LAPORAN TUGAS

PEMROGRAMAN BERBASIS OBJEK

PERTEMUAN KEEMPAT 12 SEPTEMBER 2023

UIN SUNAN AMPEL
S U R A B A Y A

DOSEN PEMBIMBING

Bayu Adhi Nugroho, Ph.D.

(197905182014031001)

DISUSUN OLEH

Mochamad Roiyan Rintiarno

(09020622033)

UIN SUNAN AMPEL SURABAYA

TAHUN 2023

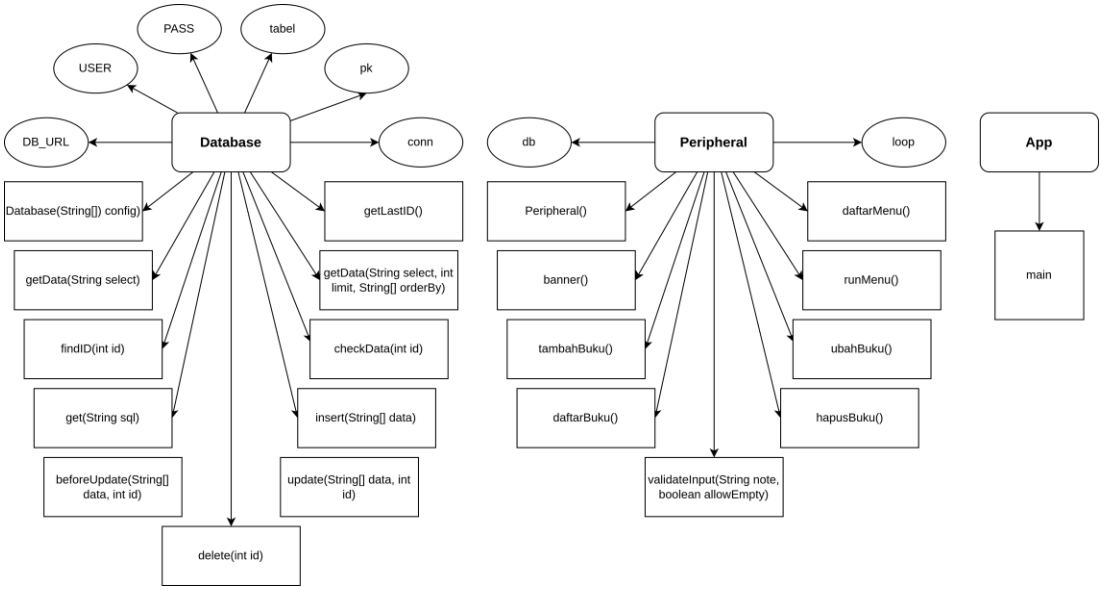
1. Tugas

→ Membuat CRUD sederhana menggunakan java dan PostgreSQL

2. Pembahasan dan Isi

