**TUGAS SQLITE MATAKULIAH PEMROGRAMAN DEKSTOP**

Disusun untuk Memenuhi Tugas Matakuliah Pemrograman Dekstop yang Dibimbing oleh Moh. Kautsar Sophan, S.Kom., M.MT.

****

Disusun oleh:

Achmad Farid Alfa Waid (190411100073)

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

**2020**

1. Pada tugas kali ini, bertujuan agar mahasiswa mengenal sqlite dan mampu menampilan data, manambah data, serta memfilter data yang ada di dalam database dengan PyQt5
2. Script untuk menampilkan data table

#Fungsi untuk menampilankan data di dalam table

def tampilData(self):

#Membuat Model

model = QSqlQueryModel()

#Mendefinisikan sql

sql = "SELECT \* FROM mahasiswa"

#Mengeksekusi Model Query

model.setQuery(sql)

#Mengeset Data Model Ke table view

self.tableview.setModel(model)

#self.tableview.setWindowTitle(title)

return self.tableview

1. Script untuk memasukkan data ke dalam table

#Fungsi tambah Data ke dalam tabel mahasiswa

def tambahData(self):

#Mengambil Text inputan

nim = str(self.Nim.text())

nama = str(self.Nama.text())

jurusan = str(self.Jurusan.text())

angkatan = str(self.Angkatan.text())

#Mendefinisikan Query

query = QSqlQuery()

#Menjalankan Perintah Sql

query.prepare("INSERT INTO mahasiswa VALUES ('" + nim + "', '" + nama + "', '" + jurusan + "', '" + angkatan + "')")

#Mengecek apakah query berjalan dengan baik

if query.exec\_():

self.Nim.setText("")

self.Nama.setText("")

self.Jurusan.setText("")

self.Angkatan.setText("")

#Menampilkan Data

self.tampilData()

else:

#Apabila error akan menampilkan errornya ke dalam terminal

print("Insert Error: ", query.lastError().text())

1. Script untuk menampilkan data table dengan filter search

#Fungsi Filter Data

def filterData(self):

#Membuat Model

model = QSqlQueryModel()

#Mengambil Inputan filter

filter\_search = str(self.Cari.text())

sql = "SELECT \* FROM mahasiswa WHERE nim LIKE '%"+str(filter\_search)+"%' OR nama LIKE '%"+str(filter\_search)+"%' OR jurusan LIKE '%"+str(filter\_search)+"%' OR angkatan LIKE '%"+str(filter\_search)+"%'"

self.Cari.setText("")

#Mengeksekusi Model Query

model.setQuery(sql)

#Mengeset Data Model Ke table view

self.tableview.setModel(model)

#mengembalikan nilai table view

return self.tableview

1. Listing Program

import sys

from PyQt5 import \*

from PyQt5.QtWidgets import \*

from PyQt5.Qt import \*

from PyQt5.QtCore import \*

from PyQt5.QtGui import \*

from PyQt5 import QtSql

import sqlite3

class Mahasiswa(QWidget):

#Membuat fungsi init untuk inisialisasi class Mahasiswa

def \_\_init\_\_(self):

#untuk mengembalikan semua atribut dan method yang ada

super().\_\_init\_\_()

#membuka database

self.OpenDatabase()

#memanggil fungsi Layout yang sudah dibuat agar ditampilkan hasilnya

self.Layout()

def OpenDatabase(self):

#Mendeklarasikan database

db = QtSql.QSqlDatabase.addDatabase('QSQLITE')

#Membuat nama database

db.setDatabaseName('test.db')

#Mengecek Database Apakah sudah terkoneksi atau belum

if db.open():

print('Berhasil membuka Database')

else:

print('Gagal membuka Database!')

def Layout(self):

#Membuat Grid Layout

grid = QGridLayout()

