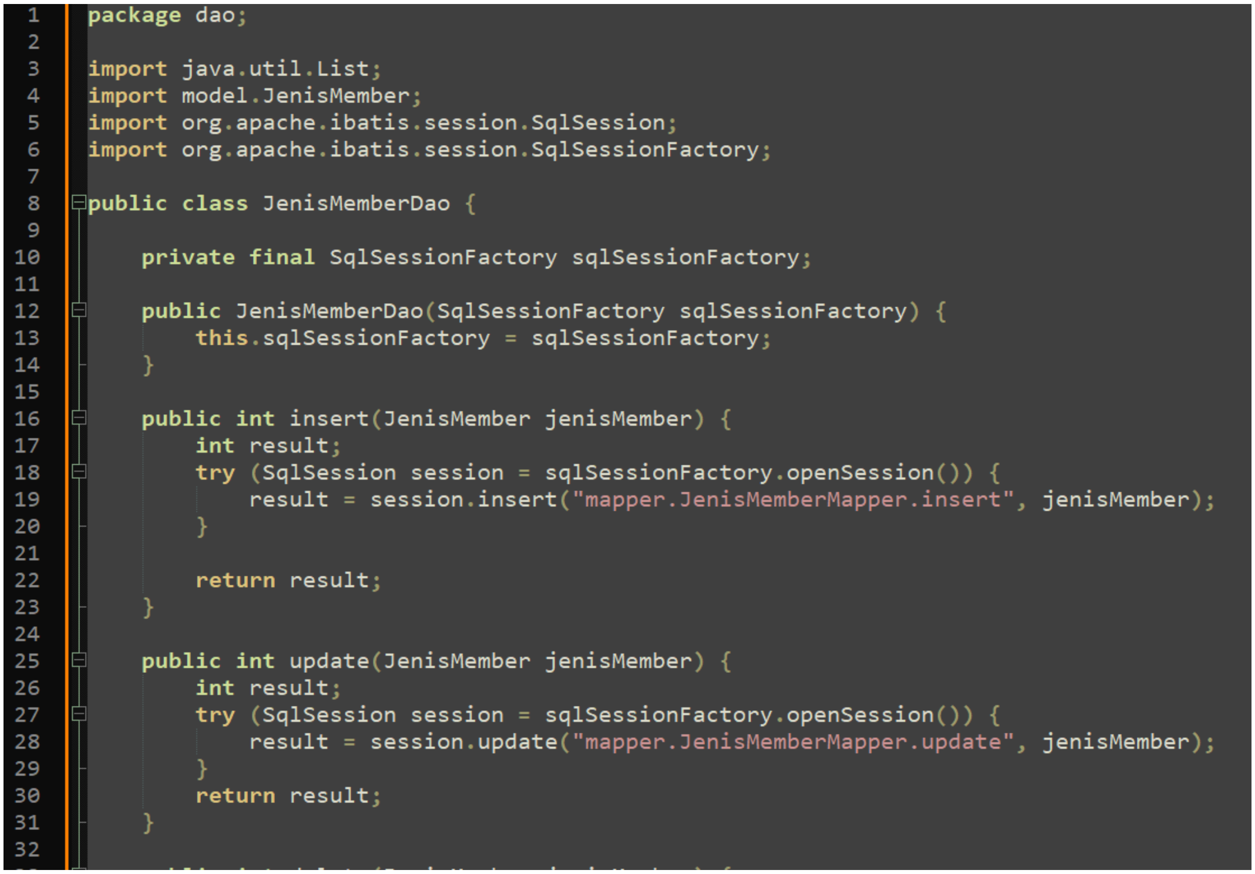


```java
package mapper;

import java.util.List;
import model.JenisMember;
import org.apache.ibatis.annotations.Delete;
import org.apache.ibatis.annotations.Insert;
import org.apache.ibatis.annotations.Result;
import org.apache.ibatis.annotations.Results;
import org.apache.ibatis.annotations.Select;
import org.apache.ibatis.annotations.Update;

public interface JenisMemberMapper {

    @Insert("Insert into jenis_member (id, nama) values(#{id},#{nama})")
    public Integer insert(JenisMember jenisMember);

    @Update("update jenis_member set nama = #{nama} where id = #{id}")
    public Integer update(JenisMember jenisMember);

    @Delete("delete from jenis_member where id = #{id}")
    public Integer delete(JenisMember jenisMember);

    @Select("SELECT * FROM jenis_member")
    @Results(value = {
        @Result(property = "id", column = "id"),
        @Result(property = "nama", column = "nama")})
    List<JenisMember> findAll();

}

```

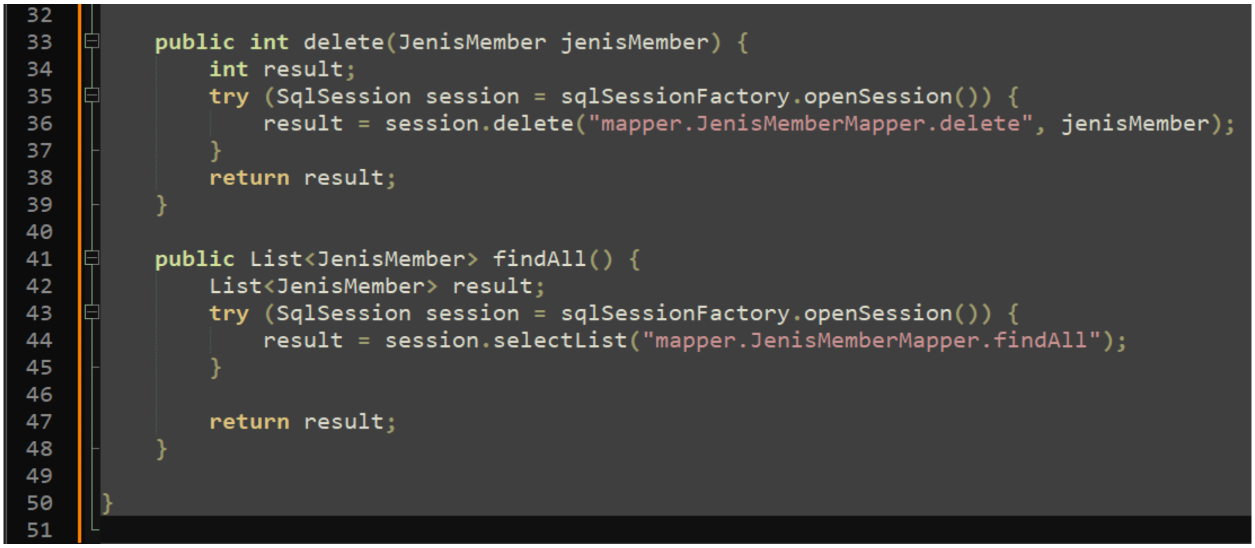


```java
package mapper;

import java.util.List;
import model.JenisMember;
import model.Member;
import org.apache.ibatis.annotations.Delete;
import org.apache.ibatis.annotations.Insert;
import org.apache.ibatis.annotations.One;
import org.apache.ibatis.annotations.Result;
import org.apache.ibatis.annotations.Results;
import org.apache.ibatis.annotations.Select;
import org.apache.ibatis.annotations.Update;

public interface MemberMapper {

    @Insert("Insert into member (id, nama, jenis_member_id) values(#{id},#{nama},#{jenisMemberId})")
    public Integer insert(Member member);

    @Update("update member set nama = #{nama}, jenis_member_id = #{jenisMemberId} where id = #{id}")
    public Integer update(JenisMember jenisMember);

    @Delete("delete from member where id = #{id}")
    public Integer delete(JenisMember jenisMember);

    @Select("SELECT * FROM member")
    @Results(value = {
        @Result(property = "id", column = "id"),
        @Result(property = "nama", column = "nama"),
        @Result(property = "jenisMember", column = "jenis_member_id",
        javaType = JenisMember.class, one = @One(select = "getJenisMember"))})
    List<Member> findAll();

    @Select("SELECT * FROM jenis_member WHERE id = #{id}")
    @Results(value = {
        @Result(property = "id", column = "id"),
        @Result(property = "nama", column = "nama")})
    JenisMember getJenisMember(String jenisMemberId);
}

```

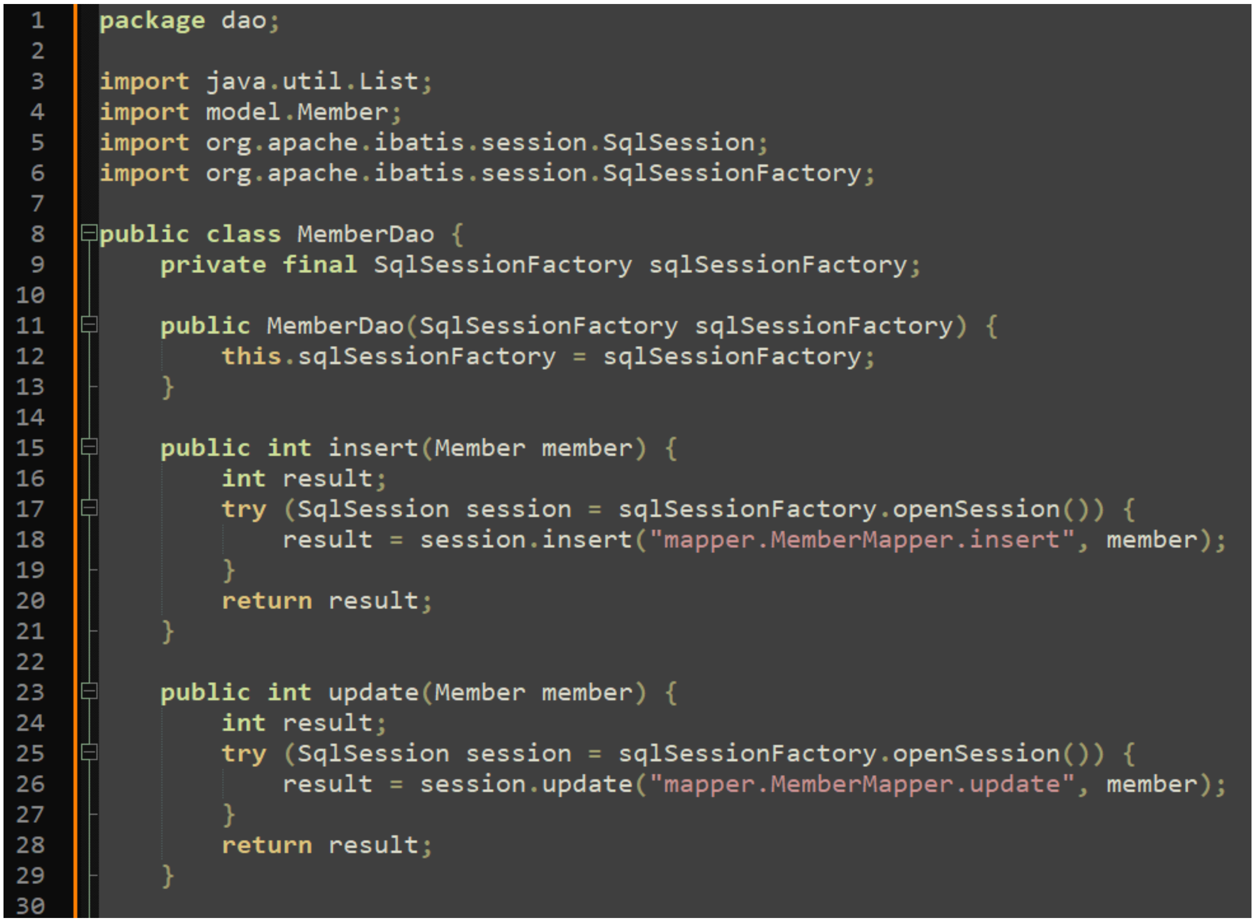
Lalu, buatlah package dao dengan kelas MemberDao dan JenisMemberDao.