a. Persiapan dan Langkah-langkah

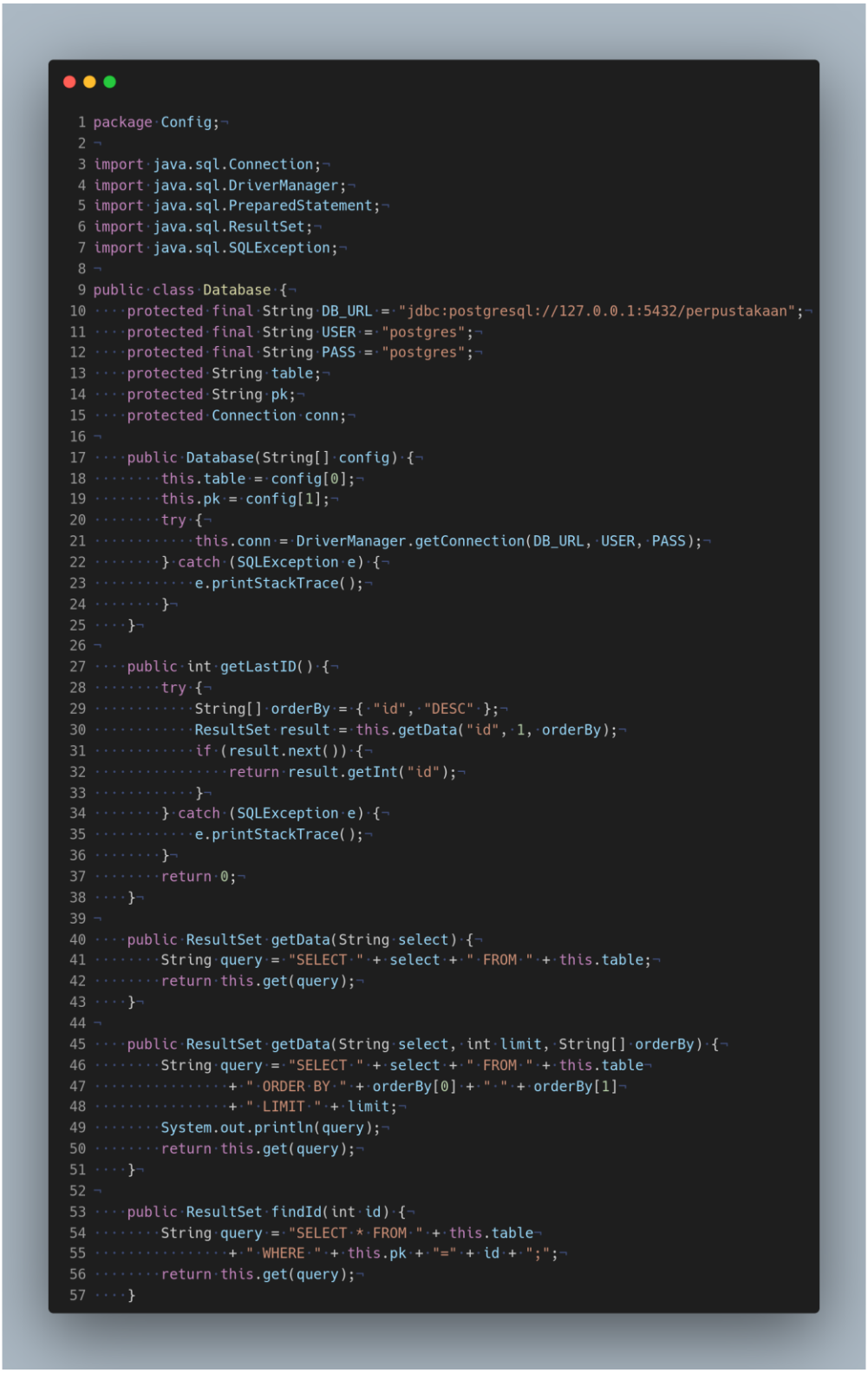
1. Membuat Diagram



2. Menyiapkan file Source Code

   Terdapat beberapa kelas yang nantinya akan kita buat sebagai kerangka, yakni kelas Database, sebagai koneksi menuju database, lalu kelas Peripheral, yang menghubungkan kelas App dengan kelas Database. Dan kelas App sendiri yang nantinya akan diberi method main untuk menjalankan program.

