

LAPORAN PRAKTIKUM
PRAKTIKUM 9:
“PERSISTENT OBJECT”



Disusun Oleh :

Zenobia Wirandi Zenaide
24060121140164

PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK
LAB B2

DEPARTEMEN ILMU KOMPUTER / INFORMATIKA
FAKULTAS SAINS DAN MATEMATIKA
UNIVERSITAS DIPONEGORO
SEMARANG
2023

A. Menggunakan Persistensi Objek sebagai model Basis Data

1. PersonDAO.java

```
/**
 * Penulis : Zenobia Wirandi Zenaide
 * File : PersonDAO.java
 * Deskripsi : interface untuk person access object
 * Tanggal : 06/06/2023
 */

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

2. DAOManager.java

```
/**
 * Penulis : Zenobia Wirandi Zenaide
 * File : DAOManager.java
 * Deskripsi : pengelola DAO dalam program
 * Tanggal : 06/06/2023
 */

public class DAOManager{
    private PersonDAO personDAO;

    public void setPersonDAO(PersonDAO person){
        personDAO = person;
    }
    public PersonDAO getPersonDAO(){
        return personDAO;
    }
}
```

3. MainDAO.java

```
/**
 * Penulis : Zenobia Wirandi Zenaide
 * File : MainDAO.java
 * Deskripsi : Main program untuk akses DAO
 * tanggal : 06/06/2023
 */

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();
        }
    }
}
```

```
}  
}
```

4. MySQLPersonDAO.java

```
/**  
 * Penulis : Zenobia Wirandi Zenaide  
 * File : MySQLPersonDAO.java  
 * Deskripsi : implementasi PersonDAO untuk MySQL  
 * tanggal : 06/06/2023  
 **/  
import java.sql.*;  
  
public class MySQLPersonDAO implements PersonDAO{  
    public void SavePerson(Person person) throws Exception{  
        String name = person.getName();  
        //membuat koneksi,nama db,user,password menyesuaikan  
        Class.forName("com.mysql.jdbc.Driver");  
        Connection con = DriverManager.getConnection(  
  
            "jdbc:mysql://localhost/pbo","root","fira1212");  
        //kerjakan mysql query  
        String query = "INSERT INTO person(name) VALUES ('"+name+"')";  
        System.out.println(query);  
        Statement s = con.createStatement();  
        s.executeUpdate(query);  
        //tutup koneksi database  
        con.close();  
    }  
}
```

5. Person

```
/**  
 * Penulis : Zenobia Wirandi Zenaide  
 * File : Person.java  
 * Deskripsi : person database model  
 * Tanggal : 06/06/2023  
 **/  
  
