**LAPORAN TUGAS PROJECT**

**PEMOGRAMAN BERORIENTASI OBJEK PRAKTIKUM**

**VII**

**Dosen Pengampu : Ibu Rodhiyah Mardhiyyah, S.Kom., M.Kom.**



**Disusun Oleh :**

**Nama : Amallia Az Zahra**

**Npm : 5220411239**

**UNIVERSITAS TEKNOLOGI YOGYAKARTA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**PRODI TEKNIK INFORMATIKA**

**YOGYAKARTA**

**2024**

1. **Source code program**

import mysql.connector

# Kelas untuk Database (CRUD)

class Database:

    def \_\_init\_\_(self):

        self.connection = mysql.connector.connect(

            host="localhost",

            user="root",

            password="",

            database="5220411239"

        )

        self.cursor = self.connection.cursor()

        self.create\_tables()

    def create\_tables(self):

        # Tabel untuk PertanianUmum

        self.cursor.execute("CREATE TABLE IF NOT EXISTS pertanian (id INT AUTO\_INCREMENT PRIMARY KEY, luas\_lahan VARCHAR(255), jumlah\_tanaman INT, jenis\_pupuk VARCHAR(255))")

        # Tabel untuk PertanianSayuran

        self.cursor.execute("CREATE TABLE IF NOT EXISTS pertanian\_sayuran (id INT AUTO\_INCREMENT PRIMARY KEY, jenis\_sayuran VARCHAR(255), pertanian\_id INT, FOREIGN KEY (pertanian\_id) REFERENCES pertanian(id))")

        # Tabel untuk PertanianBuah

        self.cursor.execute("CREATE TABLE IF NOT EXISTS pertanian\_buah (id INT AUTO\_INCREMENT PRIMARY KEY, jenis\_buah VARCHAR(255), pertanian\_id INT, FOREIGN KEY (pertanian\_id) REFERENCES pertanian(id))")

        self.connection.commit()

    def tambah\_data\_pertanian(self, pertanian):

        if isinstance(pertanian, Pertanian):

            self.cursor.execute("INSERT INTO pertanian (luas\_lahan, jumlah\_tanaman, jenis\_pupuk) VALUES (%s, %s, %s)",

                                (pertanian.luas\_lahan, pertanian.jumlah\_tanaman, pertanian.jenis\_pupuk))

            self.connection.commit()

            pertanian\_id = self.cursor.lastrowid

            if isinstance(pertanian, PertanianSayuran):

                self.cursor.execute("INSERT INTO pertanian\_sayuran (jenis\_sayuran, pertanian\_id) VALUES (%s, %s)",

                                    (pertanian.jenis\_sayuran, pertanian\_id))

            elif isinstance(pertanian, PertanianBuah):

                self.cursor.execute("INSERT INTO pertanian\_buah (jenis\_buah, pertanian\_id) VALUES (%s, %s)",

                                    (pertanian.jenis\_buah, pertanian\_id))

            self.connection.commit()

            print("Data pertanian berhasil ditambahkan.")

        else:

            print("Objek yang ditambahkan bukan instance dari kelas Pertanian.")

    def tampilkan\_data\_pertanian(self):

        self.cursor.execute(

            "SELECT pertanian.id, pertanian.luas\_lahan, pertanian.jumlah\_tanaman, pertanian.jenis\_pupuk, pertanian\_sayuran.jenis\_sayuran, pertanian\_buah.jenis\_buah FROM pertanian LEFT JOIN pertanian\_sayuran ON pertanian.id = pertanian\_sayuran.pertanian\_id LEFT JOIN pertanian\_buah ON pertanian.id = pertanian\_buah.pertanian\_id")

        data = self.cursor.fetchall()

        if not data:

            print("Tidak ada data pertanian.")

        else:

            for row in data:

                print(f"ID: {row[0]}, Luas Lahan: {row[1]}, Jumlah Tanaman: {row[2]}, Jenis Pupuk: {row[3]}")

                if row[4]:

                    print(f"Jenis Sayuran: {row[4]}")

                elif row[5]:

                    print(f"Jenis Buah: {row[5]}")

    def update\_data\_pertanian(self, id, luas\_lahan, jumlah\_tanaman, jenis\_pupuk, jenis\_tanaman):

        self.cursor.execute("UPDATE pertanian SET luas\_lahan = %s, jumlah\_tanaman = %s, jenis\_pupuk = %s WHERE id = %s",