#Membuat label, Line Edit, dan Button yang akan dimasukkan ke dalam layout Grid

open\_database = QLabel("Buka Database")

grid.addWidget(open\_database,0,0)

open\_button = QPushButton("Open Database")

grid.addWidget(open\_button,0,1,1,2)

add\_data = QLabel("Tambah Data:")

grid.addWidget(add\_data,1,0,1,0)

add\_nim = QLabel("Nim:")

grid.addWidget(add\_nim,2,0)

self.Nim = QLineEdit(self)

grid.addWidget(self.Nim,2,1,1,2)

add\_nama = QLabel("Nama:")

grid.addWidget(add\_nama,3,0)

self.Nama = QLineEdit(self)

grid.addWidget(self.Nama,3,1,1,2)

add\_jurusan = QLabel("Jurusan:")

grid.addWidget(add\_jurusan,4,0)

self.Jurusan = QLineEdit(self)

grid.addWidget(self.Jurusan,4,1,1,2)

add\_angkatan = QLabel("Angkatan:")

grid.addWidget(add\_angkatan,5,0)

self.Angkatan = QLineEdit(self)

grid.addWidget(self.Angkatan,5,1,1,2)

add\_button = QPushButton("Tambah Data")

grid.addWidget(add\_button,6,0,1,0)

search = QLabel("Cari Data:")

grid.addWidget(search,7,0)

self.Cari = QLineEdit(self)

grid.addWidget(self.Cari,7,1)

search\_button = QPushButton("Cari")

grid.addWidget(search\_button,7,2)

#Membuat widget table view yang diberi nama "Data" dan akan dimasukkan ke dalam layout Grid

self.tableview = QTableView(self)

self.tableview.setObjectName("Data")

grid.addWidget(self.tableview,8,0,1,0)

#Ketika button di klik akan memanggi fungsi masing - masing

open\_button.clicked.connect(self.tampilData)

add\_button.clicked.connect(self.tambahData)

search\_button.clicked.connect(self.filterData)

#Layout grid di jadikan layout utama

self.setLayout(grid)

#Fungsi untuk menampilankan data di dalam table

def tampilData(self):

#Membuat Model

model = QSqlQueryModel()

#Mendefinisikan sql

sql = "SELECT \* FROM mahasiswa"

#Mengeksekusi Model Query

model.setQuery(sql)

#Mengeset Data Model Ke table view

self.tableview.setModel(model)

#self.tableview.setWindowTitle(title)

return self.tableview

#Fungsi tambah Data ke dalam tabel mahasiswa

def tambahData(self):

#Mengambil Text inputan

nim = str(self.Nim.text())

nama = str(self.Nama.text())

jurusan = str(self.Jurusan.text())

angkatan = str(self.Angkatan.text())

#Mendefinisikan Query

query = QSqlQuery()

#Menjalankan Perintah Sql

query.prepare("INSERT INTO mahasiswa VALUES ('" + nim + "', '" + nama + "', '" + jurusan + "', '" + angkatan + "')")

#Mengecek apakah query berjalan dengan baik

if query.exec\_():

self.Nim.setText("")

self.Nama.setText("")

self.Jurusan.setText("")

self.Angkatan.setText("")

#Menampilkan Data

self.tampilData()

else:

#Apabila error akan menampilkan errornya ke dalam terminal

print("Insert Error: ", query.lastError().text())

#Fungsi Filter Data

def filterData(self):

#Membuat Model

model = QSqlQueryModel()

#Mengambil Inputan filter

filter\_search = str(self.Cari.text())

sql = "SELECT \* FROM mahasiswa WHERE nim LIKE '%"+str(filter\_search)+"%' OR nama LIKE '%"+str(filter\_search)+"%' OR jurusan LIKE '%"+str(filter\_search)+"%' OR angkatan LIKE '%"+str(filter\_search)+"%'"

self.Cari.setText("")

#Mengeksekusi Model Query

model.setQuery(sql)

#Mengeset Data Model Ke table view

self.tableview.setModel(model)

#mengembalikan nilai table view

return self.tableview

if \_\_name\_\_ == '\_\_main\_\_':

#Inisisalisai pyqt

app = QApplication(sys.argv)

#mengatur style di window menjadi style fusion

app.setStyle("fusion")

#membuat variabel ex yang berisi class FormulaMath

ex = Mahasiswa()

#Menentukan ukuran window dan title untuk menampilkan

ex.setGeometry(100,100,700,500)

#membuat judul window

ex.setWindowTitle("Database Sqlite in Pyqt5")