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;
}

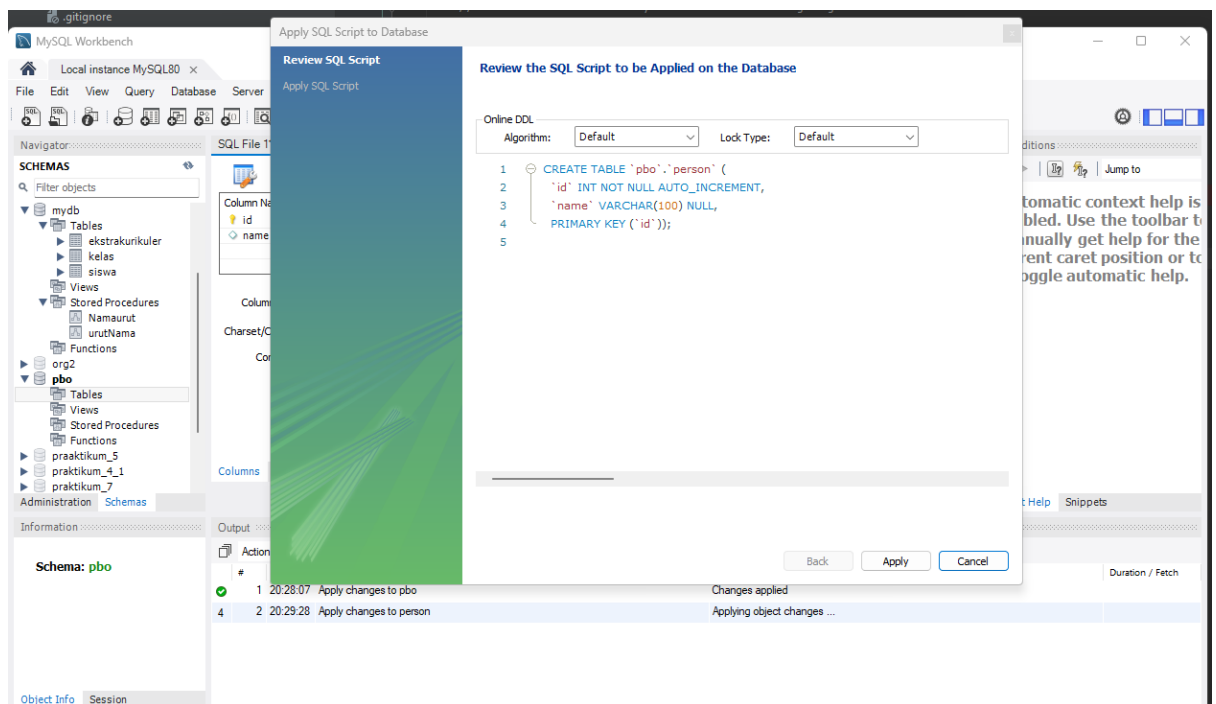
public String getName(){
    return name;
}
}

```

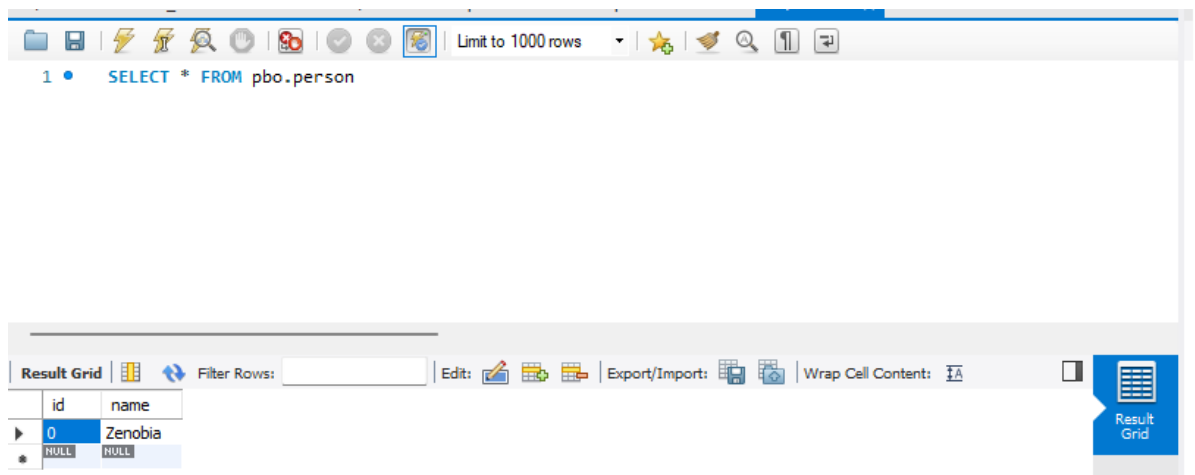
B. Membuat database dengan nama 'pbo' dan table pada database.

Pada pembuatan database pbo, dan beberapa tabelnya, dibuat menggunakan software IntelliJ IDEA, seperti pada lampiran di bawah

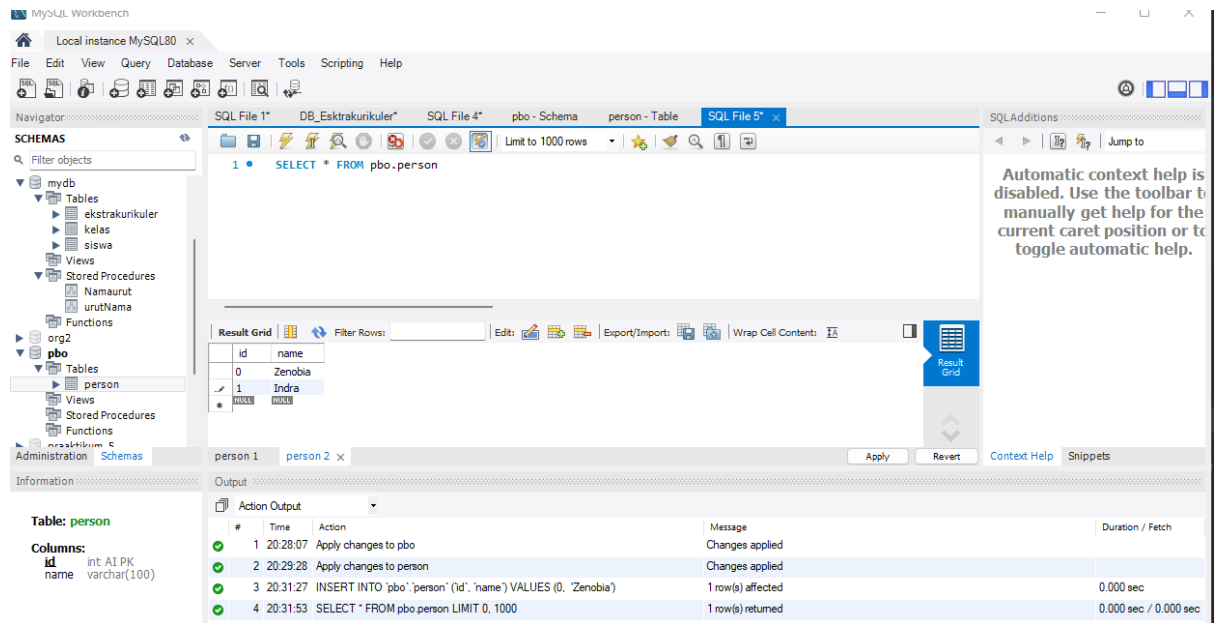
- Input create table



- Menunjukkan hasil table dengan SELECT * FROM



- Setelah mengeksekusi kode program di atas, maka data baru berhasil di tambahkan seperti pada lampiran di bawah ini:



C. Menggunakan Persistent Object sebagai objek terserialis

1. SerializePerson.java

