

**LAPORAN PRAKTIKUM 9:**

**“PERSISTENT OBJECT”**



**Disusun Oleh :**

**Zaeri Haikal Rabbani**

**24060121140159**

**PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK LAB B1**

**DEPARTEMEN ILMU KOMPUTER/INFORMATIKA**

**FAKULTAS SAINS DAN MATEMATIKA**

**UNIVERSITAS DIPONEGORO**

**SEMARANG**

**2023**

## Persistent Object

### A. Menggunakan Persistent Object sebagai Model Basis Data Relasional

#### 1. PersonDAO.java

```
/**
 * File : PersonDAO.java
 * Nama : Zaeri Haikal Rabbani
 * NIM : 24060121140159
 * Tanggal : 05 Juni 2023
 * Deskripsi : interface untuk person access object
 */
public interface PersonDAO{
    public void savePerson(Person p) throws Exception;
}
```

#### 2. Person.java

```
/**
 * File : PersonDAO.java
 * Nama : Zaeri Haikal Rabbani
 * NIM : 24060121140159
 * Tanggal : 05 Juni 2023
 * 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. MySQLPersonDAO.java

```
/**
 * File : PersonDAO.java
 * Nama : Zaeri Haikal Rabbani
 * NIM : 24060121140159
 * Tanggal : 05 Juni 2023
 * Deskripsi : implementasi PersonDAO untuk MySQL
 */
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","haikal123");
        //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();
    }
}
```

### 4. DAOManager.java

```
/**
 * File : DAOManager.java
 * Nama : Zaeri Haikal Rabbani
 * NIM : 24060121140159
 * Tanggal : 05 Juni 2023
 * 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

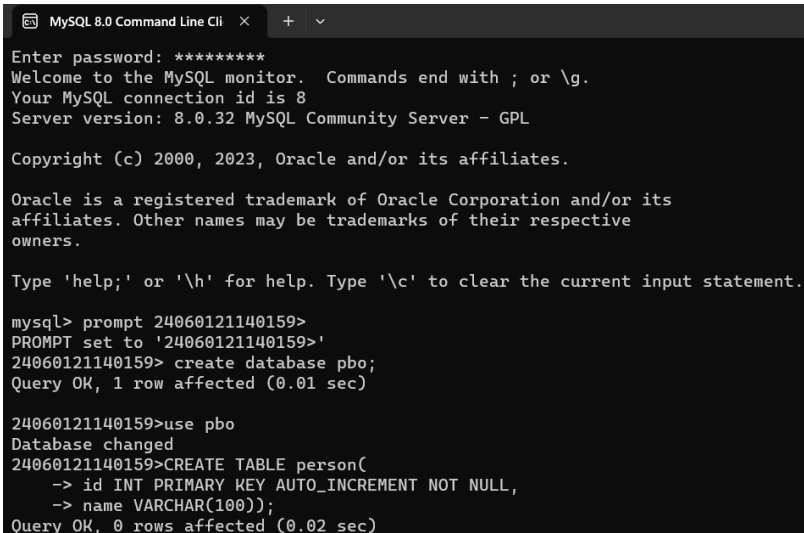
```

/**
 * File : MainDAO.java
 * Nama : Zaeri Haikal Rabbani
 * NIM : 24060121140159
 * Tanggal : 05 Juni 2023
 * 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. Buat database pbo dan tabelnya



```

MySQL 8.0 Command Line Cli  x  +  v
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> prompt 24060121140159>
PROMPT set to '24060121140159>'
24060121140159> create database pbo;
Query OK, 1 row affected (0.01 sec)

24060121140159> use pbo
Database changed
24060121140159> CREATE TABLE person(
    -> id INT PRIMARY KEY AUTO_INCREMENT NOT NULL,
    -> name VARCHAR(100));
Query OK, 0 rows affected (0.02 sec)

```

Untuk membuat database 'pbo' dan tabel dilakukan perintah CREATE TABLE pbo;. Setelah berhasil dibuat, gunakan database tersebut dengan perintah USE pbo;. Selanjutnya, menggunakan perintah CREATE TABLE person untuk membuat tabel person pada database pbo. Tabel yang akan dibuat berisi dua kolom yaitu id dan name. Kolom id ini memiliki tipe data INT dan diatur sebagai primary key dengan opsi AUTO\_INCREMENT, yang berarti nilai id akan dihasilkan secara otomatis. Kolom juga diatur sebagai NOT NULL yang berarti kolom tidak boleh kosong. Pada kolom memiliki tipe data VARCHAR(100) yang berarti dapat menampung data string dengan maksimal 100 karakter.

## 7. *Compile semua source code*




