Nama : Muhammad Dandy Chrisnandy

Kelas : XI RPL 2

**Tutorial Database Java Dengan Eclipse**

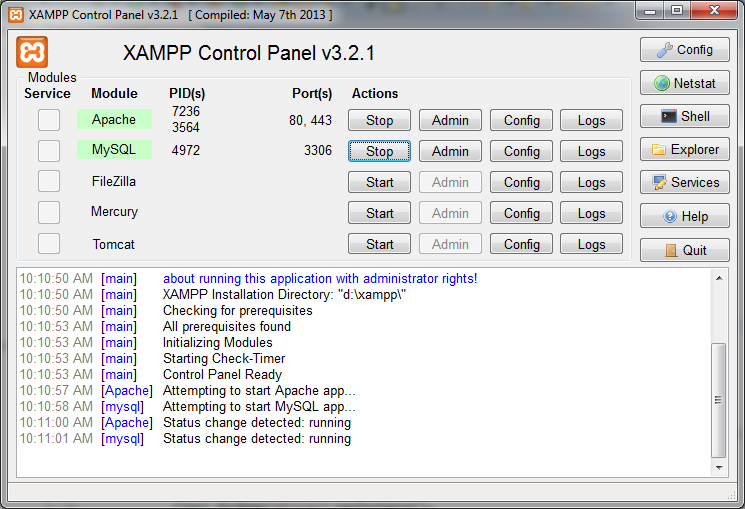
**Membuat Koneksi Eclipse ke Database Mysql (XAMPP)**

Sebelum memulainnya kita harus menginstall terlebih dahulu jdk dan Xampp.

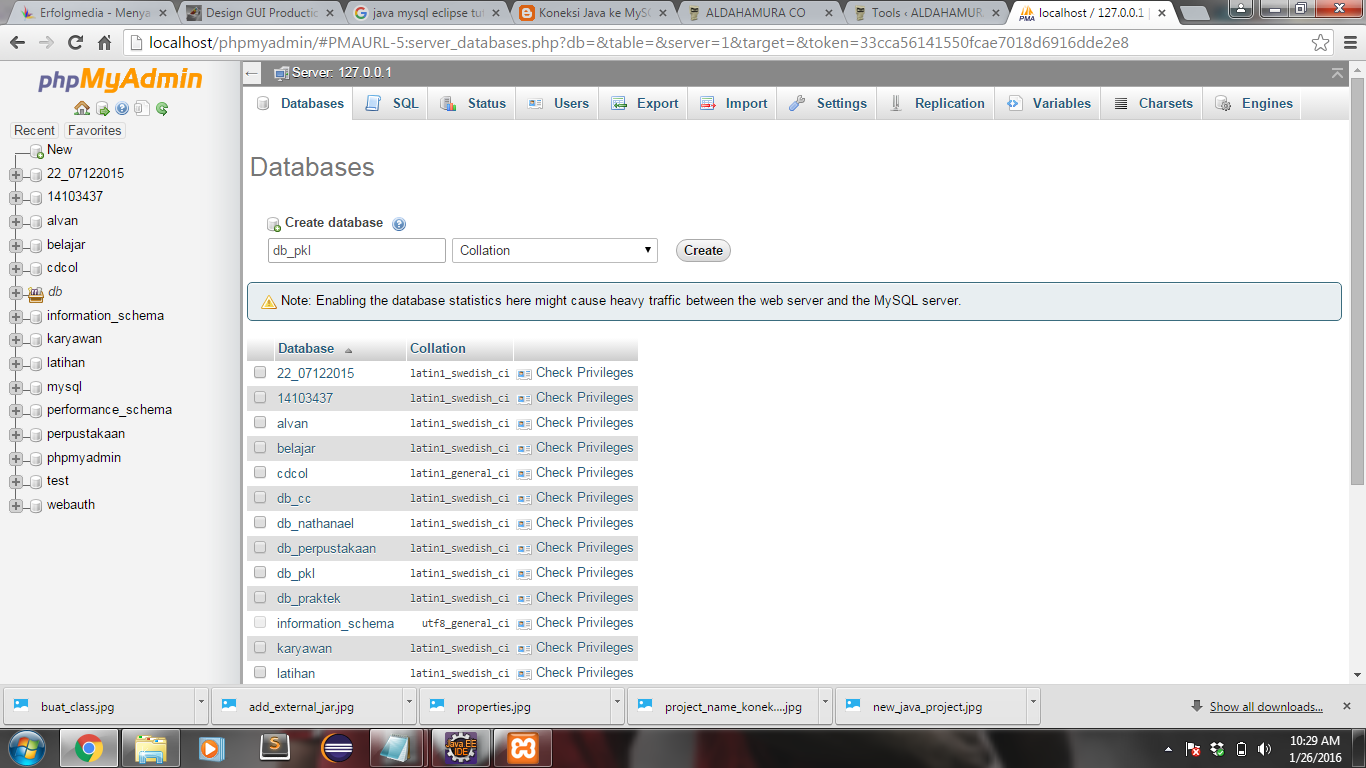
Setelah itu, downloadlah library berikut :