a. Kelas Database, untuk menghubungkan java denganPostgreSQL



```java
package Config;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class Database {
    protected final String DB_URL = "jdbc:postgresql://127.0.0.1:5432/perpustakaan";
    protected final String USER = "postgres";
    protected final String PASS = "postgres";
    protected String table;
    protected String pk;
    protected Connection conn;

    public Database(String[] config) {
        this.table = config[0];
        this.pk = config[1];
        try {
            this.conn = DriverManager.getConnection(DB_URL, USER, PASS);
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }

    public int getLastID() {
        try {
            String[] orderBy = { "id", "DESC" };
            ResultSet result = this.getData("id", 1, orderBy);
            if (result.next()) {
                return result.getInt("id");
            }
        } catch (SQLException e) {
            e.printStackTrace();
        }
        return 0;
    }

    public ResultSet getData(String select) {
        String query = "SELECT " + select + " FROM " + this.table;
        return this.get(query);
    }

    public ResultSet getData(String select, int limit, String[] orderBy) {
        String query = "SELECT " + select + " FROM " + this.table
                + " ORDER BY " + orderBy[0] + " " + orderBy[1]
                + " LIMIT " + limit;
        System.out.println(query);
        return this.get(query);
    }

    public ResultSet findId(int id) {
        String query = "SELECT * FROM " + this.table
                + " WHERE " + this.pk + "=" + id + ";";
        return this.get(query);
    }
}
```

```
58 ¬
59 ····public·boolean·checkData(int·id)·{¬
60 ········try·{¬
61 ············return·this.findId(id).next();¬
62 ········}·catch·(Exception·e)·{¬
63 ············e.printStackTrace();¬
64 ········}¬
65 ········return·false;¬
66 ····}¬
67 ¬
68 ····private·ResultSet·get(String·sql)·{¬
69 ········try·{¬
70 ············ResultSet·result·=·this.conn.createStatement().executeQuery(sql);¬
71 ············return·result;¬
72 ········}·catch·(Exception·e)·{¬
73 ············e.printStackTrace();¬
74 ········}¬
75 ········return·null;¬
76 ····}¬
77 ¬
78 ····public·boolean·insert(String[]·data)·{¬
79 ········try·{¬
80 ············String·query·=·"INSERT·INTO·"·+·this.table¬
81 ····················+·"(id,·name,·pengarang,·jenis)·VALUES·(?,·?,·?,·?);";¬
82 ············PreparedStatement·preparedStatement·=·this.conn.prepareStatement(query);¬
83 ············preparedStatement.setInt(1,·(this.getLastID()·+·1));¬
84 ············preparedStatement.setString(2,·data[0]);¬
85 ············preparedStatement.setString(3,·data[1]);¬
86 ············preparedStatement.setString(4,·data[2]);¬
87 ············if·(preparedStatement.executeUpdate()·<=·0)·{¬
88 ················return·false;¬
89 ············}¬
90 ········}·catch·(SQLException·e)·{¬
91 ············e.printStackTrace();¬
92 ········}¬
93 ········return·true;¬
94 ····}¬
95 ¬
96 ····private·String[]·beforeUpdate(String[]·data,·int·id)·{¬
97 ········try·{¬
98 ············String·query·=·"SELECT·*·FROM·"·+·this.table·+·"·WHERE·"·+·this.pk·+·"=?·LIMIT·1;";¬
99 ············PreparedStatement·preparedStatement·=·this.conn.prepareStatement(query);¬
100 ···········preparedStatement.setInt(1,·id);¬
101 ···········ResultSet·result·=·preparedStatement.executeQuery();¬
102 ···········if·(result.next())·{¬
103 ···············for·(int·i·=·0;·i·<·data.length;·i++)·{¬
104 ··················if·(data[i].length()·==·0)·{¬
105 ·····················data[i]·=·result.getString(i·+·2);¬
106 ··················}¬
107 ···············}¬
108 ···········}¬
109 ·······}·catch·(Exception·e)·{¬
110 ···········e.printStackTrace();¬
111 ·······}¬
112 ·······return·data;¬
113 ····}
```

```
114 ¬
115 ¬
116 ····public·boolean·update(String[]·data,·int·id)·{¬
117 ········data·=·this.beforeUpdate(data,·id);¬
118 ········try·{¬
119 ············String·query·=·"UPDATE·"·+·this.table·+·"·SET·name=?,·pengarang=?,·jenis=?·WHERE·"·+·
    this.pk·+·"=?;";¬
120 ············PreparedStatement·preparedStatement·=·this.conn.prepareStatement(query);¬
121 ············preparedStatement.setString(1,·data[0]);¬
122 ············preparedStatement.setString(2,·data[1]);¬
123 ············preparedStatement.setString(3,·data[2]);¬
124 ············preparedStatement.setInt(4,·id);¬
125 ············return·preparedStatement.executeUpdate()·>·0·?·true·:·false;¬
126 ········}·catch·(Exception·e)·{¬
127 ············e.printStackTrace();¬
128 ········}¬
129 ········return·false;¬
130 ····}¬
131 ¬
132 ····public·boolean·delete(int·id)·{¬
133 ········try·{¬
134 ············String·query·=·"DELETE·FROM·"·+·this.table·+·"·WHERE·"·+·this.pk·+·"=?;";¬
135 ············PreparedStatement·preparedStatement·=·this.conn.prepareStatement(query);¬
136 ············preparedStatement.setInt(1,·id);¬
137 ············return·preparedStatement.executeUpdate()·>·0·?·true·:·false;¬
138 ········}·catch·(SQLException·e)·{¬
139 ············e.printStackTrace();¬
140 ········}¬
141 ········return·false;¬
142 ····}¬
143 }
```