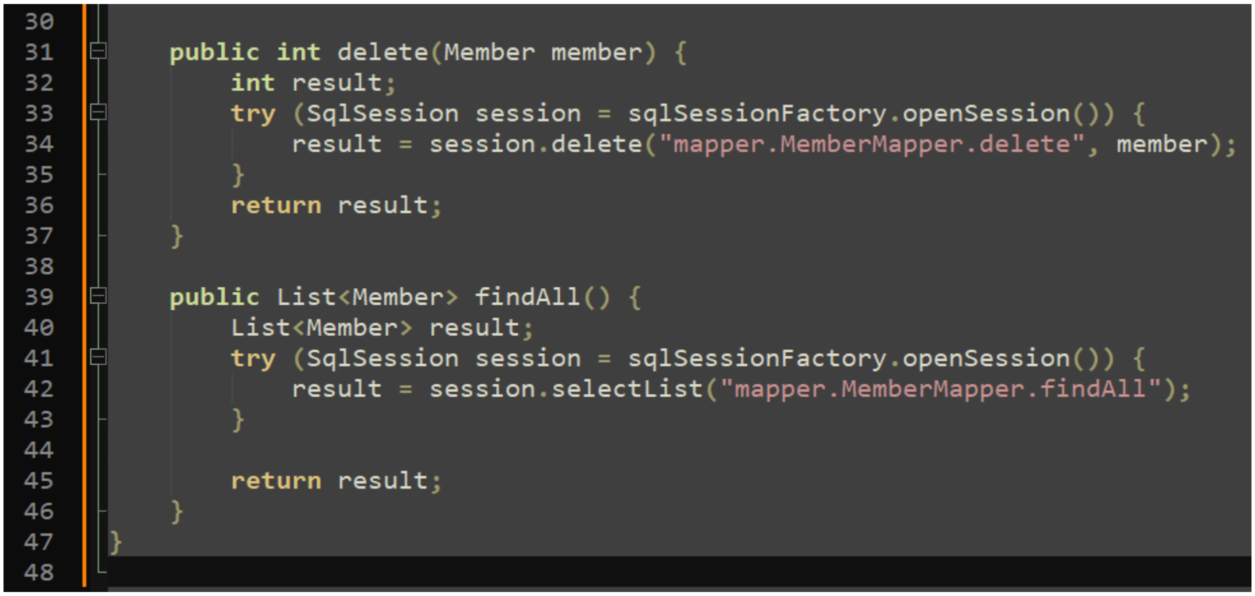
```java
package dao;

import java.util.List;
import model.JenisMember;
import org.apache.ibatis.session.SqlSession;
import org.apache.ibatis.session.SqlSessionFactory;

public class JenisMemberDao {

    private final SqlSessionFactory sqlSessionFactory;

    public JenisMemberDao(SqlSessionFactory sqlSessionFactory) {
        this.sqlSessionFactory = sqlSessionFactory;
    }

    public int insert(JenisMember jenisMember) {
        int result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.insert("mapper.JenisMemberMapper.insert", jenisMember);
        }

        return result;
    }

    public int update(JenisMember jenisMember) {
        int result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.update("mapper.JenisMemberMapper.update", jenisMember);
        }
        return result;
    }

```



```java

    public int delete(JenisMember jenisMember) {
        int result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.delete("mapper.JenisMemberMapper.delete", jenisMember);
        }
        return result;
    }

    public List<JenisMember> findAll() {
        List<JenisMember> result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.selectList("mapper.JenisMemberMapper.findAll");
        }

        return result;
    }

}

```

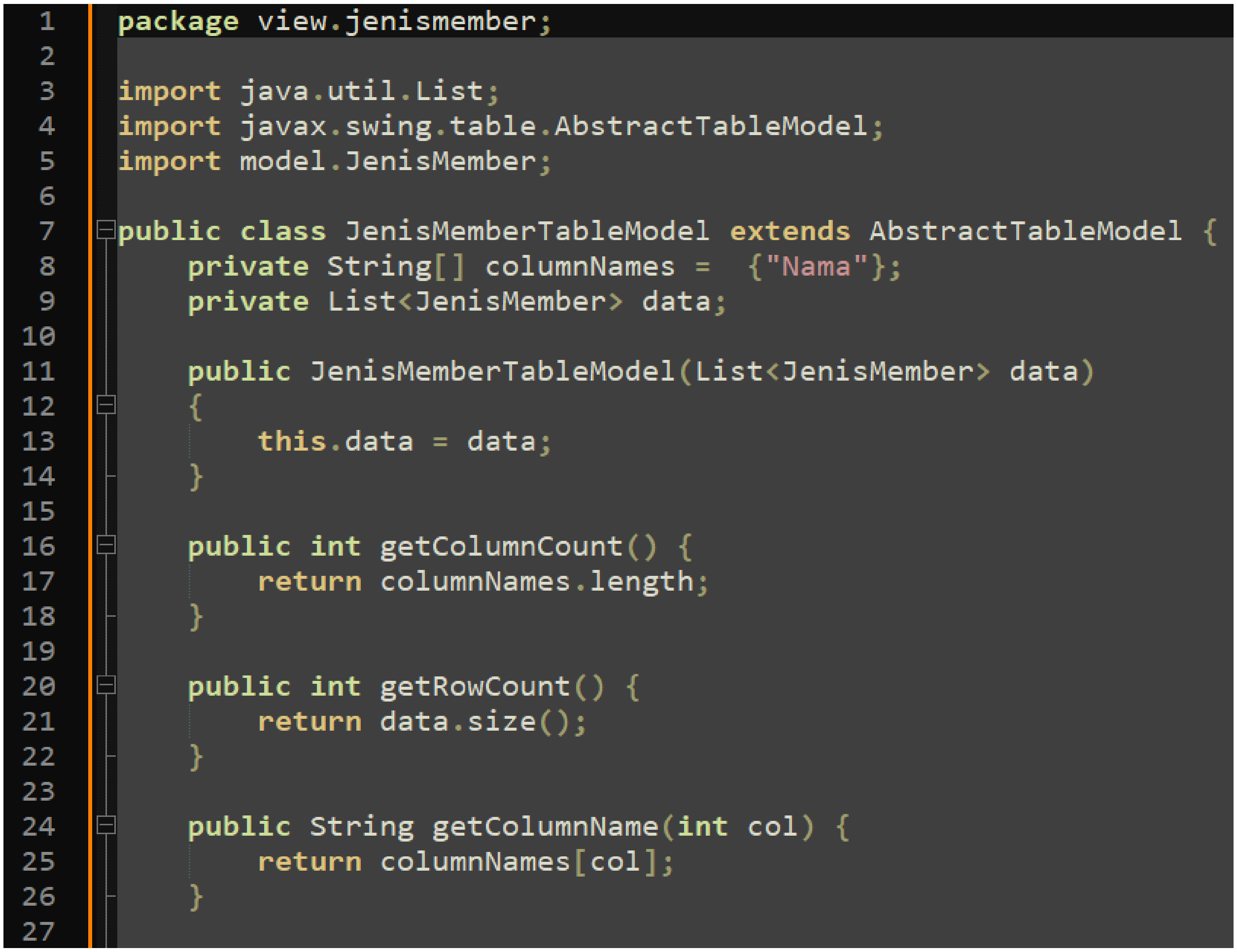


```java
package dao;

import java.util.List;
import model.Member;
import org.apache.ibatis.session.SqlSession;
import org.apache.ibatis.session.SqlSessionFactory;

public class MemberDao {
    private final SqlSessionFactory sqlSessionFactory;

    public MemberDao(SqlSessionFactory sqlSessionFactory) {
        this.sqlSessionFactory = sqlSessionFactory;
    }

    public int insert(Member member) {
        int result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.insert("mapper.MemberMapper.insert", member);
        }
        return result;
    }

    public int update(Member member) {
        int result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.update("mapper.MemberMapper.update", member);
        }
        return result;
    }