#menampilan isi dari variabel ex

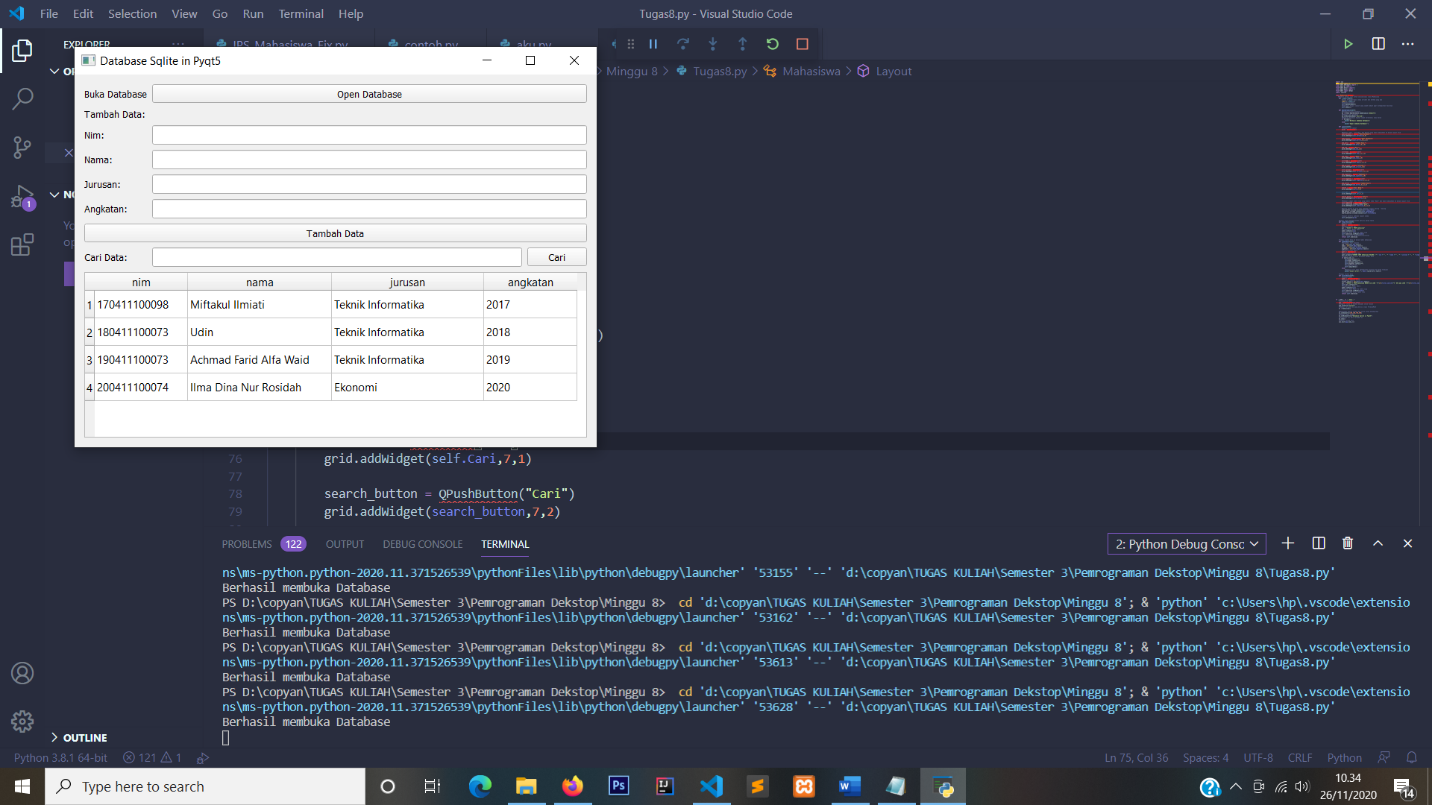
ex.show()