-mysql-connector-java-5.1.7-bin.jar.

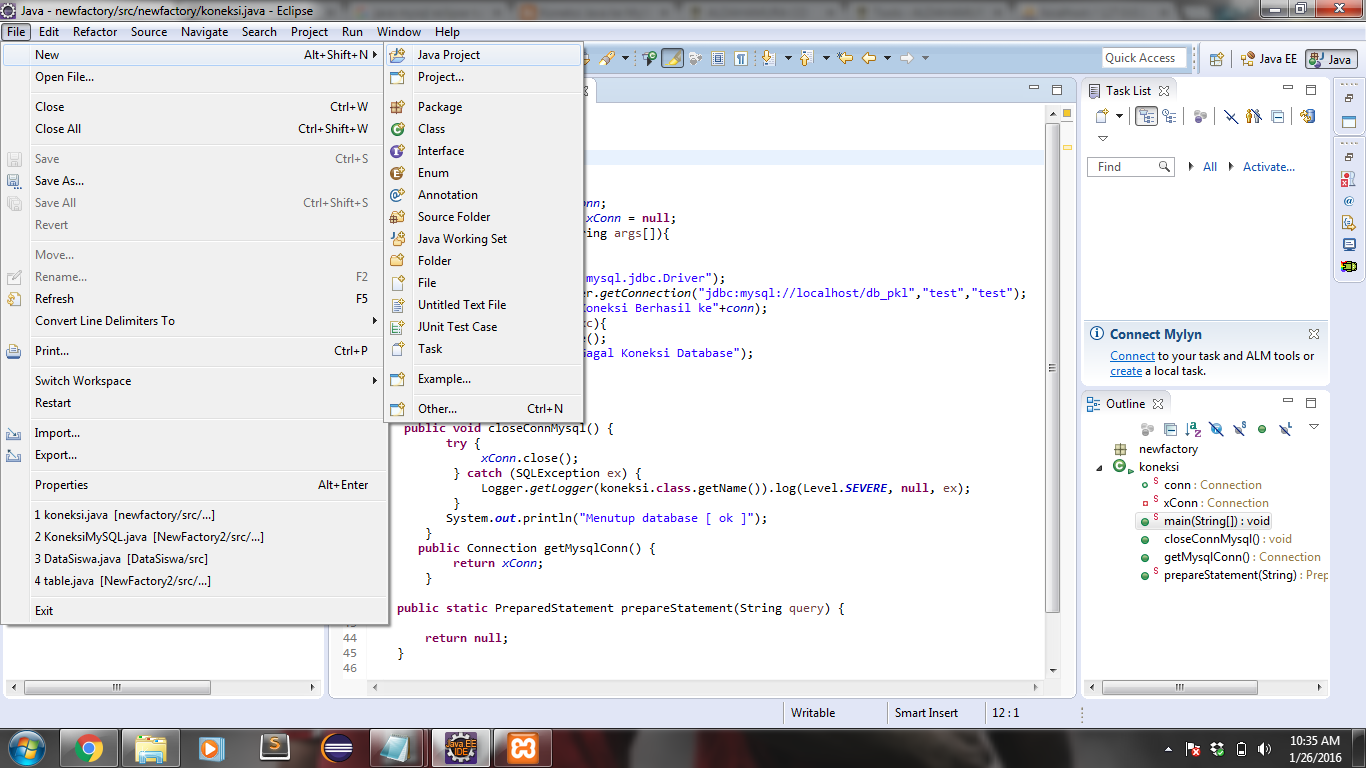
Jika sudah download Xampp,Nyalakan service Apache dan MySQL yang terdapat pada XAMPP. Seperti gambar berikut :