                            (luas\_lahan, jumlah\_tanaman, jenis\_pupuk, id))

        if jenis\_tanaman == 'sayuran':

            self.cursor.execute("UPDATE pertanian\_sayuran SET jenis\_sayuran = %s WHERE pertanian\_id = %s",

                                (jenis\_tanaman, id))

        elif jenis\_tanaman == 'buah':

            self.cursor.execute("UPDATE pertanian\_buah SET jenis\_buah = %s WHERE pertanian\_id = %s",

                                (jenis\_tanaman, id))

        self.connection.commit()

        print("Data pertanian berhasil diupdate.")

    def hapus\_data\_pertanian(self, id):

        self.cursor.execute("DELETE FROM pertanian WHERE id = %s", (id,))

        self.connection.commit()

        print("Data pertanian berhasil dihapus.")

    def \_\_del\_\_(self):

        try:

            if self.cursor:

                self.cursor.close()

            if self.connection:

                self.connection.close()

        except AttributeError:

            pass

# Kelas Utama untuk Pertanian

class Pertanian:

    def \_\_init\_\_(self, luas\_lahan, jumlah\_tanaman, jenis\_pupuk):

        self.luas\_lahan = luas\_lahan

        self.jumlah\_tanaman = jumlah\_tanaman

        self.jenis\_pupuk = jenis\_pupuk

    def get\_info(self):

        return f"Lahan:{self.luas\_lahan}, Tanaman:{self.jumlah\_tanaman}, Pupuk:{self.jenis\_pupuk}"

class PertanianSayuran(Pertanian):

    def \_\_init\_\_(self, luas\_lahan, jumlah\_tanaman, jenis\_pupuk, jenis\_sayuran):

        super().\_\_init\_\_(luas\_lahan, jumlah\_tanaman, jenis\_pupuk)

        self.jenis\_sayuran = jenis\_sayuran

    def tampilkan\_info(self):

        print(

            f"Pertanian sayuran dengan luas lahan {self.luas\_lahan}, menggunakan pupuk {self.jenis\_pupuk}, dan menanam {self.jumlah\_tanaman} tanaman {self.jenis\_sayuran}.")

class PertanianBuah(Pertanian):

    def \_\_init\_\_(self, luas\_lahan, jumlah\_tanaman, jenis\_pupuk, jenis\_buah):

        super().\_\_init\_\_(luas\_lahan, jumlah\_tanaman, jenis\_pupuk)

        self.jenis\_buah = jenis\_buah

    def tampilkan\_info(self):

        print(

            f"Pertanian buah dengan luas lahan {self.luas\_lahan}, menggunakan pupuk {self.jenis\_pupuk}, dan menanam {self.jumlah\_tanaman} tanaman {self.jenis\_buah}.")

# Buat subclass PertanianBuah yang mewarisi Pertanian

class PertanianBuah(Pertanian):

    def \_\_init\_\_(self, luas\_lahan, jumlah\_tanaman, jenis\_pupuk, jenis\_buah):

        super().\_\_init\_\_(luas\_lahan, jumlah\_tanaman, jenis\_pupuk)

        self.jenis\_buah = jenis\_buah

    def tampilkan\_info(self):

        print(

            f"Pertanian buah dengan luas lahan {self.luas\_lahan}, menggunakan pupuk {self.jenis\_pupuk}, dan menanam {self.jumlah\_tanaman} tanaman {self.jenis\_buah}.")

# Fungsi untuk menampilkan menu

def menu():

    print("\n===== Menu =====")

    print("1. Tambah Data Pertanian")

    print("2. Tampilkan Data Pertanian")

    print("3. Update Data Pertanian")

    print("4. Hapus Data Pertanian")

    print("5. Keluar")

db = Database()

while True:

    menu()

    choice = input("Pilih menu (1-5): ")

    if choice == '1':

        print("\n===== Jenis Pertanian =====")

        print("1. Pertanian Umum")

        print("2. Pertanian Sayuran")

        print("3. Pertanian Buah")

        jenis\_pertanian = input("Pilih jenis pertanian [1-3]: ")

        luas\_lahan = input("Masukkan luas lahan: ")