```

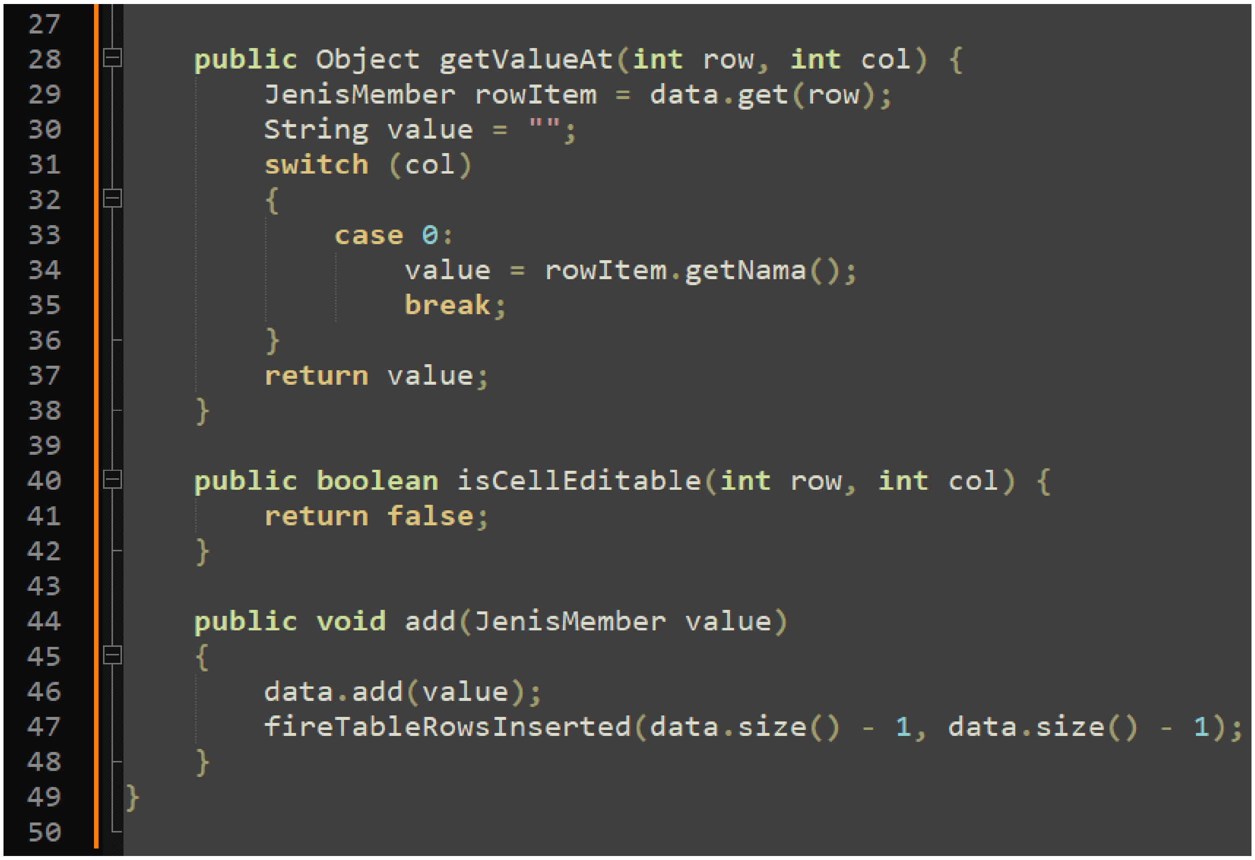


```java

    public int delete(Member member) {
        int result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.delete("mapper.MemberMapper.delete", member);
        }
        return result;
    }

    public List<Member> findAll() {
        List<Member> result;
        try (SqlSession session = sqlSessionFactory.openSession()) {
            result = session.selectList("mapper.MemberMapper.findAll");
        }

        return result;
    }
}

```

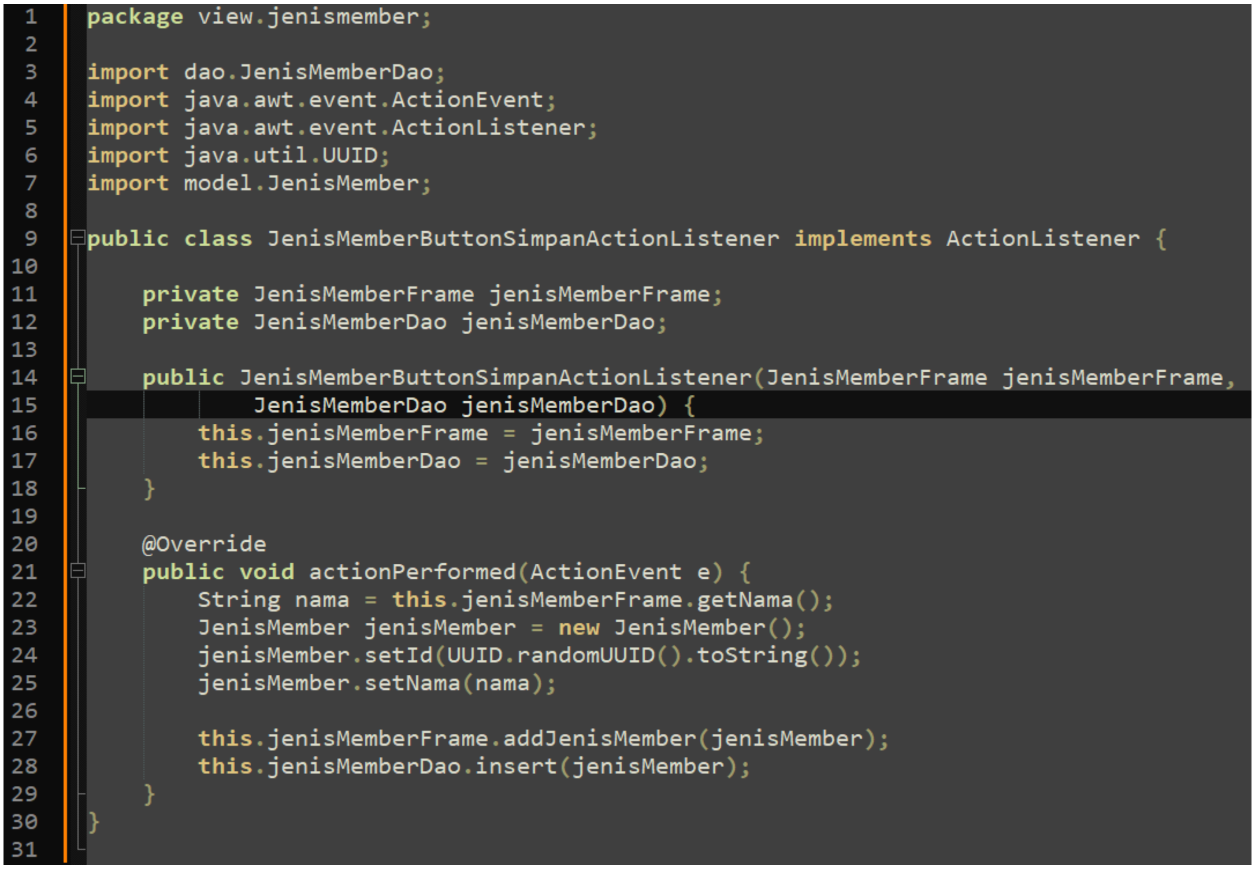
Kemudian, buat package view.jenismember lalu buat kelas JenisMemberTableModel, JenisMemberButtonSimpanActionListener, dan JenisMemberFrame.



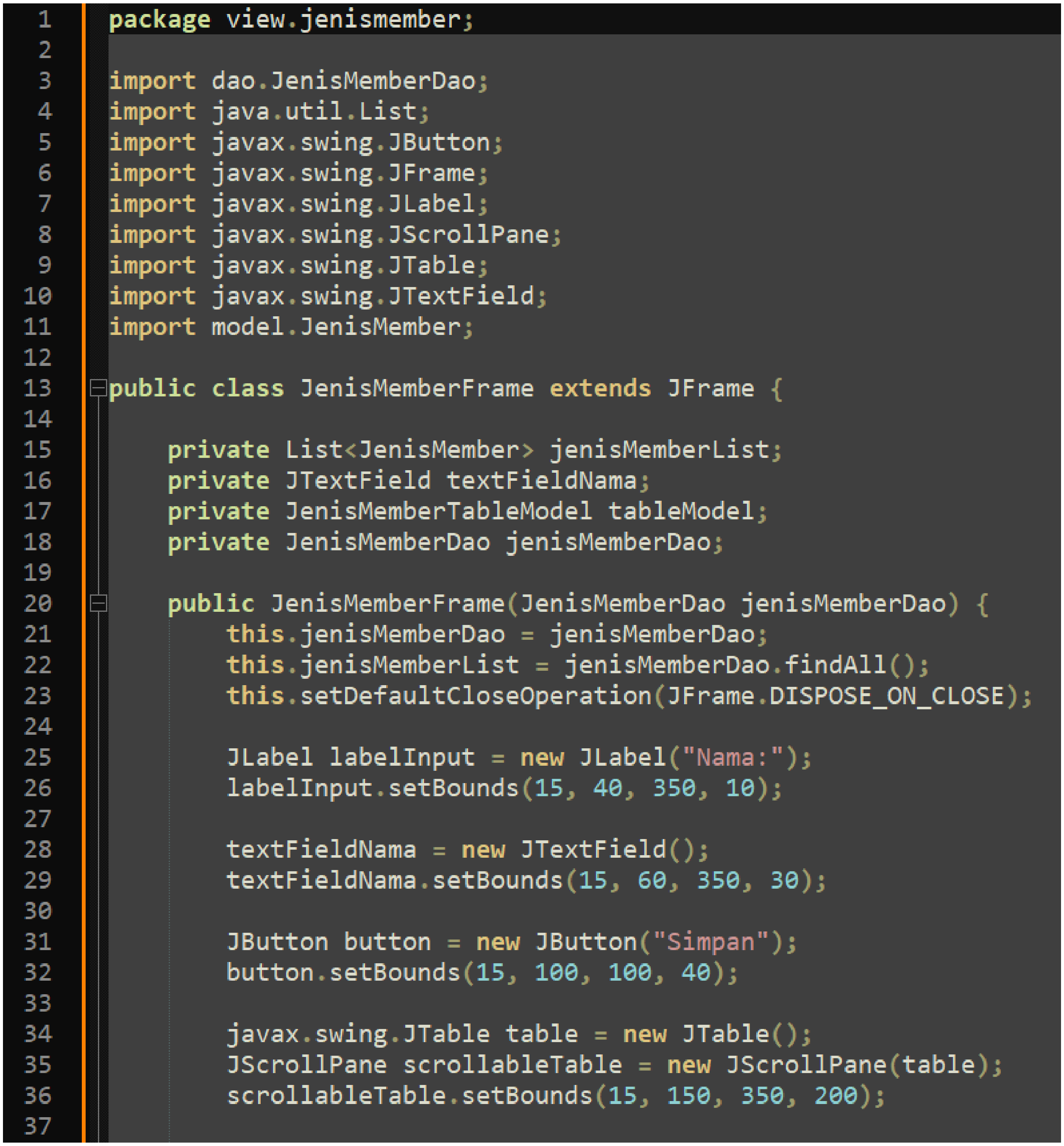
```java
package view.jenismember;

import java.util.List;
import javax.swing.table.AbstractTableModel;
import model.JenisMember;

public class JenisMemberTableModel extends AbstractTableModel {
    private String[] columnNames =  {"Nama"};
    private List<JenisMember> data;

    public JenisMemberTableModel(List<JenisMember> data)
    {
        this.data = data;
    }

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

    public int getRowCount() {
        return data.size();
    }

    public String getColumnName(int col) {
        return columnNames[col];
    }