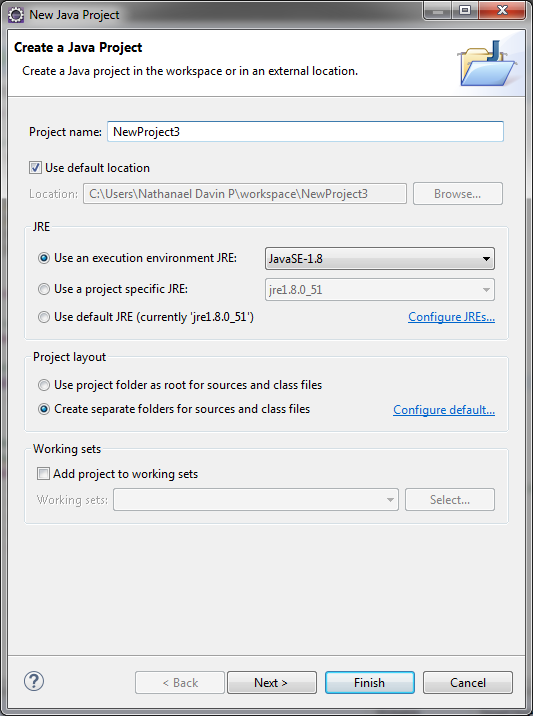


Setelah itu kita buat database MySQL, sebagai contoh : db\_pkl



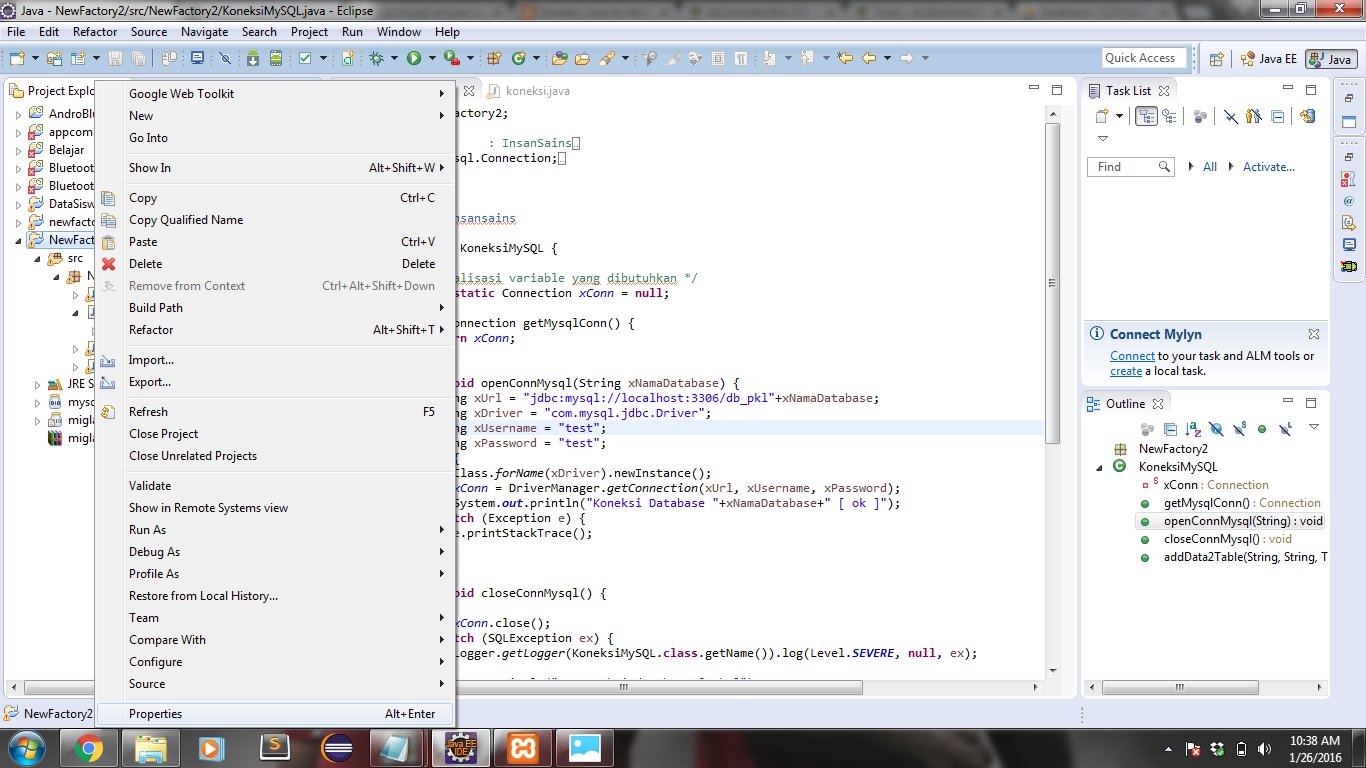
Buka eclipse, Buat New Project java –  Project namenya contohnya: **“NewProject3”**