        jumlah\_tanaman = input("Masukkan jumlah tanaman: ")

        jenis\_pupuk = input("Masukkan jenis pupuk: ")

        if jenis\_pertanian == '1':

            pertanian = Pertanian(luas\_lahan, jumlah\_tanaman, jenis\_pupuk)

        elif jenis\_pertanian == '2':

            jenis\_sayuran = input("Masukkan jenis sayuran: ")

            pertanian = PertanianSayuran(luas\_lahan, jumlah\_tanaman, jenis\_pupuk, jenis\_sayuran)

        elif jenis\_pertanian == '3':

            jenis\_buah = input("Masukkan jenis buah: ")

            pertanian = PertanianBuah(luas\_lahan, jumlah\_tanaman, jenis\_pupuk, jenis\_buah)

        else:

            print("Jenis pertanian tidak valid.")

        db.tambah\_data\_pertanian(pertanian)

    elif choice == '2':

        db.tampilkan\_data\_pertanian()

    elif choice == '3':

        db.tampilkan\_data\_pertanian()

        id\_pertanian = input("Masukkan ID pertanian yang ingin diupdate: ")

        luas\_lahan\_baru = input("Masukkan luas lahan baru: ")

        jumlah\_tanaman\_baru = input("Masukkan jumlah tanaman baru: ")

        jenis\_pupuk\_baru = input("Masukkan jenis pupuk baru: ")

        jenis\_tanaman\_baru = input("Masukkan jenis tanaman (sayuran/buah): ")

        db.update\_data\_pertanian(id\_pertanian, luas\_lahan\_baru, jumlah\_tanaman\_baru, jenis\_pupuk\_baru, jenis\_tanaman\_baru)

    elif choice == '4':

        db.tampilkan\_data\_pertanian()

        id\_pertanian\_hapus = input("Masukkan ID pertanian yang ingin dihapus: ")

        db.hapus\_data\_pertanian(id\_pertanian\_hapus)

    elif choice == '5':

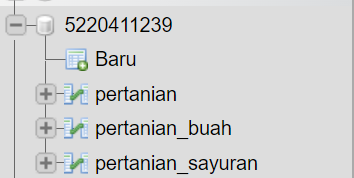
        print("Keluar dari program.")

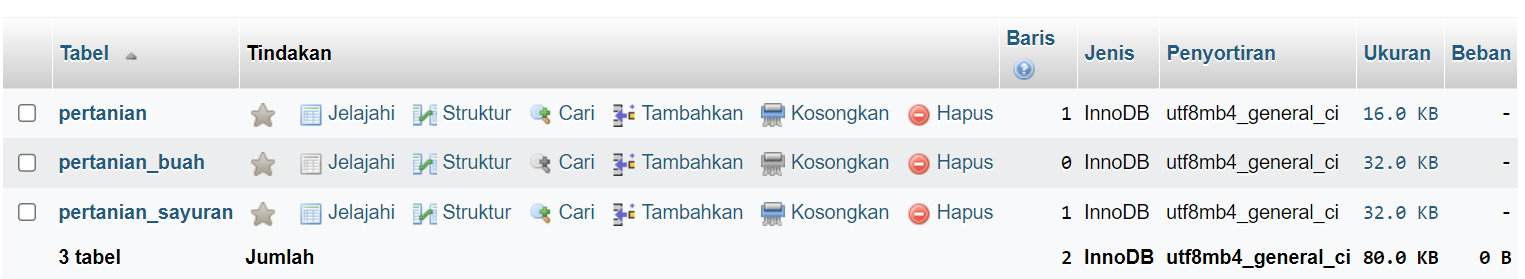
        break

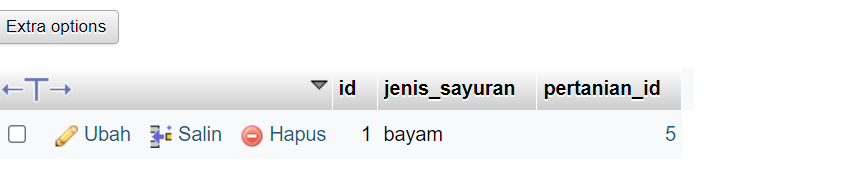
    else:

        print("Pilihan tidak valid. Silakan pilih lagi.")

1. **Database**

****

****

****

1. **Hasil Running**