b. Kelas Peripheral, untuk menghubungkan antara database dengan class App

```java
package Config;

import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.Scanner;

public class Peripheral {
    private Database db;
    public boolean loop = true;

    public Peripheral() {
        String[] config = { "books", "id" };
        this.db = new Database(config);
    }

    public void banner() {
        System.out.println("================================================");
        System.out.println("================================================");
        System.out.println("==========  Selamat datang di Bukarta  ==========");
        System.out.println("================================================");
        System.out.println("================================================");
    }

    public void daftarMenu() {
        System.out.println("=Menu=");
        System.out.println("1. Tambahkan daftar");
        System.out.println("2. Tampilkan daftar");
        System.out.println("3. Ubah data");
        System.out.println("4. Hapus data buku");
        System.out.println("5. Exit");
    }

    public void runMenu() {
        Scanner scan = new Scanner(System.in);
        int count = 0;
        int input = 0;
        while (input < 1 || input > 5) {
            if (count > 0) {
                System.out.println("Masukkan tidak sesuai, coba kembali!");
            }
            System.out.print("Pilih menu: ");
            input = Integer.parseInt(scan.next());
            count++;
        }
        if (input == 1) {
            this.tambahBuku();
        } else if (input == 2) {
            this.loop = true;
            this.daftarBuku();
        } else if (input == 3) {
            this.ubahBuku();
        } else if (input == 4) {
            this.hapusBuku();
        } else {
            this.loop = false;
        }
    }
```

```
58 ¬
59 ····public·void·tambahBuku()·{¬
60 ········String[]·buku·=·new·String[3];¬
61 ········System.out.println("===Tambah·Buku===");¬
62 ········System.out.println("ID·buku\t\t\t:·"·+·(this.db.getLastID()·+·1));¬
63 ········buku[0]·=·this.validateInput("Masukkan·judul·buku\t:·",·false);¬
64 ········buku[1]·=·this.validateInput("Masukkan·pengarang·buku\t:·",·false);¬
65 ········buku[2]·=·this.validateInput("Masukkan·jenis·buku\t:·",·false);¬
66 ¬
67 ········System.out.println(this.db.insert(buku)¬
68 ····················?·"=>Berhasil·menambahkan·data!"¬
69 ····················:·"=>Gagal·menambah·data!");¬
70 ····}¬
71 ¬
72 ····public·void·ubahBuku()·{¬
73 ········System.out.println("===Ubah·Buku===");¬
74 ········String[]·buku·=·new·String[3];¬
75 ········int·id·=·Integer.parseInt(validateInput("Pilih·id·yang·akan·diubah:·",·false));¬
76 ········if·(!this.db.checkData(id))·{¬
77 ············System.out.println("=>Data·tidak·ditemukan!");¬
78 ········}·else·{¬
79 ············try·{¬
80 ················ResultSet·result·=·this.db.findId(id);¬
81 ················if·(result.next())·{¬
82 ····················System.out.println("Data·yang·akan·diubah:");¬
83 ····················System.out.println("id\t\t:·"·+·result.getString("id"));¬
84 ····················System.out.println("judul\t\t:·"·+·result.getString("name"));¬
85 ····················System.out.println("pengarang\t:·"·+·result.getString("pengarang"));¬
86 ····················System.out.println("jenis\t\t:·"·+·result.getString("jenis"));¬
87 ················}¬
88 ············}·catch·(SQLException·e)·{¬
89 ················e.printStackTrace();¬
90 ············}¬
91 ············System.out.println("=·>Biarkan·kosong·jika·tidak·ingin·merubah·data<·=");¬
92 ············buku[0]·=·this.validateInput("Masukkan·judul·baru\t:·",·true);¬
93 ············buku[1]·=·this.validateInput("Masukkan·pengarang·baru\t:·",·true);¬
94 ············buku[2]·=·this.validateInput("Masukkan·jenis·baru\t:·",·true);¬
95 ¬
96 ············System.out.println(this.db.update(buku,·id)¬
97 ····················?·"=>Berhasil·mengubah·data!"¬
98 ····················:·"=>Gagal·mengubah·data");¬
99 ········}¬
100 ····}¬
101 ¬
102 ····public·void·daftarBuku()·{¬
103 ········System.out.println("===Daftar·Buku===");¬
104 ········try·{¬
105 ············System.out.println("ID"·+·"\t"·+·"Judul·Buku"·+·"\t\t"·+·"Pengarang"·+·"\t"¬
106 ····················+·"Jenis·buku");¬
107 ¬
108 ············ResultSet·result·=·this.db.getData("*");¬
109 ············while·(result.next())·{¬
110 ················System.out.println(String.valueOf(result.getObject(1))·+·"\t"·+¬
111 ····························String.valueOf(result.getObject(2))·+·"\t"·+¬
112 ····························String.valueOf(result.getObject(3))·+·"\t"·+¬
113 ····························String.valueOf(result.getObject(4)));¬
114 ············}¬
115 ········}·catch·(SQLException·e)·{¬
116 ············e.printStackTrace();¬
117 ········}¬
118 ····}
```

