**Proje 6**

import sys

import sqlite3

from PyQt5.QtWidgets import QApplication, QWidget, QVBoxLayout, QHBoxLayout, QLabel, QLineEdit, QPushButton, QTextEdit, QMessageBox

class TarifUygulamasi(QWidget):

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

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

self.setWindowTitle("Tarif Uygulaması")

self.arayuz\_olustur()

self.veritabani\_baglantisi\_olustur()

def arayuz\_olustur(self):

self.duzen = QVBoxLayout()

# Tarif

self.tarif\_ad\_label = QLabel("Tarif Adı:")

self.tarif\_ad\_input = QLineEdit()

self.duzen.addWidget(self.tarif\_ad\_label)

self.duzen.addWidget(self.tarif\_ad\_input)

self.malzemeler\_label = QLabel("Malzemeler:")

self.malzemeler\_input = QTextEdit()

self.duzen.addWidget(self.malzemeler\_label)

self.duzen.addWidget(self.malzemeler\_input)

# Malzeme

self.malzeme\_ad\_label = QLabel("Malzeme Adı:")

self.malzeme\_ad\_input = QLineEdit()

self.duzen.addWidget(self.malzeme\_ad\_label)

self.duzen.addWidget(self.malzeme\_ad\_input)

self.miktar\_label = QLabel("Miktarı:")

self.miktar\_input = QLineEdit()

self.duzen.addWidget(self.miktar\_label)

self.duzen.addWidget(self.miktar\_input)

self.tarif\_label = QLabel("Tarif:")

self.tarif\_input = QTextEdit()

self.duzen.addWidget(self.tarif\_label)

self.duzen.addWidget(self.tarif\_input)

self.tarif\_ekle\_button = QPushButton("Tarifi Ekle")

self.tarif\_ekle\_button.clicked.connect(self.tarif\_ekle)

self.duzen.addWidget(self.tarif\_ekle\_button)

# Kullanıcı

self.kullanici\_ad\_label = QLabel("Kullanıcı Adı:")

self.kullanici\_ad\_input = QLineEdit()

self.duzen.addWidget(self.kullanici\_ad\_label)

self.duzen.addWidget(self.kullanici\_ad\_input)

self.sifre\_label = QLabel("Şifre:")

self.sifre\_input = QLineEdit()

self.sifre\_input.setEchoMode(QLineEdit.Password)

self.duzen.addWidget(self.sifre\_label)

self.duzen.addWidget(self.sifre\_input)

self.setLayout(self.duzen)

def veritabani\_baglantisi\_olustur(self):

self.veritabani\_baglantisi = sqlite3.connect('tarifler.db')

self.cursor = self.veritabani\_baglantisi.cursor()

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Tarifler

(tarif\_id INTEGER PRIMARY KEY,

tarif\_adi TEXT,

malzemeler TEXT,

tarif\_metni TEXT)''')

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Kullanicilar

(kullanici\_id INTEGER PRIMARY KEY,

ad TEXT,

sifre TEXT)''')

self.veritabani\_baglantisi.commit()

def tarif\_ekle(self):

tarif\_ad = self.tarif\_ad\_input.text().strip()

malzemeler = self.malzemeler\_input.toPlainText().strip()

tarif\_icerik = self.tarif\_input.toPlainText().strip()

if tarif\_ad and malzemeler and tarif\_icerik:

self.cursor.execute("INSERT INTO Tarifler (tarif\_adi, malzemeler, tarif\_metni) VALUES (?, ?, ?)",

(tarif\_ad, malzemeler, tarif\_icerik))

self.veritabani\_baglantisi.commit()

QMessageBox.information(self, "Başarılı", "Tarif başarıyla eklendi!")

else:

QMessageBox.warning(self, "Uyarı", "Lütfen tüm alanları doldurun.")

def closeEvent(self, event):

self.veritabani\_baglantisi.close()

event.accept()

if \_\_name\_\_ == "\_\_main\_\_":

uygulama = QApplication(sys.argv)

pencere = TarifUygulamasi()

pencere.show()

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

**proje 7**

import sys

import sqlite3

from PyQt5.QtWidgets import QApplication, QWidget, QVBoxLayout, QHBoxLayout, QLabel, QLineEdit, QPushButton, QListWidget, QTextEdit, QMessageBox, QComboBox

class SporTakipUygulamasi(QWidget):

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

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

self.setWindowTitle("Spor Takip Uygulaması")

self.arayuz\_olustur()

self.veritabani\_baglantisi\_olustur()

def arayuz\_olustur(self):

self.duzen = QVBoxLayout()