```



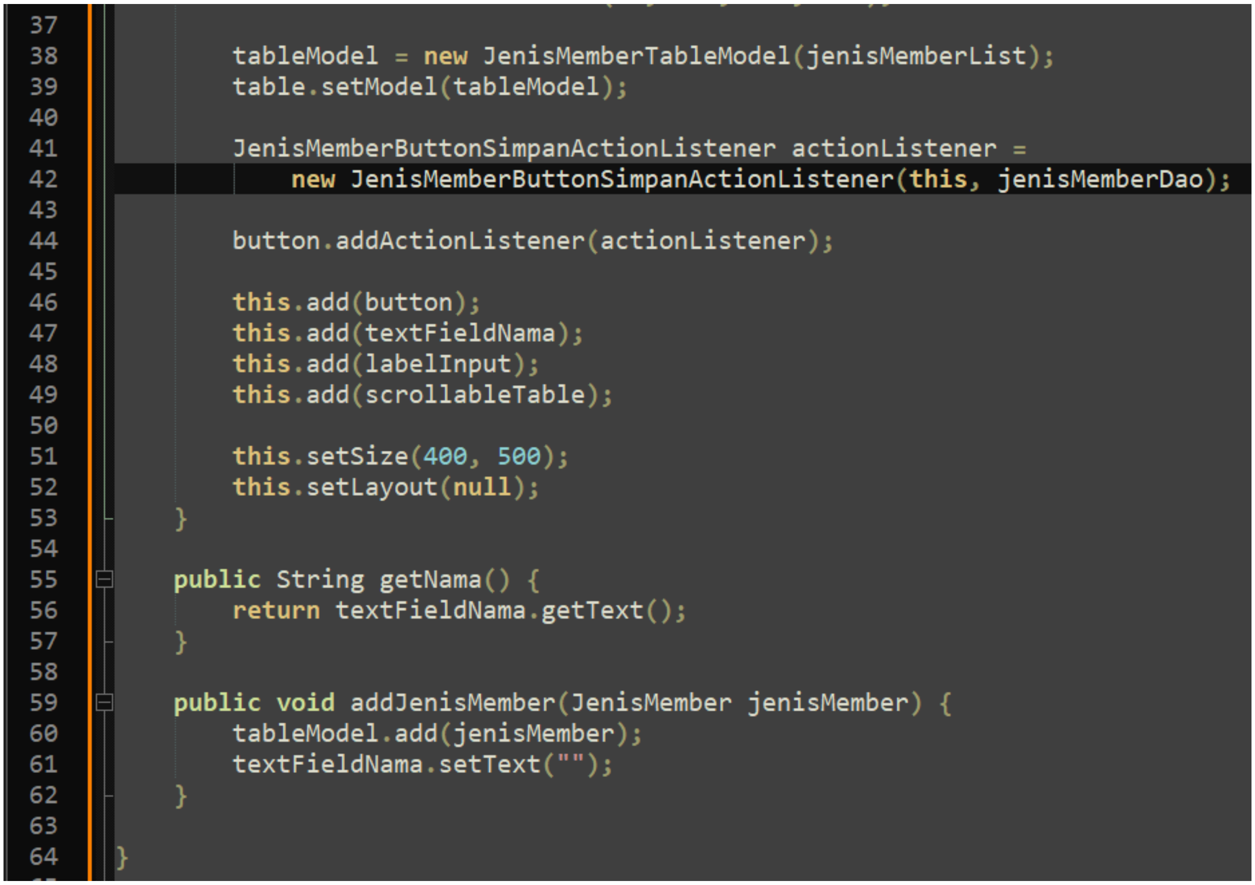
```java

    public Object getValueAt(int row, int col) {
        JenisMember rowItem = data.get(row);
        String value = "";
        switch (col)
        {
            case 0:
                value = rowItem.getNama();
                break;
        }
        return value;
    }

    public boolean isCellEditable(int row, int col) {
        return false;
    }

    public void add(JenisMember value)
    {
        data.add(value);
        fireTableRowsInserted(data.size() - 1, data.size() - 1);
    }
}

```



```java
package view.jenismember;

import dao.JenisMemberDao;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.UUID;
import model.JenisMember;

public class JenisMemberButtonSimpanActionListener implements ActionListener {

    private JenisMemberFrame jenisMemberFrame;
    private JenisMemberDao jenisMemberDao;

    public JenisMemberButtonSimpanActionListener(JenisMemberFrame jenisMemberFrame,
            JenisMemberDao jenisMemberDao) {
        this.jenisMemberFrame = jenisMemberFrame;
        this.jenisMemberDao = jenisMemberDao;
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        String nama = this.jenisMemberFrame.getNama();
        JenisMember jenisMember = new JenisMember();
        jenisMember.setId(UUID.randomUUID().toString());
        jenisMember.setNama(nama);

        this.jenisMemberFrame.addJenisMember(jenisMember);
        this.jenisMemberDao.insert(jenisMember);
    }
}
```

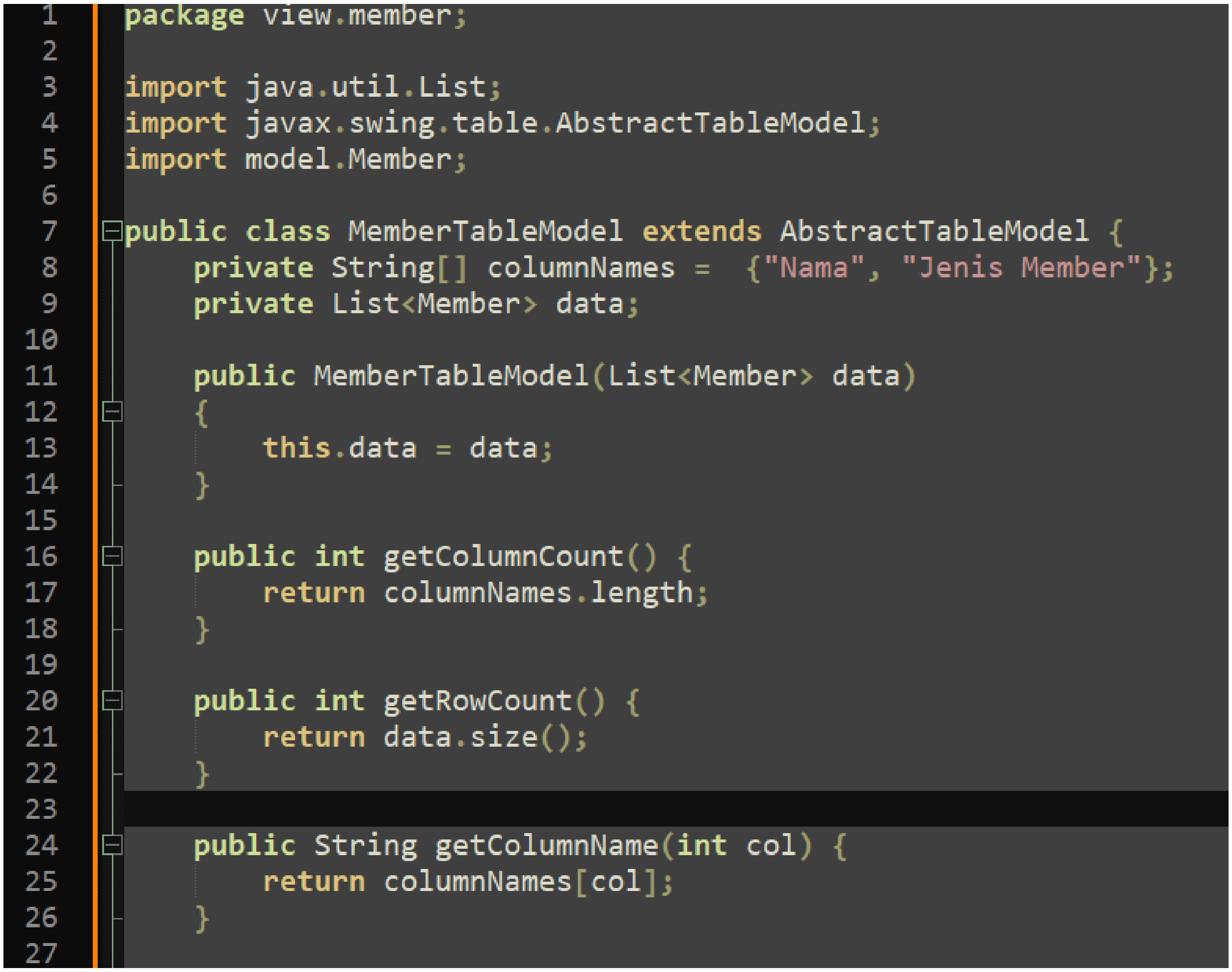


```java
package view.jenismember;

import dao.JenisMemberDao;
import java.util.List;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.JTextField;
import model.JenisMember;

public class JenisMemberFrame extends JFrame {

    private List<JenisMember> jenisMemberList;
    private JTextField textFieldNama;
    private JenisMemberTableModel tableModel;
    private JenisMemberDao jenisMemberDao;

    public JenisMemberFrame(JenisMemberDao jenisMemberDao) {
        this.jenisMemberDao = jenisMemberDao;
        this.jenisMemberList = jenisMemberDao.findAll();
        this.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

        JLabel labelInput = new JLabel("Nama:");
        labelInput.setBounds(15, 40, 350, 10);

        textFieldNama = new JTextField();
        textFieldNama.setBounds(15, 60, 350, 30);

        JButton button = new JButton("Simpan");
        button.setBounds(15, 100, 100, 40);

        javax.swing.JTable table = new JTable();
        JScrollPane scrollableTable = new JScrollPane(table);
        scrollableTable.setBounds(15, 150, 350, 200);