#membuat system exit

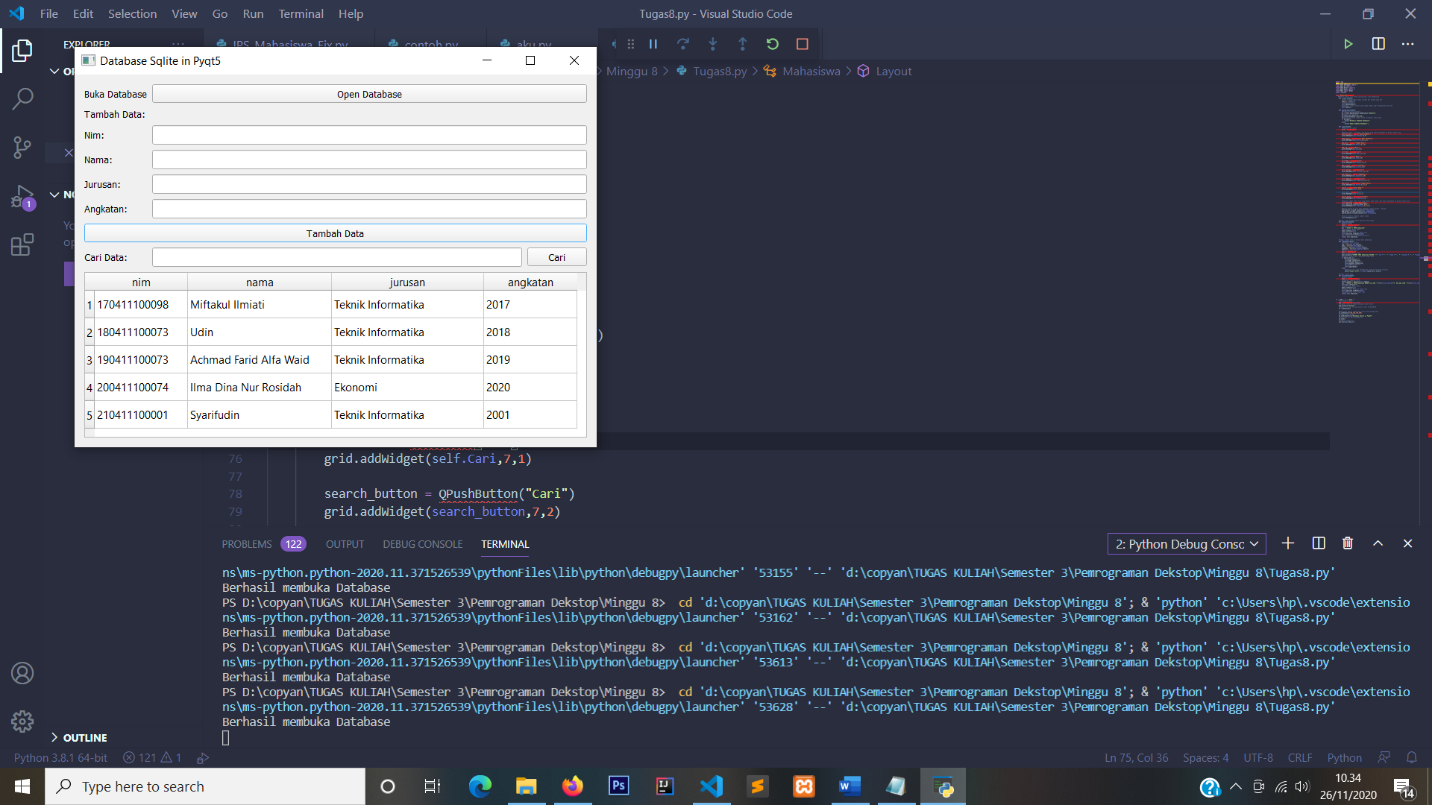