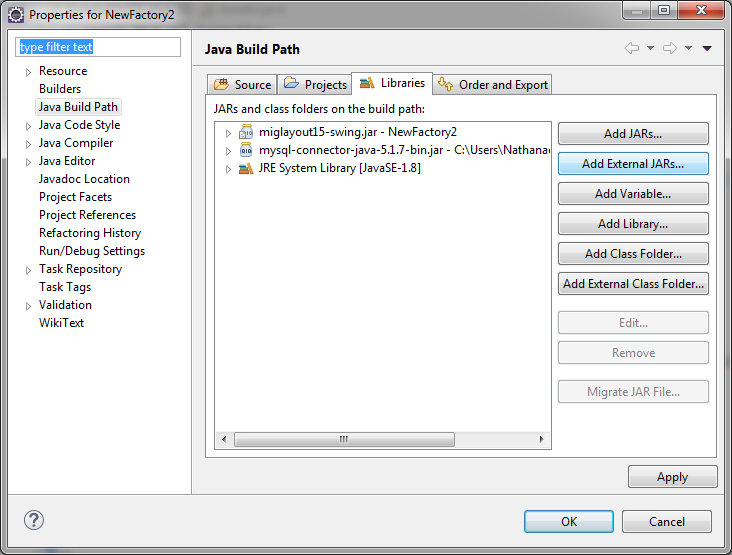


Kemudian klik finish, setelah itu kita koneksikan **mysql-connector-java-5.1.7-bin.jar**ke eclipse.