```

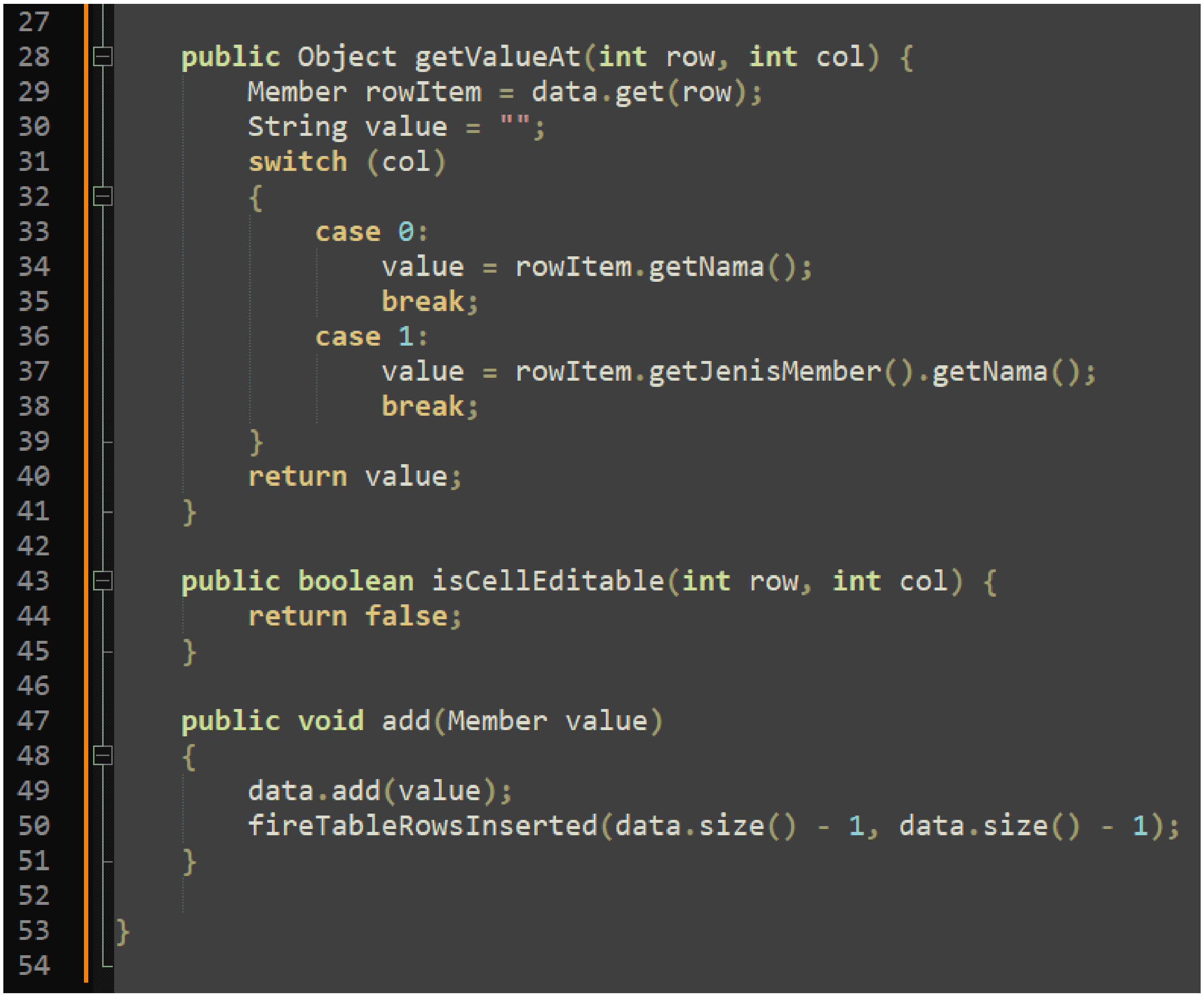


```
        tableModel = new JenisMemberTableModel(jenisMemberList);
        table.setModel(tableModel);

        JenisMemberButtonSimpanActionListener actionListener =
            new JenisMemberButtonSimpanActionListener(this, jenisMemberDao);

        button.addActionListener(actionListener);

        this.add(button);
        this.add(textFieldNama);
        this.add(labelInput);
        this.add(scrollableTable);

        this.setSize(400, 500);
        this.setLayout(null);
    }

    public String getNama() {
        return textFieldNama.getText();
    }

    public void addJenisMember(JenisMember jenisMember) {
        tableModel.add(jenisMember);
        textFieldNama.setText("");
    }

}
```

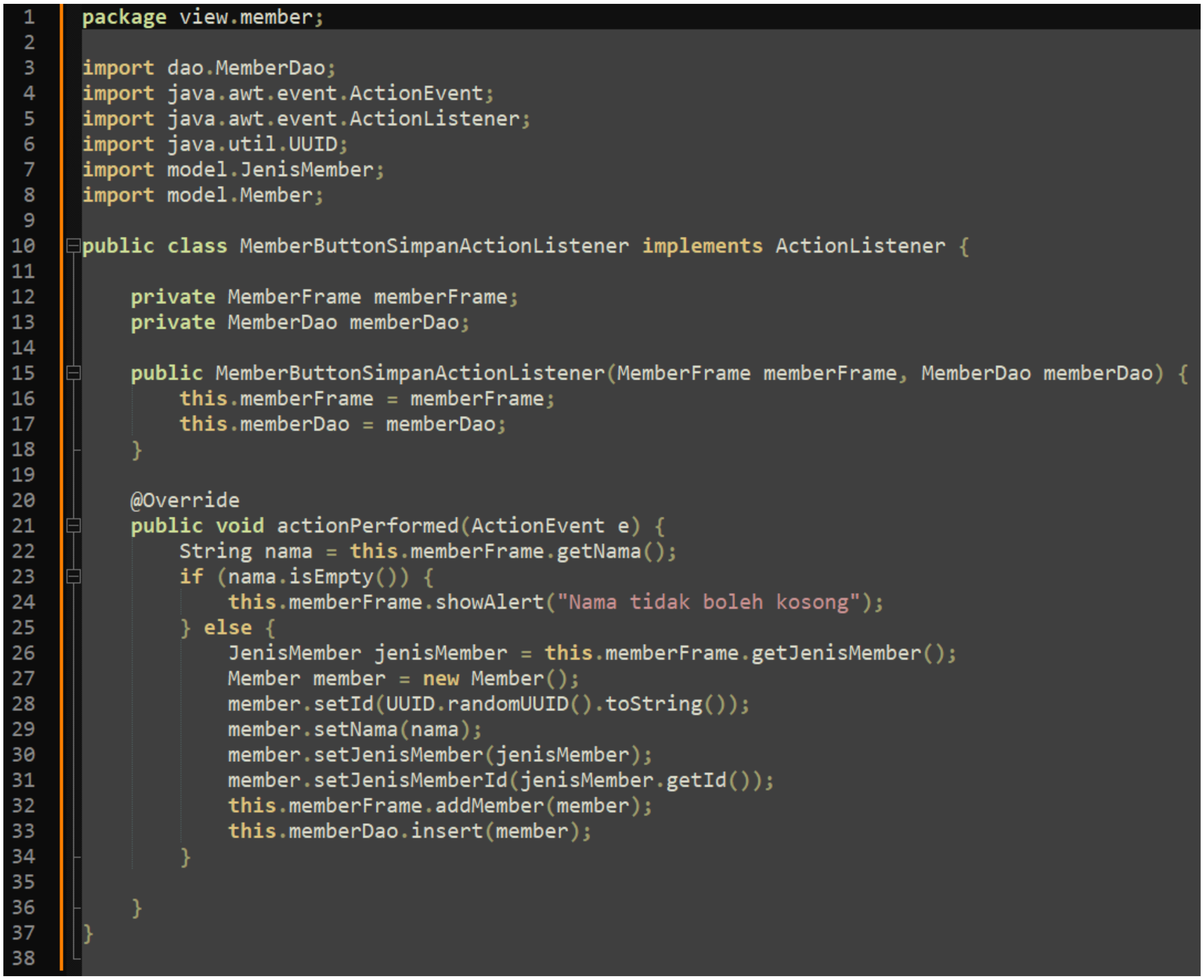
Kemudian, buatlah package view.member dan buat kelas MemberTableModel, MemberButtonSimpanActionListener, dan MemberFrame.



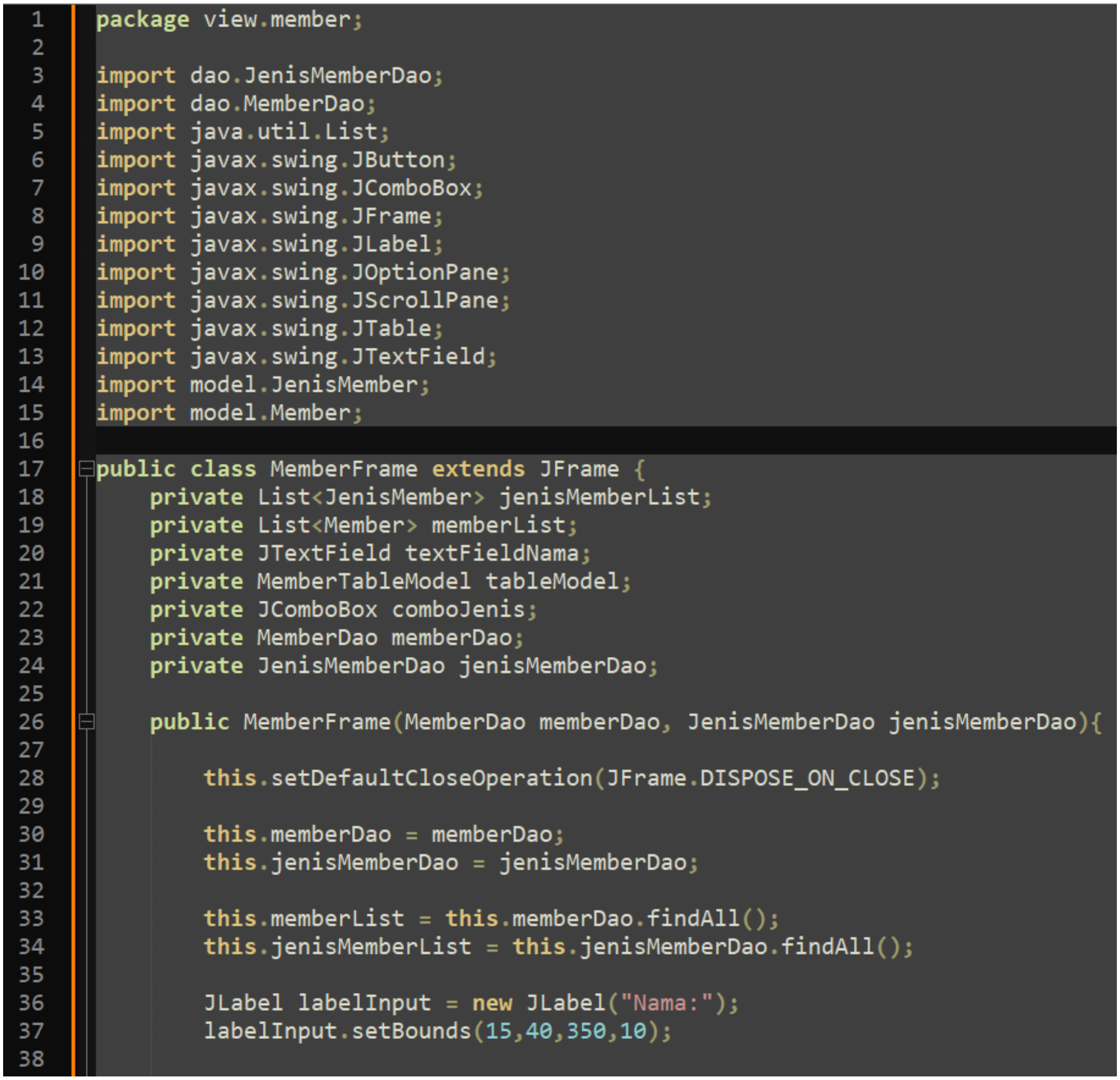
```java
package view.member;

import java.util.List;
import javax.swing.table.AbstractTableModel;
import model.Member;

public class MemberTableModel extends AbstractTableModel {
    private String[] columnNames =  {"Nama", "Jenis Member"};
    private List<Member> data;

    public MemberTableModel(List<Member> data)
    {
        this.data = data;
    }

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

    public int getRowCount() {
        return data.size();
    }

    public String getColumnName(int col) {
        return columnNames[col];
    }