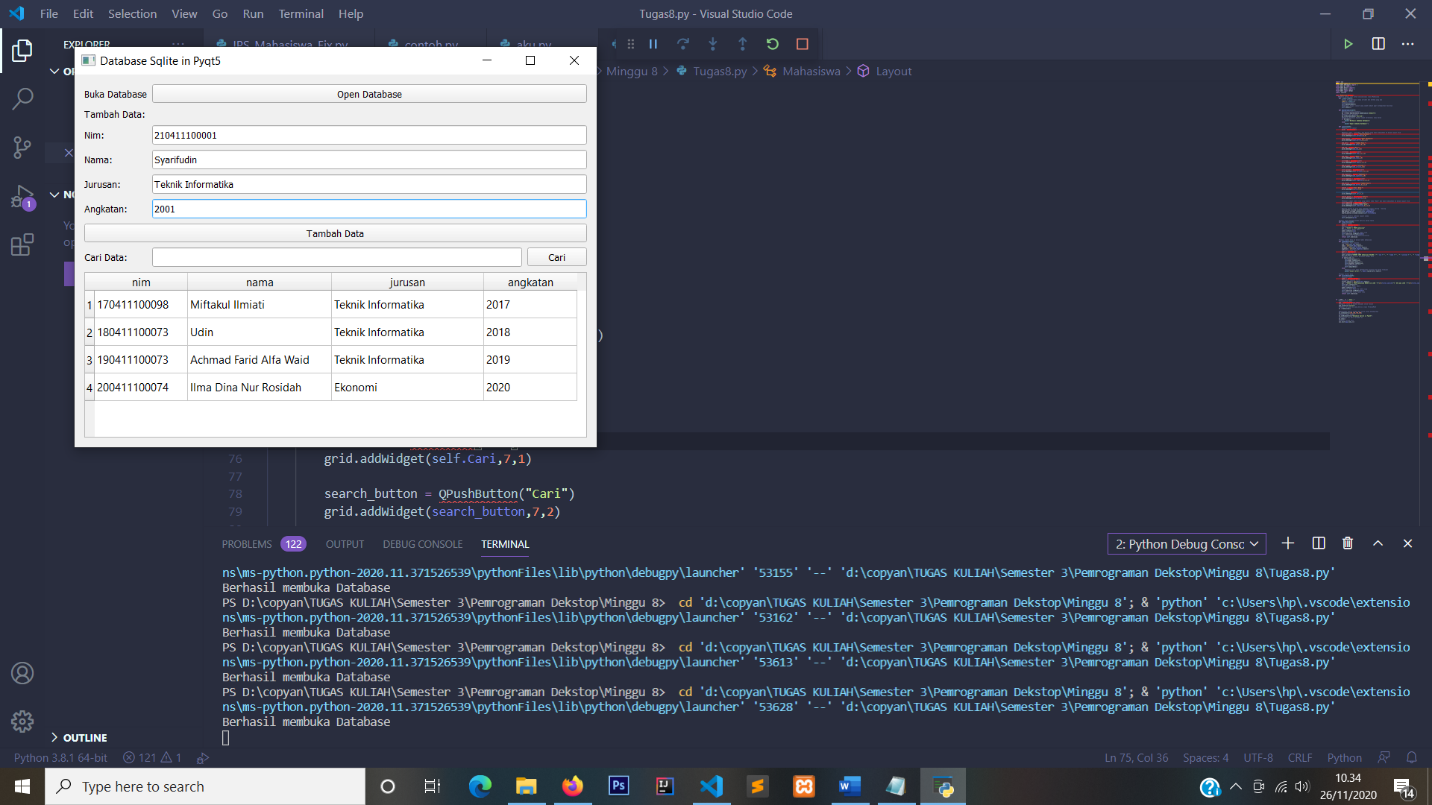
sys.exit(app.exec\_())