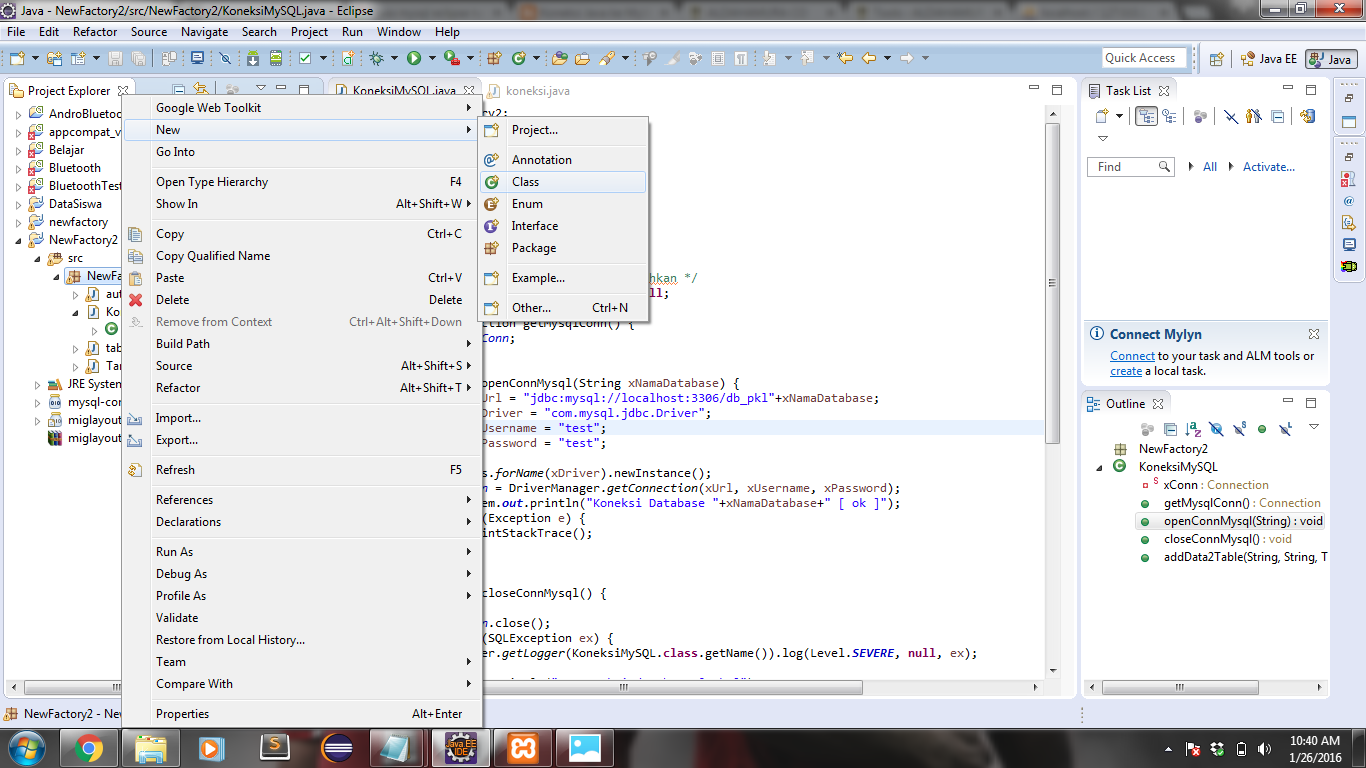
klik kanan pada project yang telah di buat dan pilih properties :



Kemudian pilih Java Build Path – Libraries – Add External JARs…Tambahkan **mysql-connector-java-5.1.7-bin.jar :**



lalu buat class baru di project **“NewProject3”** dan beri nama classnya **“koneksi” :**



Lalu kita ketikan source code bertikut di class **“koneksi”**

package NewProject3;

/\* import libraries yang dibutuhkan \*/

import javax.swing.\*;

import java.sql.\*;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import com.mysql.jdbc.PreparedStatement;

public class koneksi {

//variabel yang di butuhkan

public static Connection conn;

private static Connection xConn = null;

// Method untuk membuka koneksi dengan database MySQL

public static void main(String args[]){

try{

Class.forName("com.mysql.jdbc.Driver");

conn = DriverManager.getConnection("jdbc:mysql://localhost/db\_pkl","test","test");//database,username,password disesuaikan

System.out.print("Koneksi Berhasil ke"+conn);

}catch(Exception exc){

exc.printStackTrace();

System.out.print("Gagal Koneksi Database");

}

}

// Method untuk menutup koneksi dengan database MySQL

public void closeConnMysql() {

try {

xConn.close();

} catch (SQLException ex) {

Logger.getLogger(koneksi.class.getName()).log(Level.SEVERE, null, ex);

}

System.out.println("Menutup database [ ok ]");

}

//Method untuk me-return nilai (variable) xConn

public Connection getMysqlConn() {

return xConn;