```

/**
    Penulis : Zenobia Wirandi Zenaide 06/06/2023
    File : SerializePerson.java
    Deskripsi : Program untuk serialisasi objek person
    tanggal : 06/06/2023
    */

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

```

```

        s.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}
}

```

```

D:\UNDIP\semester 4\PRAK PBO\prak 9>jvac SerializePerson.java
'jvac' is not recognized as an internal or external command,
operable program or batch file.

D:\UNDIP\semester 4\PRAK PBO\prak 9>javac SerializePerson.java

D:\UNDIP\semester 4\PRAK PBO\prak 9>java SerializePerson
selesai menulis onjek person

D:\UNDIP\semester 4\PRAK PBO\prak 9>

```

Ketika kita membuka command prompt, dan menuliskan javac SerializedPerson.java, dan berhasil di compile, maka kita telah sukses menjalankan perintah tersebut tanpa kesalahan dan Ketika di java, menghasilkan keluaran “selesai menulis objek person”. Hal tersebut menandakan bahwa penulisan objek person sudah sesuai dengan pengaturan serialisasi sebelumnya.

2. ReadSerializedPerson.java

```

/**
 * Penulis : Zenobia Wirandi Zenaide
 * File : ReadSerializedPerson.java
 * Deskripsi : Program untuk serialisasi objek person
 * tanggal : 06/06/2023
 */

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();
        }
    }
}

```

```
Microsoft Windows [Version 10.0.22621.1702]
(c) Microsoft Corporation. All rights reserved.

D:\UNDIP\semester 4\PRAK PBO\prak 9>javac ReadSerializedPerson.java

D:\UNDIP\semester 4\PRAK PBO\prak 9>java ReadSerializedPerson
serialized person name = Panji

D:\UNDIP\semester 4\PRAK PBO\prak 9>
```

Ketika menjalankan perintah seperti pada lampiran, maka program `readSerializedPerson` akan di compile dengan menggunakan `javac` "`namafile`".`java` kalau berhasil, maka akan keluar pesan "`serialized person name = panji`" dimana hal itu menampilkan hasil informasi dari objek yang sebelumnya telah di serialized.