```



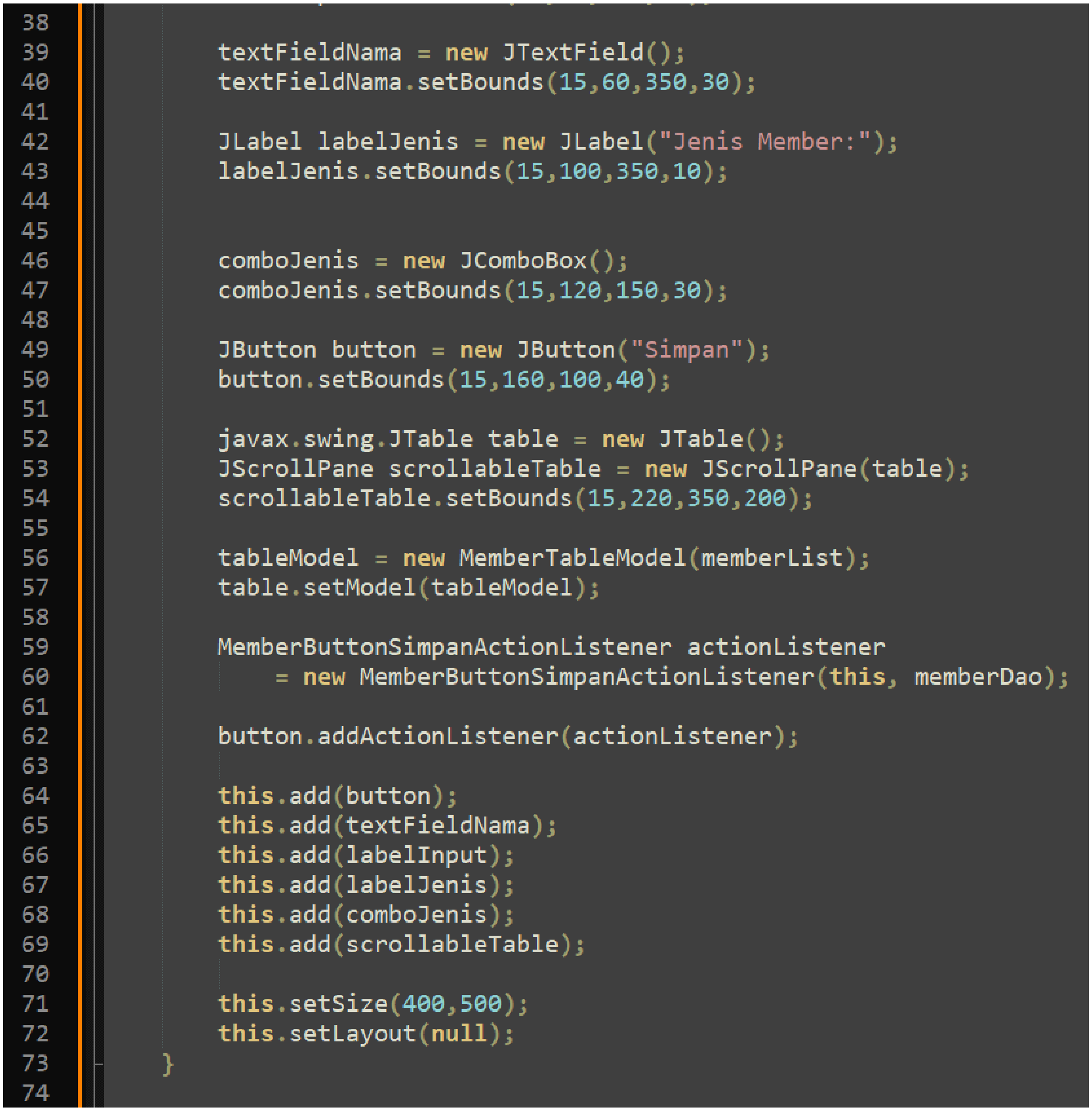
```java

    public Object getValueAt(int row, int col) {
        Member rowItem = data.get(row);
        String value = "";
        switch (col)
        {
            case 0:
                value = rowItem.getNama();
                break;
            case 1:
                value = rowItem.getJenisMember().getNama();
                break;
        }
        return value;
    }

    public boolean isCellEditable(int row, int col) {
        return false;
    }

    public void add(Member value)
    {
        data.add(value);
        fireTableRowsInserted(data.size() - 1, data.size() - 1);
    }

}

```



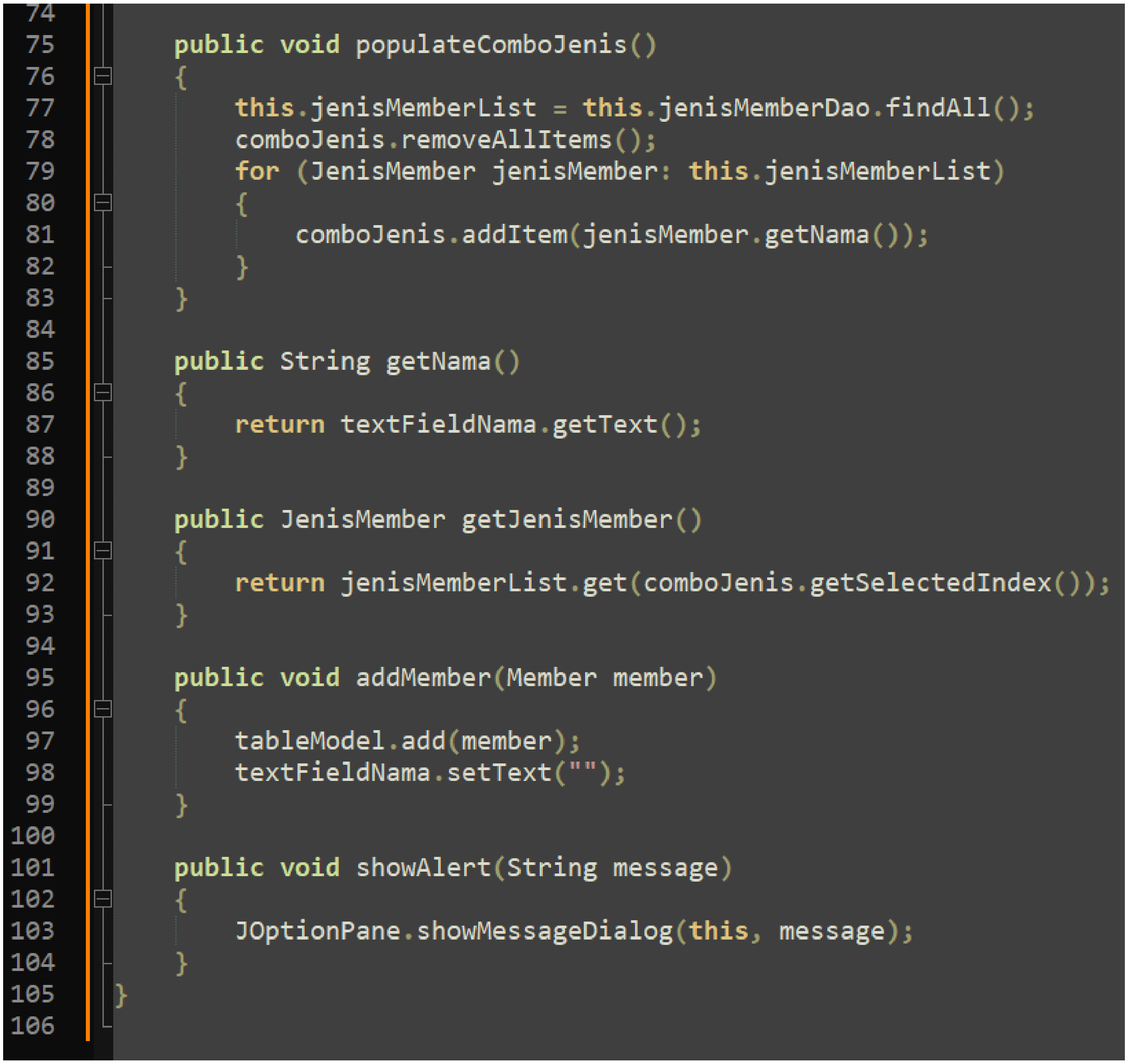
```java
package view.member;

import dao.MemberDao;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.UUID;
import model.JenisMember;
import model.Member;

public class MemberButtonSimpanActionListener implements ActionListener {

    private MemberFrame memberFrame;
    private MemberDao memberDao;

    public MemberButtonSimpanActionListener(MemberFrame memberFrame, MemberDao memberDao) {
        this.memberFrame = memberFrame;
        this.memberDao = memberDao;
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        String nama = this.memberFrame.getNama();
        if (nama.isEmpty()) {
            this.memberFrame.showAlert("Nama tidak boleh kosong");
        } else {
            JenisMember jenisMember = this.memberFrame.getJenisMember();
            Member member = new Member();
            member.setId(UUID.randomUUID().toString());
            member.setNama(nama);
            member.setJenisMember(jenisMember);
            member.setJenisMemberId(jenisMember.getId());
            this.memberFrame.addMember(member);
            this.memberDao.insert(member);
        }

    }
}
```



```java
package view.member;

import dao.JenisMemberDao;
import dao.MemberDao;
import java.util.List;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.JTextField;
import model.JenisMember;
import model.Member;

public class MemberFrame extends JFrame {
    private List<JenisMember> jenisMemberList;
    private List<Member> memberList;
    private JTextField textFieldNama;
    private MemberTableModel tableModel;
    private JComboBox comboJenis;
    private MemberDao memberDao;
    private JenisMemberDao jenisMemberDao;

    public MemberFrame(MemberDao memberDao, JenisMemberDao jenisMemberDao){

        this.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

        this.memberDao = memberDao;
        this.jenisMemberDao = jenisMemberDao;

        this.memberList = this.memberDao.findAll();
        this.jenisMemberList = this.jenisMemberDao.findAll();

        JLabel labelInput = new JLabel("Nama:");
        labelInput.setBounds(15,40,350,10);