1. Screenshot hasil program

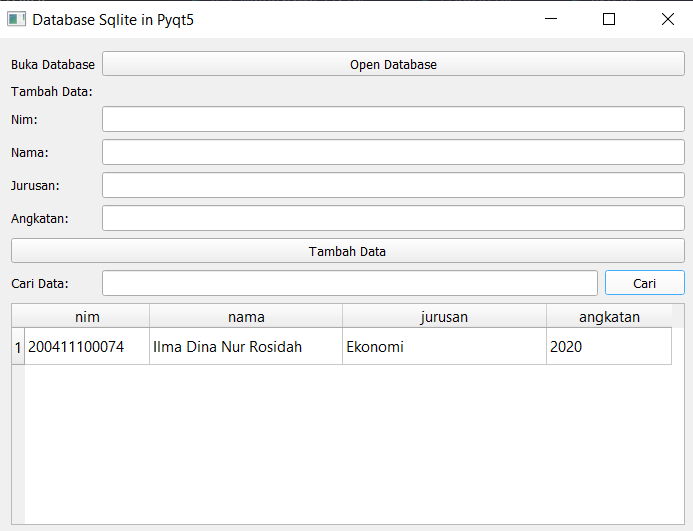
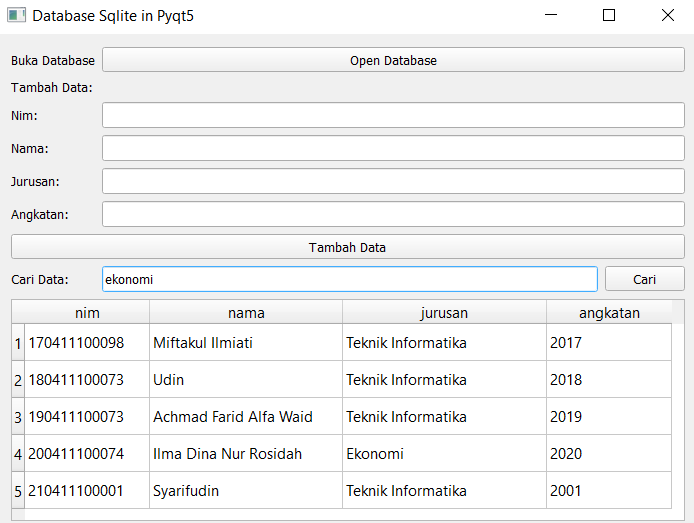
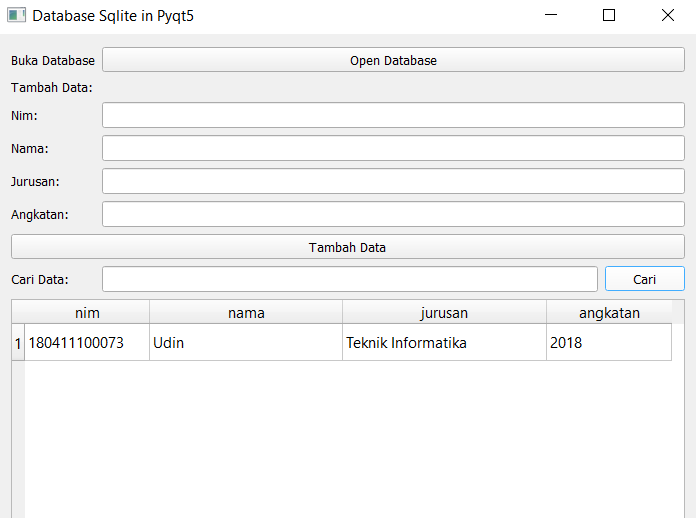
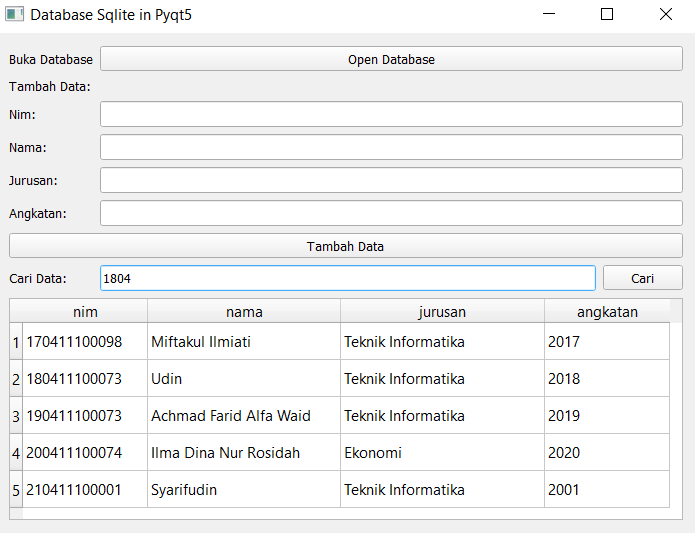
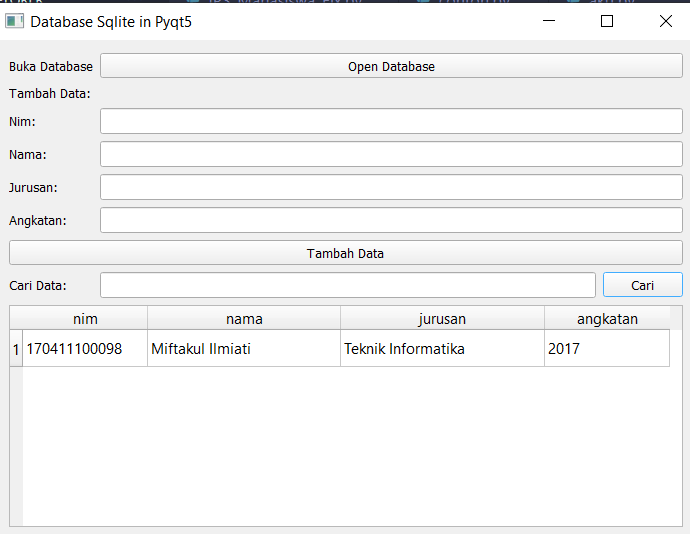
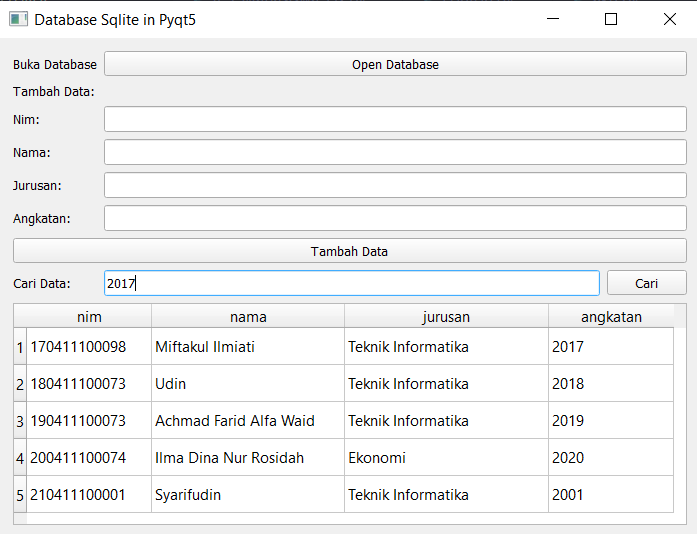
