

Nama : Muhammad Rafasya Putra Ristianto

NIM : 24060121130074

Lab : C1

1. PersonDAO.java

```
/**
 * File : PersonDAO.java 31/05/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
 * Deskripsi : Interface untuk person access object
 */

public interface PersonDAO{
    public void savePerson(Person p) throws Exception;
}
```

2. Person.java

```
/**
 * File : Person.java 31/05/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
 * Deskripsi : Person database model
 */

public class Person {
    private int id;
    private String name;

    public Person(String n){
        name = n;
    }

    public Person(int i, String n){
        id = i;
        name = n;
    }

    public int getId(){
        return id;
    }

    public String getName(){
        return name;
    }
}
```

3. MySQLPerson.java

```
/**
 * File : MySQLPersonDAO.java 31/05/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
```

```

    * Deskripsi : Implementasi PersonDAO untuk MySQL
    */

import java.sql.*;

public class MySQLPersonDAO implements PersonDAO{
    public void savePerson(Person person) throws
    Exception{
        String name = person.getName();
        Class.forName("com.mysql.jdbc.Driver");
        Connection con =
        DriverManager.getConnection("jdbc:mysql://localhost/pbo
        ", "root", "");
        String query = "INSERT INTO person(name)
        VALUES ('"+name+"')";
        System.out.println(query);
        Statement s = con.createStatement();
        s.executeUpdate(query);
        con.close();
    }
}

```

#### 4. DAOManager.java

```

/**
 * File : DAOManager.java 31/05/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
 * Deskripsi : Pengelola DAO dalam program
 */

public class DAOManager {
    private PersonDAO personDAO;
    public void setPersonDAO(PersonDAO person){
        personDAO = person;
    }

    public PersonDAO getPersonDAO(){
        return personDAO;
    }
}

```

#### 5. MainDAO.java

```

/**
 * File : MainDAO.java 31/05/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
 * Deskripsi : Main program untuk akses DAO
 */

public class MainDAO {
    public static void main(String args[]){
        Person person = new Person("Indra");
    }
}

```

```

        DAOManager m = new DAOManager();
        m.setPersonDAO(new MySQLPersonDAO());
        try{
            m.getPersonDAO().savePerson(person);
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}

```

## 6. Command Line MainDAO

```

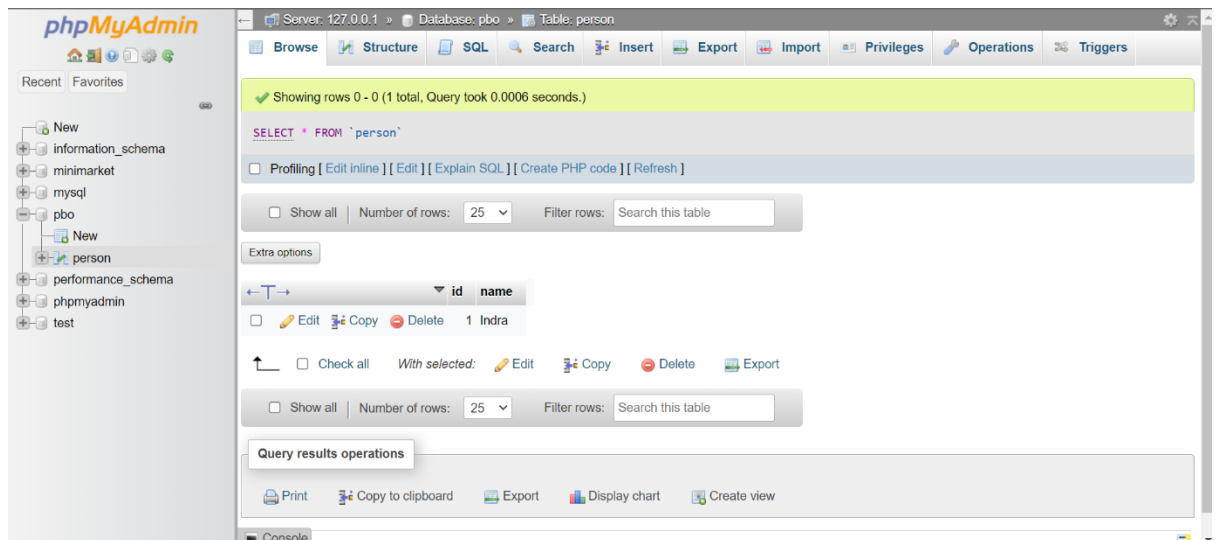
D:\Kuliah\Semester 4\Prak PBO\Praktikum 9>javac *.java

D:\Kuliah\Semester 4\Prak PBO\Praktikum 9>java -classpath .\mysql-connector-j-8.0.33.jar;. MainDAO
Loading class 'com.mysql.jdbc.Driver'. This is deprecated. The new driver class is 'com.mysql.cj.jdbc.Driver'. The driver is automati
cally registered via the SPI and manual loading of the driver class is generally unnecessary.
INSERT INTO person(name) VALUES('Indra')

D:\Kuliah\Semester 4\Prak PBO\Praktikum 9>

```

## 7. Database



## 8. SerializePerson.java

```

/**
 * File : SerializePerson.java 1/06/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
 * Deskripsi : program untuk serialisasi objek person
 */

import java.io.*;
class Person implements Serializable {
    private String name;
    public Person(String n) {
        name = n;
    }
    public String getName() {
        return name;
    }
}

```

```

public class SerializePerson {
    public static void main(String[] args) {
        Person person = new Person("Panji");
        try {
            FileOutputStream f = new
FileOutputStream("person.ser");
            ObjectOutputStream s = new
ObjectOutputStream(f);
            s.writeObject(person);
            System.out.println("Selesai menulis objek
person");
            s.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

```

#### 9. ReadSerializedPerson.java

```

/**
 * File : ReadSerializedPerson.java 1/06/2023
 * Penulis : Muhammad Rafasya Putra Ristianto
 * Deskripsi : program untuk serialisasi objek person
 */

import java.io.*;
public class ReadSerializedPerson {
    public static void main(String[] args) {
        Person person = null;
        try {
            FileInputStream f = new
FileInputStream("person.ser");
            ObjectInputStream s = new
ObjectInputStream(f);
            person = (Person) s.readObject();
            s.close();
            System.out.println("serialized person name
= " + person.getName());
        } catch (Exception ioe) {
            ioe.printStackTrace();
        }
    }
}

```

#### 10. Hasil Output

```
D:\Kuliah\Semester 4\Prak PBO\Praktikum 9>java SerializePerson  
Selesai menulis objek person
```

```
D:\Kuliah\Semester 4\Prak PBO\Praktikum 9>java ReadSerializedPerson  
serialized person name = Panji
```