```

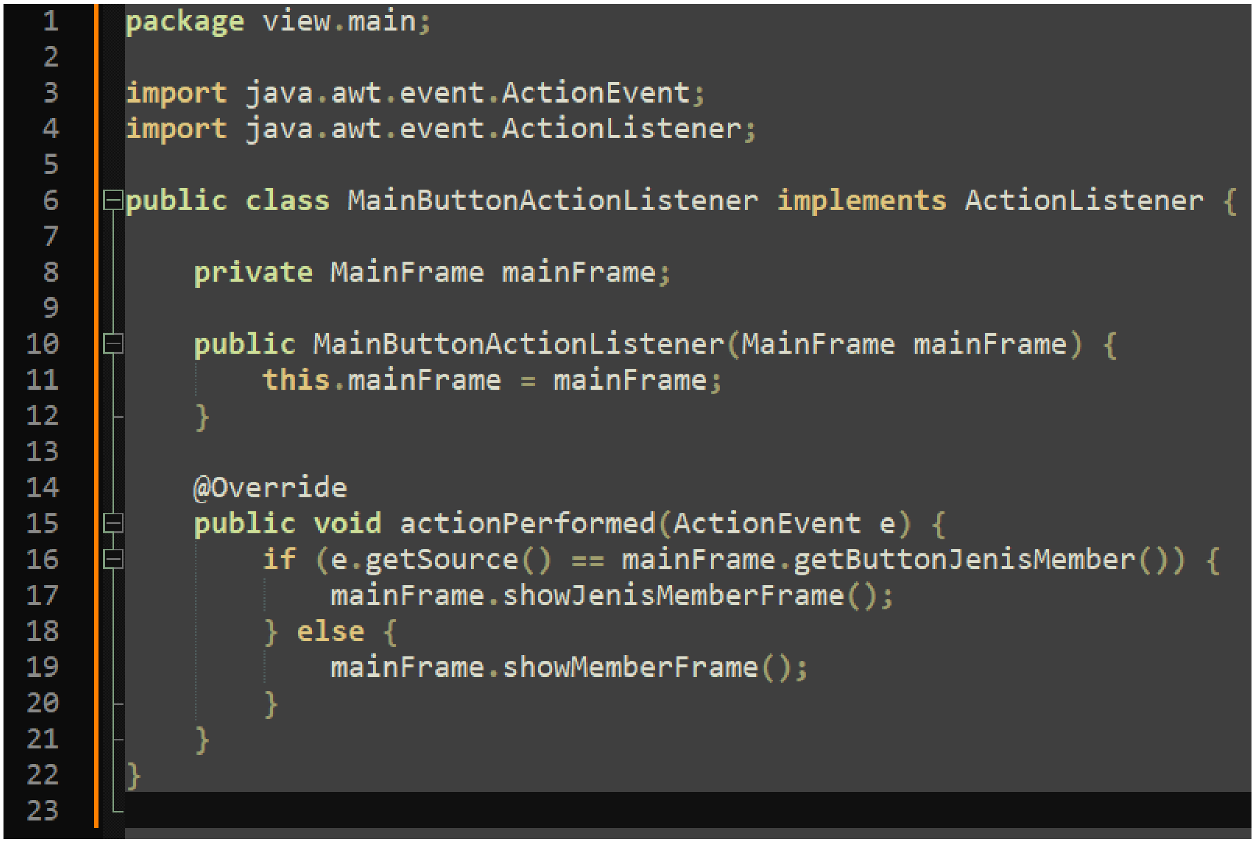


```
        textFieldNama = new JTextField();
        textFieldNama.setBounds(15,60,350,30);

        JLabel labelJenis = new JLabel("Jenis Member:");
        labelJenis.setBounds(15,100,350,10);


        comboJenis = new JComboBox();
        comboJenis.setBounds(15,120,150,30);

        JButton button = new JButton("Simpan");
        button.setBounds(15,160,100,40);

        javax.swing.JTable table = new JTable();
        JScrollPane scrollableTable = new JScrollPane(table);
        scrollableTable.setBounds(15,220,350,200);

        tableModel = new MemberTableModel(memberList);
        table.setModel(tableModel);

        MemberButtonSimpanActionListener actionListener
            = new MemberButtonSimpanActionListener(this, memberDao);

        button.addActionListener(actionListener);

        this.add(button);
        this.add(textFieldNama);
        this.add(labelInput);
        this.add(labelJenis);
        this.add(comboJenis);
        this.add(scrollableTable);

        this.setSize(400,500);
        this.setLayout(null);
    }

```

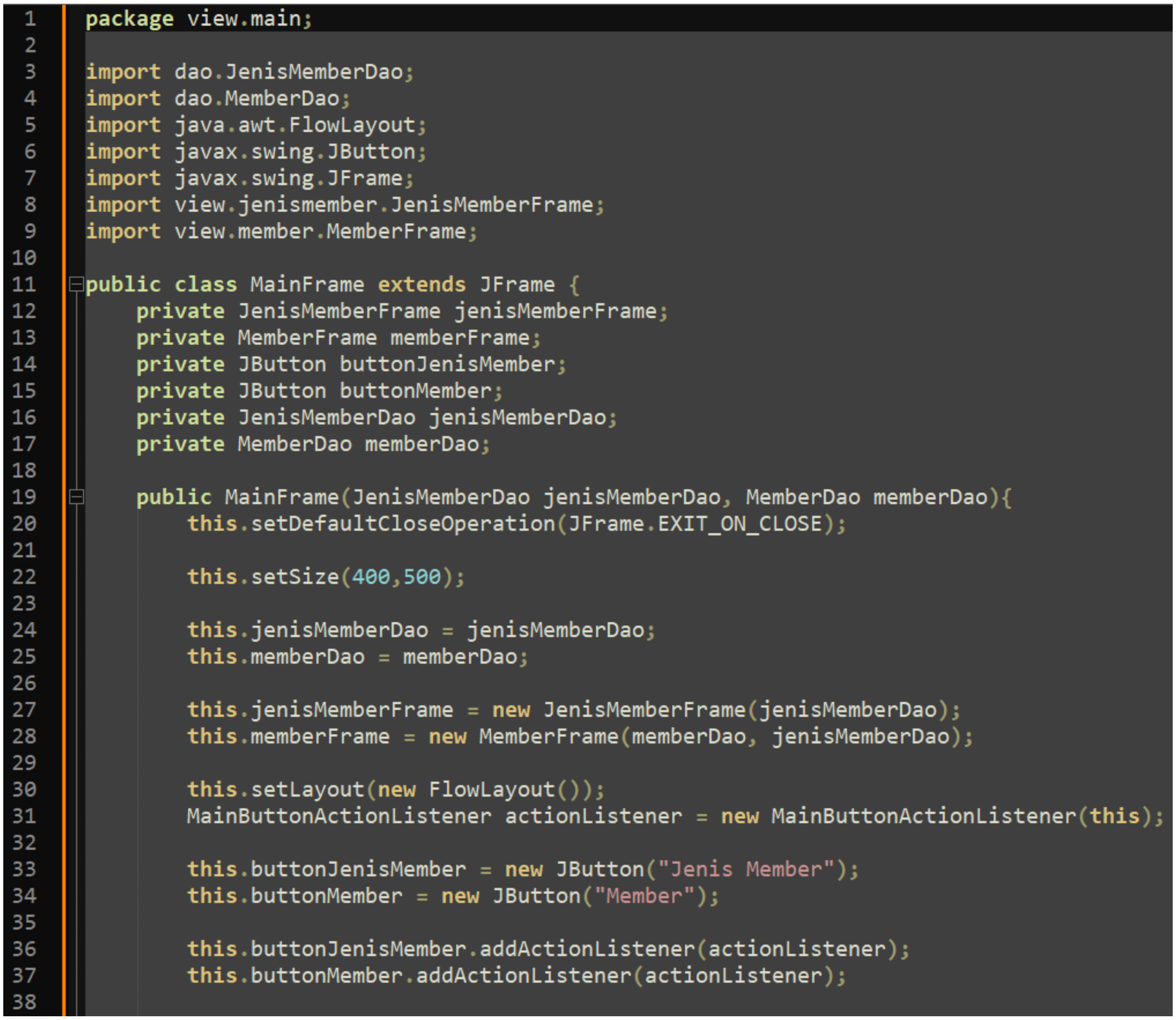


```java

    public void populateComboJenis()
    {
        this.jenisMemberList = this.jenisMemberDao.findAll();
        comboJenis.removeAllItems();
        for (JenisMember jenisMember: this.jenisMemberList)
        {
            comboJenis.addItem(jenisMember.getNama());
        }
    }

    public String getNama()
    {
        return textFieldNama.getText();
    }

    public JenisMember getJenisMember()
    {
        return jenisMemberList.get(comboJenis.getSelectedIndex());
    }

    public void addMember(Member member)
    {
        tableModel.add(member);
        textFieldNama.setText("");
    }

    public void showAlert(String message)
    {
        JOptionPane.showMessageDialog(this, message);
    }
}