}

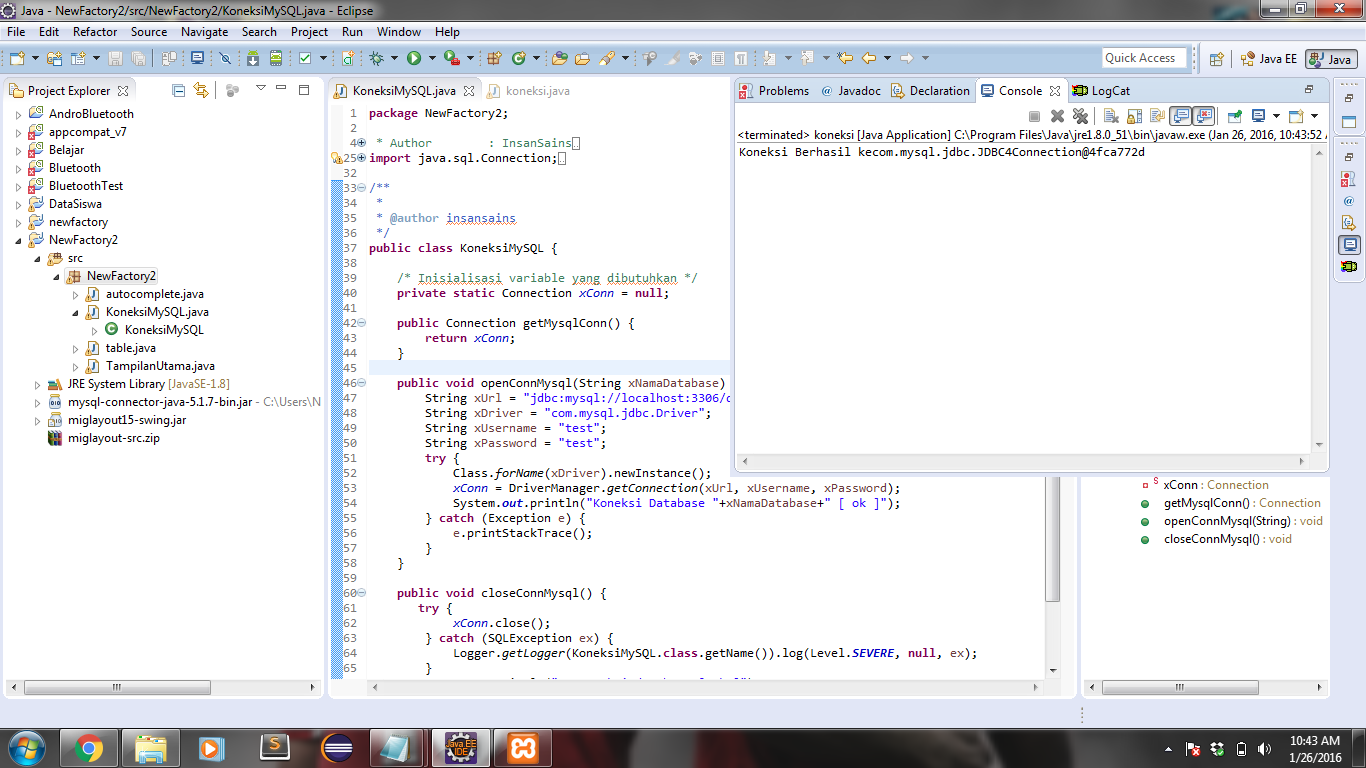
public static PreparedStatement prepareStatement(String query) {

return null;

}

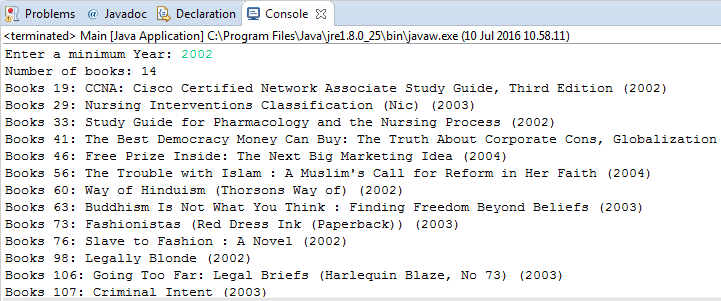
}

Jika keluar Pesan seperti ini Maka Koneksi ke Database Sukses



**Tutorial Filter Data Dari Database PhpMyadmin dengan Input Statement Java Eclipse**

Nah pada tutorial kali ini kita akan menambahkan class input untuk memfilter data yang akan di tampilkan dari PhpMyadmin. Fungsi class input ini nantinya yang akan menciptakan object dari kriteria jenis input yang diinginkan.