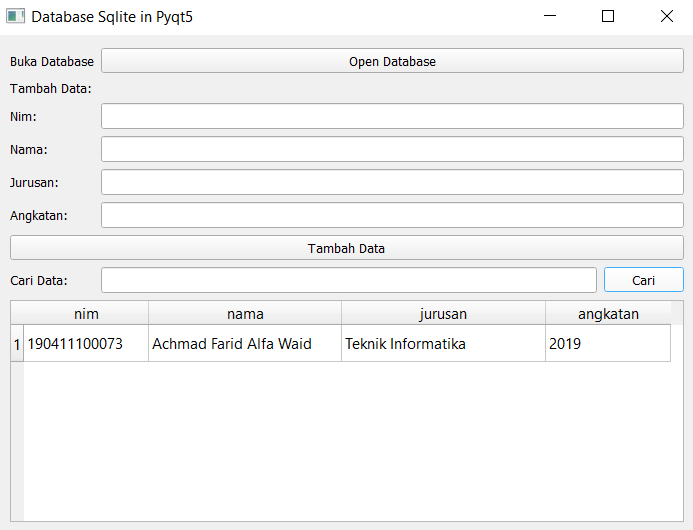
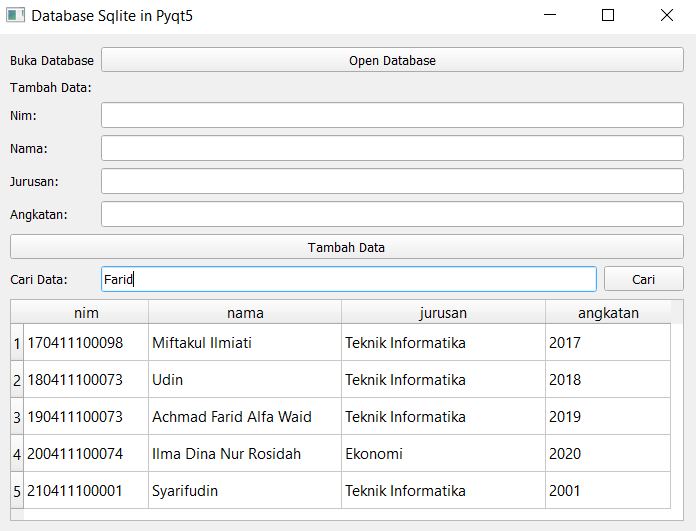
* Untuk menampilkan data table (dengan klik button open database)



* Untuk menambah data (dengan klik button Tambah Data)



* Untuk menampilkan data table dengan filter search (dengan klik button Cari)

****