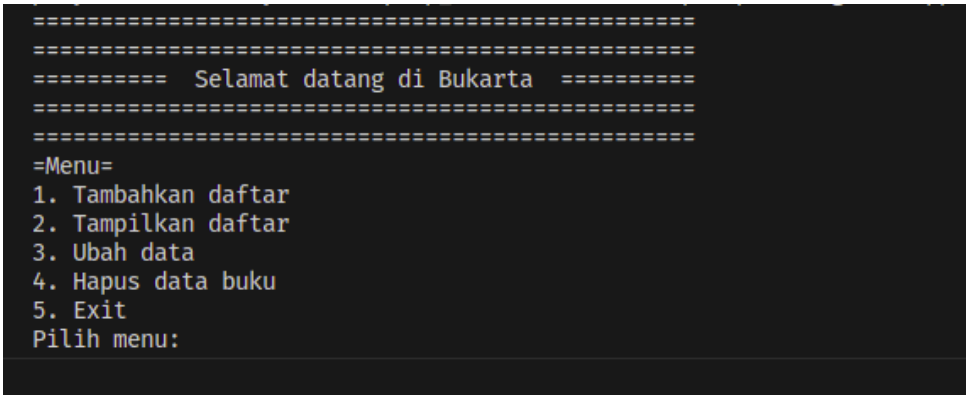
```
119 ¬
120 ····public·void·hapusBuku()·{¬
121 ········int·id·=·Integer.parseInt(this.validateInput("Masukkan·id·yang·ingin·duhapus:·",·false));¬
122 ········if·(!this.db.checkData(id))·{¬
123 ············System.out.println("=>Data·tidak·ditemukan!");¬
124 ········}·else·{¬
125 ············System.out.println(this.db.delete(id)¬
126 ····················?·"=>Data·berhasil·dihapus!"¬
127 ····················:·"=>Data·gagal·dihapus!");¬
128 ········}¬
129 ¬
130 ····}¬
131 ¬
132 ····private·String·validateInput(String·note,·boolean·allowEmpty)·{¬
133 ········System.out.print(note);¬
134 ········Scanner·scan·=·new·Scanner(System.in);¬
135 ········String·input·=·"";¬
136 ········int·count·=·0;¬
137 ········if·(allowEmpty)·{¬
138 ············input·=·scan.nextLine();¬
139 ········}·else·{¬
140 ············while·(input.equals(""))·{¬
141 ················input·=·scan.nextLine();¬
142 ················if·(count·>·0)·{¬
143 ····················System.out.println("Masukkan·tidak·sesuai,·coba·kembali!");¬
144 ················}¬
145 ················count++;¬
146 ············}¬
147 ········}¬
148 ········return·input;¬
149 ····}¬
150 }
```

c. Kelas App, sebagai tempat method main

```java
import Config.Peripheral;

public class App {
    public static void main(String[] args) {
        Peripheral peripheral = new Peripheral();
        peripheral.banner();
        while (peripheral.loop) {
            peripheral.daftarMenu();
            peripheral.runMenu();
        }
    }
}
```

3. Output
    a. Tampilan Awal



    b. Ketika menambahkan data

```
==================================================
==================================================
==========  Selamat datang di Bukarta  ==========
==================================================
==================================================
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu: 1
===Tambah Buku===
SELECT id FROM books ORDER BY id DESC LIMIT 1
ID buku                 : 5
Masukkan judul buku     : Bila kau tak disampingku
Masukkan pengarang buku : Sheila on 7
Masukkan jenis buku     : Lirik lagu
SELECT id FROM books ORDER BY id DESC LIMIT 1
=>Berhasil menambahkan data!
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu:
```

c. Ketika menampilkan data

```
==================================================
==================================================
==========  Selamat datang di Bukarta  ==========
==================================================
==================================================
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu: 2
===Daftar Buku===
ID      Judul Buku              Pengarang       Jenis buku
1       bandung lautan api      mpu tantular    dongeng
2       bandung lautan air      mpu gandring    fiksi
4       Mereka tak pernah mengerti      Tipe X  nostalgia
3       Pemuja Rahasia  Sheila on 7     Sheilagank
5       Bila kau tak disampingku        Sheila on 7     Lirik lagu
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu:
```

d. Ketika mengubah data

```
==================================================
==================================================
==========  Selamat datang di Bukarta  ==========
==================================================
==================================================
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu: 3
===Ubah Buku===
Pilih id yang akan diubah: 3
Data yang akan diubah:
id              : 3
judul           : Pemuja Rahasia
pengarang       : Sheila on 7
jenis           : Sheilagank
= >Biarkan kosong jika tidak ingin merubah data< =
Masukkan judul baru     :
Masukkan pengarang baru :
Masukkan jenis baru     : Lirik lagu
=>Berhasil mengubah data!
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu:
```

e. Ketika menghapus data

```
==================================================
==================================================
==========  Selamat datang di Bukarta  ==========
==================================================
==================================================
=Menu=
1. Tambahkan daftar
2. Tampilkan daftar
3. Ubah data
4. Hapus data buku
5. Exit
Pilih menu: 5
```