```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\Kulijing\java gamink\Pertemuan 9> javac *.java
PS D:\Kulijing\java gamink\Pertemuan 9>
```

Compile dengan perintah javac \*.java. Dapat dilihat gambar di atas bahwa setelah perintah di jalankan, tidak ada erorr muncul yang berarti source code yang dibuat sudah berhasil.

## 8. **Jalankan MainDAO**



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.22621.1782]
(c) Microsoft Corporation. All rights reserved.

D:\Kulijing\java gamink\Pertemuan 9>javac *.java

D:\Kulijing\java gamink\Pertemuan 9>
D:\Kulijing\java gamink\Pertemuan 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 automatically registered via the SPI and manual loading of the driver class is generally unnecessary.
INSERT INTO person(name) VALUES('Indra')

D:\Kulijing\java gamink\Pertemuan 9>
```

Menjalankan MainDAO dengan perintah java -classpath .\mysql-connector-j-8.0.33.jar;. MainDAO. Ketika ingin menjalankan perintah di atas, file program MainDAO dan mysql.jar harus berada pada satu folder.

Setelah menjalankan perintah tersebut, akan muncul pesan bahwa MainDAO berhasil dijalankan dan perintah untuk memasukkan data dalam tabel person sudah dilakukan dengan muncul pesan INSERT INTO person(name) VALUES('Indra').

```
MySQL 8.0 Command Line Cll  x  +  v
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8
Server version: 8.0.32 MySQL Community Server - GPL

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> prompt 24060121140159>
PROMPT set to '24060121140159>'
24060121140159> create database pbo;
Query OK, 1 row affected (0.01 sec)

24060121140159> use pbo
Database changed
24060121140159> CREATE TABLE person(
    -> id INT PRIMARY KEY AUTO_INCREMENT NOT NULL,
    -> name VARCHAR(100));
Query OK, 0 rows affected (0.02 sec)

24060121140159> SELECT * FROM Person;
+----+-----+
| id | name |
+----+-----+
|  1 | Indra |
+----+-----+
1 row in set (0.01 sec)

24060121140159>
```

## B. Menggunakan Persistent Object sebagai Object Terealisasi

### 1. SerializePerson.java

```
/**
File : SerializePerson.java
Nama : Zaeri Haikal Rabbani
NIM : 24060121140159
Tanggal : 5 Juni 2023
Deskripsi : Program untuk serialisasi objek Person
**/

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

```

## 2. Compile dan jalankan

```

D:\Kulijing\java gamink\Pertemuan 9>javac SerializePerson.java

D:\Kulijing\java gamink\Pertemuan 9>java SerializePerson
selesai menulis objek person

D:\Kulijing\java gamink\Pertemuan 9>|

```

Program tersebut berhasil dijalankan dan mengeluarkan pesan selesai menulis objek person.

## 3. ReadSerializedPerson.java

```

/**
File : ReadSerializedPerson.java
Nama : Zaeri Haikal Rabbani
NIM : 24060121140159
Tanggal : 5 Mei 2023
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();  
        }  
    }  
}
```

#### 4. Compile dan Jalankan

```
D:\Kulijing\java gamink\Pertemuan 9>javac ReadSerializedPerson.java  
  
D:\Kulijing\java gamink\Pertemuan 9>java ReadSerializedPerson  
serialized person name = Panji
```

Dengan menjalankan perintah di atas, maka akan me-compile dan menjalankan program ReadSerializedPerson untuk membaca objek yang telah di-serialize sebelumnya. Program tersebut menampilkan objek yang telah di-serialize yaitu serialized person name = Panji.