Pada kasus ini kita menggunakan data buku yang kita ambil dari data SQL berikut: [**Download SQL**](https://drive.google.com/file/d/0BwLJnvE1KIpJM3B2OVMtMFR3b1U/view?usp=sharing)

Dari tabel buku tersebut kita akan coba tampilkan id buku, judul , serta tahun terbit buku yang akan kita filter berdasarkan minimum tahun terbitnya. Jadi, buku-buku yang ditampilkan adalah buku-buku dengan minimum tahun terbit tersebut dan buku-buku dengan terbitan tahun diatasnya. Karena disini kita menggunakan query “SELECT id, title, year\_of\_publication FROM books WHERE year\_of\_publication >= ? ”

WHERE disini yang akan memfilter bahwa yang ditampilkan adalah buku-buku dengan tahun terbit >= tahun terbit yang diinginkan.

Berikut source code java dari 5 kelas dan 3 package yang akan kita buat.

1. DBType

package com.dantikpuspita.db;

public enum DBType {

MYSQL

}

1. DBUtil

package com.dantikpuspita.db;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DBUtil {

private static final String USERNAME = "dbuser";

private static final String PASSWORD = "dbpassword";

private static final String M\_CONN\_STRING =

"jdbc:mysql://localhost/senayandb";

public static Connection getConnection(DBType dbType) throws SQLException {

switch (dbType) {

case MYSQL:

return DriverManager.getConnection(M\_CONN\_STRING, USERNAME, PASSWORD);

default:

return null;

}

}

public static void processException(SQLException e) {

System.err.println("Error message: " + e.getMessage());

System.err.println("Error code: " + e.getErrorCode());

System.err.println("SQL state: " + e.getSQLState());

}

}

1. Main

package com.dantikpuspita.db;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import com.dantikpuspita.db.table.Book;

import com.dantikpuspita.input.inputHelper;

import com.mysql.jdbc.PreparedStatement;

public class Main {

private static final String SQL =

"SELECT id, title, year\_of\_publication FROM books WHERE year\_of\_publication >= ?";

public static void main(String[] args) throws Exception {

int minYear;

try {

minYear = (int) inputHelper.getDoubleInput("Enter a minimum Year: ");

} catch (Exception e) {

System.err.println("Error: invalid number");

return;

}

ResultSet rs = null;

try (

Connection conn = DBUtil.getConnection(DBType.MYSQL);

PreparedStatement stmt = (PreparedStatement) conn.prepareStatement(

SQL,

ResultSet.TYPE\_SCROLL\_INSENSITIVE,

ResultSet.CONCUR\_READ\_ONLY);

) {

stmt.setInt(1, minYear);

rs = stmt.executeQuery();

Book.displayData(rs);

} catch (SQLException e) {

System.err.println(e);

}

finally {

if (rs != null) rs.close();

}

}

}

1. Book

package com.dantikpuspita.db.table;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.text.NumberFormat;

public class Book {

public static void displayData(ResultSet rs) throws SQLException {

rs.last();

int nRows = rs.getRow();

if (nRows == 0) {

System.out.println("No books were found");

}

else {

System.out.println("Number of books: " + nRows);

rs.beforeFirst();

while (rs.next()) {

StringBuffer buffer = new StringBuffer();

buffer.append("Books " + rs.getInt("id") + ": ");

buffer.append(rs.getString("title"));

int year = rs.getInt("year\_of\_publication");

buffer.append(" (" + year +")");

System.out.println(buffer.toString());

}

}

}

}

1. Input Helper

package com.dantikpuspita.input;

import java.io.BufferedReader;

import java.io.InputStreamReader;

public class inputHelper {

public static String getInput(String prompt){

BufferedReader stdin = new BufferedReader(

new InputStreamReader(System.in));

System.out.print(prompt);

System.out.flush();

try {

return stdin.readLine();

} catch (Exception e) {

return "Error: "+ e.getMessage();

}

}

public static double getDoubleInput(String prompt)throws NumberFormatException{

String input = getInput(prompt);

return Double.parseDouble(input);

}

}