```

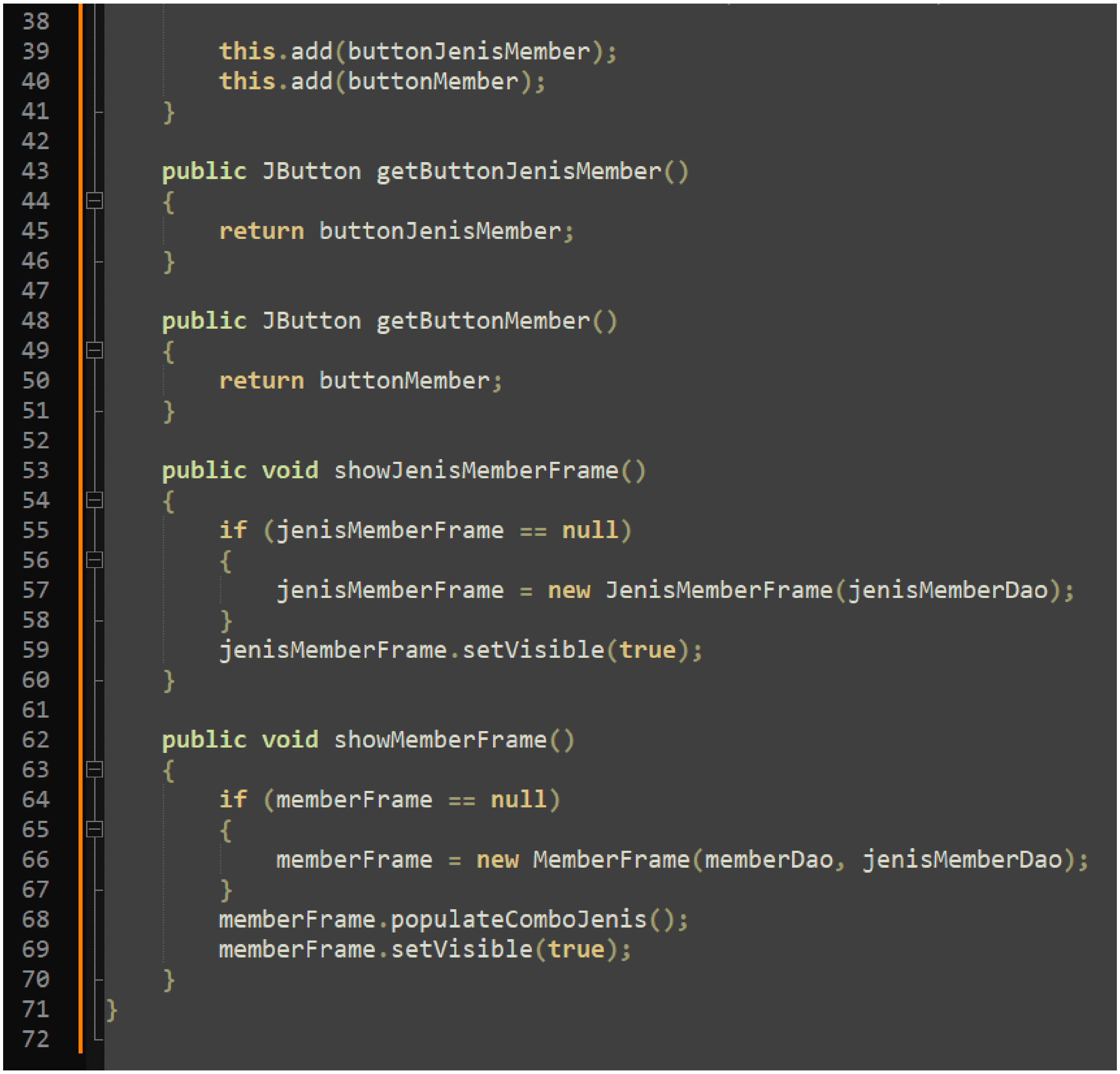
Kemudian, buat package view.main lalu kelas MainButtonActionListener dan MainFrame.



```java
package view.main;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class MainButtonActionListener implements ActionListener {

    private MainFrame mainFrame;

    public MainButtonActionListener(MainFrame mainFrame) {
        this.mainFrame = mainFrame;
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == mainFrame.getButtonJenisMember()) {
            mainFrame.showJenisMemberFrame();
        } else {
            mainFrame.showMemberFrame();
        }
    }
}

```

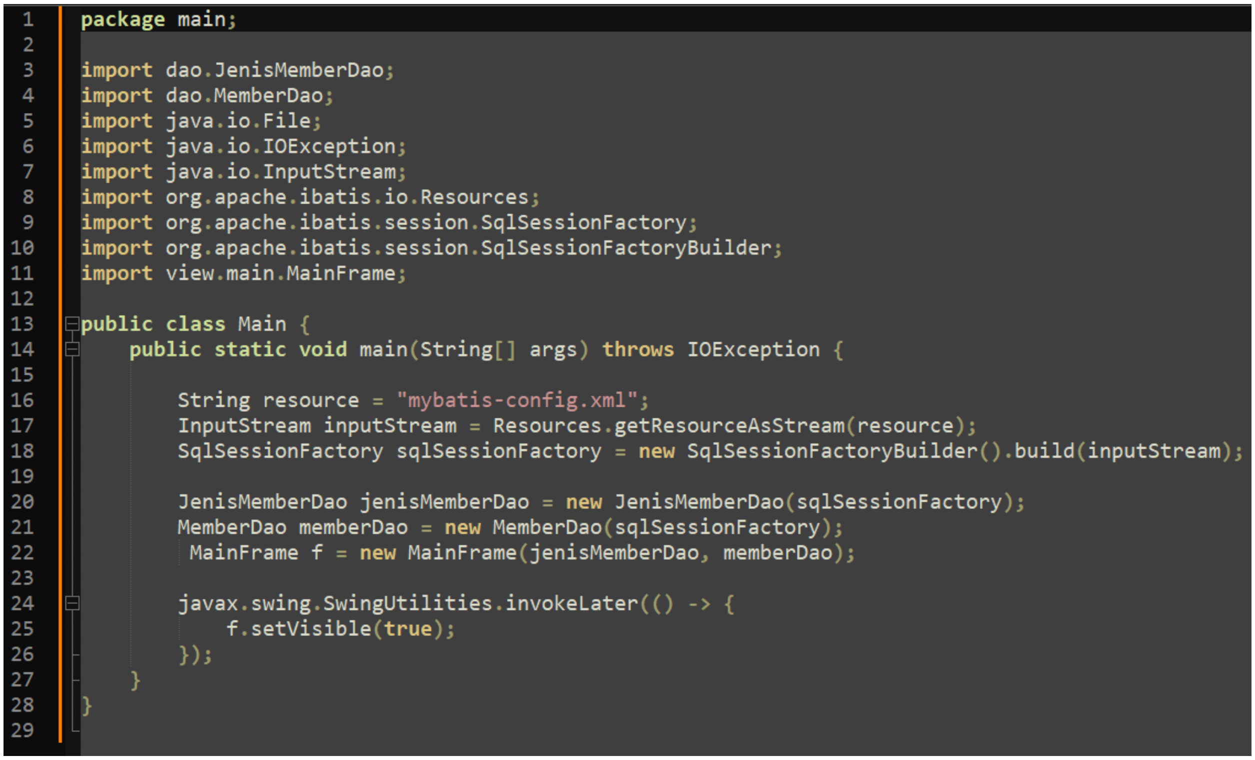


```java
package view.main;

import dao.JenisMemberDao;
import dao.MemberDao;
import java.awt.FlowLayout;
import javax.swing.JButton;
import javax.swing.JFrame;
import view.jenismember.JenisMemberFrame;
import view.member.MemberFrame;

public class MainFrame extends JFrame {
    private JenisMemberFrame jenisMemberFrame;
    private MemberFrame memberFrame;
    private JButton buttonJenisMember;
    private JButton buttonMember;
    private JenisMemberDao jenisMemberDao;
    private MemberDao memberDao;

    public MainFrame(JenisMemberDao jenisMemberDao, MemberDao memberDao){
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        this.setSize(400,500);

        this.jenisMemberDao = jenisMemberDao;
        this.memberDao = memberDao;

        this.jenisMemberFrame = new JenisMemberFrame(jenisMemberDao);
        this.memberFrame = new MemberFrame(memberDao, jenisMemberDao);

        this.setLayout(new FlowLayout());
        MainButtonActionListener actionListener = new MainButtonActionListener(this);

        this.buttonJenisMember = new JButton("Jenis Member");
        this.buttonMember = new JButton("Member");

        this.buttonJenisMember.addActionListener(actionListener);
        this.buttonMember.addActionListener(actionListener);

```



```java

        this.add(buttonJenisMember);
        this.add(buttonMember);
    }

    public JButton getButtonJenisMember()
    {
        return buttonJenisMember;
    }

    public JButton getButtonMember()
    {
        return buttonMember;
    }

    public void showJenisMemberFrame()
    {
        if (jenisMemberFrame == null)
        {
            jenisMemberFrame = new JenisMemberFrame(jenisMemberDao);
        }
        jenisMemberFrame.setVisible(true);
    }

    public void showMemberFrame()
    {
        if (memberFrame == null)
        {
            memberFrame = new MemberFrame(memberDao, jenisMemberDao);
        }
        memberFrame.populateComboJenis();
        memberFrame.setVisible(true);
    }
}

```

Kemudian, buat package main dan kelas Main sebagai berikut.



```java
package main;

import dao.JenisMemberDao;
import dao.MemberDao;
import java.io.File;
import java.io.IOException;
import java.io.InputStream;
import org.apache.ibatis.io.Resources;
import org.apache.ibatis.session.SqlSessionFactory;
import org.apache.ibatis.session.SqlSessionFactoryBuilder;
import view.main.MainFrame;

public class Main {
    public static void main(String[] args) throws IOException {

        String resource = "mybatis-config.xml";
        InputStream inputStream = Resources.getResourceAsStream(resource);
        SqlSessionFactory sqlSessionFactory = new SqlSessionFactoryBuilder().build(inputStream);

        JenisMemberDao jenisMemberDao = new JenisMemberDao(sqlSessionFactory);
        MemberDao memberDao = new MemberDao(sqlSessionFactory);
         MainFrame f = new MainFrame(jenisMemberDao, memberDao);

        javax.swing.SwingUtilities.invokeLater(() -> {
            f.setVisible(true);
        });
    }
}

```

Selanjutnya, compile kelas Main dengan perintah berikut

javac -cp "C:\Users\wdgus\Documents\Unpas\Praktikum Prog II\MembershipORM\src;C:\Users\wdgus\Documents\Unpas\Praktikum Prog II\MembershipORM\libs\*" src/main/Main.java

Kita perlu meng-compile kelas di dalam package mapper karena kelas-kelas tersebut tidak memiliki dependency langsung dengan kelas Main.

javac -cp "C:\Users\wdgus\Documents\Unpas\Praktikum Prog II\MembershipORM\src;C:\Users\wdgus\Documents\Unpas\Praktikum Prog II\MembershipORM\libs\*" src/mapper/*.java

Kemudian, jalankan kelas Main dengan perintah berikut.

java -cp "C:\Users\wdgus\Documents\Unpas\Praktikum Prog II\MembershipORM\src;C:\Users\wdgus\Documents\Unpas\Praktikum Prog II\MembershipORM\libs\*" main.Main

Latihan 2

Lengkapi implementasi update dan delete data pada aplikasi di Latihan 1.