**LAPORAN TUGAS**

**PEMOGRAMAN BERORIENTSI OBJEK**

**Nama : Muhammad Naufal Arhab**

**Nim : 5190411053**

1. **Source Code Program**

import mysql.connector

*#  pip install mysql-connector-python*

class Buku:

    def \_\_init\_\_(self, judul, pengarang, isbn):

*self*.judul = judul

*self*.pengarang = pengarang

*self*.isbn = isbn

    def tampilkan\_info(self):

        print(f"Judul: {*self*.judul}, Pengarang: {*self*.pengarang}, ISBN: {*self*.isbn}")

class Perpustakaan:

    def \_\_init\_\_(self):

*self*.koneksi = mysql.connector.connect(

            host="localhost",

            user="root",

            password="",

        )

*self*.kursor = *self*.koneksi.cursor()

*self*.kursor.execute("CREATE DATABASE IF NOT EXISTS `5190411053`")

*self*.kursor.execute("USE `5190411053`")

*self*.buat\_tabel()

    def buat\_tabel(self):

        query = """CREATE TABLE IF NOT EXISTS buku (

                        id INT AUTO\_INCREMENT PRIMARY KEY,

                        judul VARCHAR(255),

                        pengarang VARCHAR(255),

                        isbn VARCHAR(20)

                    )"""

*self*.kursor.execute(query)

*self*.koneksi.commit()

    def tambah\_buku(self, buku):

        query = "INSERT INTO buku (judul, pengarang, isbn) VALUES (%s, %s, %s)"

        nilai = (buku.judul, buku.pengarang, buku.isbn)

*self*.kursor.execute(query, nilai)

*self*.koneksi.commit()

        print("Buku berhasil ditambahkan!")

    def tampilkan\_daftar\_buku(self):

        query = "SELECT \* FROM buku"

*self*.kursor.execute(query)

        hasil = *self*.kursor.fetchall()

        for baris in hasil:

            print("=="\*15)

            print(f"ID: {baris[0]}\nJudul: {baris[1]}\nPengarang: {baris[2]}\nISBN: {baris[3]}")

            print("=="\*15)

    def hapus\_buku(self, id\_buku):

        query = "DELETE FROM buku WHERE id = %s"

        nilai = (id\_buku,)

*self*.kursor.execute(query, nilai)

*self*.koneksi.commit()

        print(f"Buku dengan ID {id\_buku} berhasil dihapus.")

    def update\_buku(self, id\_buku, judul\_baru, pengarang\_baru, isbn\_baru):

        query = "UPDATE buku SET judul=%s, pengarang=%s, isbn=%s WHERE id=%s"

        nilai = (judul\_baru, pengarang\_baru, isbn\_baru, id\_buku)

*self*.kursor.execute(query, nilai)

*self*.koneksi.commit()

        print(f"Informasi buku dengan ID {id\_buku} berhasil diperbarui.")

perpustakaan = Perpustakaan()

while True:

    print("1.tampilkan buku")

    print("2.tambah buku")

    print("3.hapus buku")

    print("4.edit data buku")

    menu = input("menu :")

    if menu =="1":

        perpustakaan.tampilkan\_daftar\_buku()

    elif menu == "2":

        perpustakaan.tambah\_buku(Buku(input("judul buku :"),input("nama pengarang :"),input("isbn buku :")))

    elif menu == "3":

        perpustakaan.tampilkan\_daftar\_buku()

        perpustakaan.hapus\_buku(input("masukan id buku :"))

    elif menu =="4":

        perpustakaan.tampilkan\_daftar\_buku()

        perpustakaan.update\_buku(input("id buku :"),input("judul buku :"),input("nama pengarang :"),input("isbn buku :"))

1. **Penjelasan Program**

Program di atas adalah implementasi sederhana dari sebuah sistem manajemen perpustakaan menggunakan Python dan MySQL. Berikut penjelasan singkat untuk setiap bagian dari program:

1. Kelas Buku:

* Mewakili objek buku dengan atribut judul, pengarang, dan isbn.
* Metode tampilkan\_info digunakan untuk menampilkan informasi buku.

1. Kelas Perpustakaan:

* Membuat koneksi ke server MySQL dan mengatur kursor.
* Membuat database dengan nama 5190411053 jika belum ada.
* Membuat tabel buku dengan kolom id (kunci utama), judul, pengarang, dan isbn jika belum ada.
* Metode tambah\_buku untuk menambahkan buku ke database.
* Metode tampilkan\_daftar\_buku untuk menampilkan seluruh buku yang ada di database.
* Metode hapus\_buku untuk menghapus buku berdasarkan ID.
* Metode update\_buku untuk memperbarui informasi buku berdasarkan ID.

1. Loop Utama:

* Membuat objek perpustakaan dari kelas Perpustakaan.
* Menampilkan menu utama dalam sebuah loop tak terbatas.
* Menerima input dari pengguna untuk memilih menu.
* Menjalankan fungsi terkait sesuai dengan pilihan pengguna.

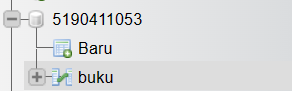
1. Menu Utama:

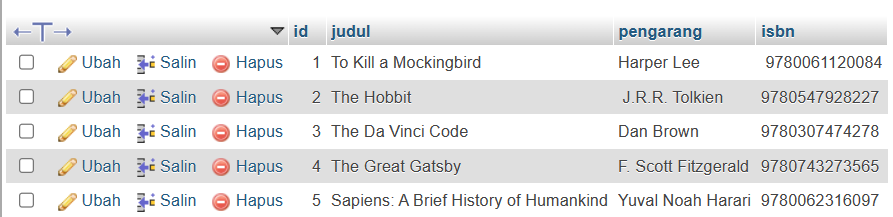
* Pilihan 1: Menampilkan daftar buku.
* Pilihan 2: Menambahkan buku baru dengan input dari pengguna.
* Pilihan 3: Menampilkan daftar buku dan meminta ID untuk menghapus buku.
* Pilihan 4: Menampilkan daftar buku, meminta ID, dan memperbarui informasi buku.

1. Penggunaan MySQL:
   * Program menggunakan modul mysql.connector untuk berinteraksi dengan server MySQL.
   * Setiap kali ada perubahan dalam database (menambah, menghapus, memperbarui), perubahan tersebut di-commit untuk menyimpan perubahan ke database.
2. Catatan:

* Pastikan MySQL server berjalan dan dapat diakses di localhost.
* Perhatikan bahwa password MySQL dalam contoh ini tidak diatur. Secara umum, sebaiknya hindari menggunakan password kosong untuk keamanan.gunakan password yang kuat untuk keamanan.

1. **Database**





1. **Hasil Running**
2. Menampilkan Data Buku



1. Menghapus Buku



1. Select untuk mengedit Data Buku

