

**LAPORAN PRAKTIKUM**  
**PRAKTIKUM 9:**  
**“PERSISTENT OBJECT”**



**Disusun Oleh :**

Aretha Khansa Maritza  
24060121140147

**PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK**  
**LAB B1**

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

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

### 1. PersonDAO.java

```
/*
 * Nama File: PersonDAO.java
 * Pembuat: Aretha Khansa Maritza
 * NIM: 24060121140147
 * Tanggal: 31 Mei 2023
 * Deskripsi: Interface untuk person access object
 * Lab: B1
 */

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

### 2. Person.java

```
/*
 * Nama File: Person.java
 * Pembuat: Aretha Khansa Maritza
 * NIM: 24060121140147
 * Tanggal: 31 Mei 2023
 * Deskripsi: Person database model
 * Lab: B1
 */

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

```
/*
 * Nama File: MySQLPersonDAO.java
 * Pembuat: Aretha Khansa Maritza
 * NIM: 24060121140147
 * Tanggal: 31 Mei 2023
 * Deskripsi: Implementasi PersonDAO untuk MySQL
 * Lab: B1
 */

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","Upin_Ipin_24");
        // 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

```
/*
 * Nama File: DAOManager.java
 * Pembuat: Aretha Khansa Maritza
 * NIM: 24060121140147
 * Tanggal: 31 Mei 2023
```

```

* Deskripsi: Pengelola DAO dalam program
* Lab: B1
*/

public class DAOManager{
    private PersonDAO personDAO;

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

```

## 5. MainDAO.java

```

/*
* Nama File: MainDAO.java
* Pembuat: Aretha Khansa Maritza
* NIM: 24060121140147
* Tanggal: 31 Mei 2023
* Deskripsi: Main program untuk akses DAO
* Lab: B1
*/

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. Pembuatan Database “pbo” dan Tabel “person” di MySQL Command Line Client

```
MySQL 8.0 Command Line Client
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.31 MySQL Community Server - GPL

Copyright (c) 2000, 2022, 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 Aretha>
PROMPT set to 'Aretha>'
Aretha> use pbo;
Database changed
Aretha> CREATE TABLE person(
  -> id INT PRIMARY KEY AUTO_INCREMENT NOT NULL,
  -> name VARCHAR(100));
Query OK, 0 rows affected (0.07 sec)

Aretha> select * from person;
Empty set (0.02 sec)
```

## 7. Melakukan compile code dan menjalankan MainDAO

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.2965]
(c) Microsoft Corporation. All rights reserved.

C:\Users\user-pc\Documents\Kuliah\Semester 4\Pemrograman Berorientasi Objek\Praktikum PBO\Praktikum 9>javac *.java
C:\Users\user-pc\Documents\Kuliah\Semester 4\Pemrograman Berorientasi Objek\Praktikum 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 automatically registered via the SPI and manual
loading of the driver class is generally unnecessary.
INSERT INTO person(name) VALUES ('Indra')
```

Berikut merupakan hasil pada MySQL Command Line Client:

```
Aretha> select * from person;
+----+-----+
| id | name |
+----+-----+
|  1 | Indra |
+----+-----+
1 row in set (0.01 sec)
```

## B. Menggunakan Persistent Object sebagai objek terserialisasi

### 1. SerializePerson.java

```
/*
 * Nama File: SerializePerson.java
 * Pembuat: Aretha Khansa Maritza
 * NIM: 24060121140147
 * Tanggal: 31 Mei 2023
```

```

/* Deskripsi: Program untuk serialisasi objek Person
 * Lab: B1
 */

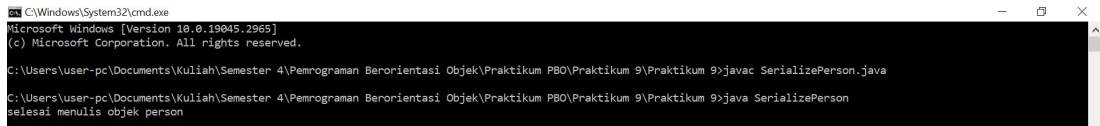
import java.io.*;
//class Person
class Person implements Serializable{
    private String name;
    public Person(String n){
        name = n;
    }

    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();
        }
    }
}

```

Hasil setelah dirun:



```

C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.2965]
(c) Microsoft Corporation. All rights reserved.

C:\Users\User-pc\Documents\Kuliah\Semester 4\Pemrograman Berorientasi Objek\Praktikum PBO\Praktikum 9>javac SerializePerson.java
C:\Users\User-pc\Documents\Kuliah\Semester 4\Pemrograman Berorientasi Objek\Praktikum PBO\Praktikum 9>java SerializePerson
selesai menulis objek person

```

## 2. ReadSerializePerson.java

```

/*
 * Nama File: ReadSerializedPerson.java
 * Pembuat: Aretha Khansa Maritza

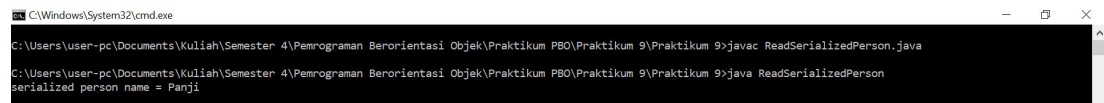
```

```
* NIM: 24060121140147
* Tanggal: 31 Mei 2023
* Deskripsi: Program untuk serialisasi objek Person
* Lab: B1
*/

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();
        }
    }
}
```

Hasil setelah dirun:



```
C:\Windows\System32\cmd.exe
C:\Users\user-pc\Documents\Kuliah\Semester 4\Pemrograman Berorientasi Objek\Praktikum PBO\Praktikum 9\Praktikum 9>javac ReadSerializedPerson.java
C:\Users\user-pc\Documents\Kuliah\Semester 4\Pemrograman Berorientasi Objek\Praktikum PBO\Praktikum 9\Praktikum 9>java ReadSerializedPerson
serialized person name = Panji
```