# Sporcu

self.sporcu\_label = QLabel("Sporcu Adı:")

self.sporcu\_input = QLineEdit()

self.duzen.addWidget(self.sporcu\_label)

self.duzen.addWidget(self.sporcu\_input)

self.spor\_dali\_label = QLabel("Spor Dalı:")

self.spor\_dali\_input = QLineEdit()

self.duzen.addWidget(self.spor\_dali\_label)

self.duzen.addWidget(self.spor\_dali\_input)

# Adding sports options

self.spor\_dali\_label = QLabel("Spor Dalı Seçin:")

self.spor\_dali\_combobox = QComboBox()

self.spor\_dali\_combobox.addItems(["Futbol", "Voleybol", "Basketbol", "Fitness", "Golf", "Masa Tenisi", "Jimnastik"])

self.duzen.addWidget(self.spor\_dali\_label)

self.duzen.addWidget(self.spor\_dali\_combobox)

# Antrenman

self.antrenman\_label = QLabel("Antrenman Adı:")

self.antrenman\_input = QLineEdit()

self.duzen.addWidget(self.antrenman\_label)

self.duzen.addWidget(self.antrenman\_input)

self.antrenman\_detay\_label = QLabel("Antrenman Detayları:")

self.antrenman\_detay\_input = QTextEdit()

self.duzen.addWidget(self.antrenman\_detay\_label)

self.duzen.addWidget(self.antrenman\_detay\_input)

# Takip

self.antrenman\_kaydet\_button = QPushButton("Antrenmanı Kaydet")

self.antrenman\_kaydet\_button.clicked.connect(self.antrenman\_kaydet)

self.duzen.addWidget(self.antrenman\_kaydet\_button)

self.antrenman\_listesi\_label = QLabel("Antrenman Listesi:")

self.antrenman\_listesi = QListWidget()

self.duzen.addWidget(self.antrenman\_listesi\_label)

self.duzen.addWidget(self.antrenman\_listesi)

self.ilerleme\_label = QLabel("İlerleme Kaydı:")

self.ilerleme\_input = QTextEdit()

self.duzen.addWidget(self.ilerleme\_label)

self.duzen.addWidget(self.ilerleme\_input)

self.ilerleme\_kaydet\_button = QPushButton("İlerlemeyi Kaydet")

self.ilerleme\_kaydet\_button.clicked.connect(self.ilerleme\_kaydet)

self.duzen.addWidget(self.ilerleme\_kaydet\_button)

self.setLayout(self.duzen)

def veritabani\_baglantisi\_olustur(self):

self.veritabani\_baglantisi = sqlite3.connect('spor\_takip.db')

self.cursor = self.veritabani\_baglantisi.cursor()

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Sporcular

(id INTEGER PRIMARY KEY,

ad TEXT,

spor\_dali TEXT)''')

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Antrenmanlar

(id INTEGER PRIMARY KEY,

sporcu\_id INTEGER,

antrenman\_adi TEXT,

antrenman\_detay TEXT,

FOREIGN KEY(sporcu\_id) REFERENCES Sporcular(id))''')

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Takip

(id INTEGER PRIMARY KEY,

sporcu\_id INTEGER,

ilerleme\_detayi TEXT,

FOREIGN KEY(sporcu\_id) REFERENCES Sporcular(id))''')

self.veritabani\_baglantisi.commit()

def antrenman\_kaydet(self):

sporcu\_adi = self.sporcu\_input.text().strip()

spor\_dali = self.spor\_dali\_combobox.currentText()

antrenman\_adi = self.antrenman\_input.text().strip()

antrenman\_detay = self.antrenman\_detay\_input.toPlainText().strip()

if sporcu\_adi and spor\_dali and antrenman\_adi and antrenman\_detay:

self.cursor.execute("INSERT INTO Sporcular (ad, spor\_dali) VALUES (?, ?)", (sporcu\_adi, spor\_dali))

sporcu\_id = self.cursor.lastrowid

self.cursor.execute("INSERT INTO Antrenmanlar (sporcu\_id, antrenman\_adi, antrenman\_detay) VALUES (?, ?, ?)",

(sporcu\_id, antrenman\_adi, antrenman\_detay))

self.veritabani\_baglantisi.commit()

self.antrenman\_listesini\_yukle()

QMessageBox.information(self, "Başarılı", "Antrenman başarıyla kaydedildi!")

else:

QMessageBox.warning(self, "Uyarı", "Lütfen sporcu adı, spor dalı, antrenman adı ve detayları girin.")

def antrenman\_listesini\_yukle(self):

self.antrenman\_listesi.clear()

sporcu\_adi = self.sporcu\_input.text().strip()

self.cursor.execute("SELECT id FROM Sporcular WHERE ad = ?", (sporcu\_adi,))

sporcu\_id = self.cursor.fetchone()[0]

self.cursor.execute("SELECT antrenman\_adi FROM Antrenmanlar WHERE sporcu\_id = ?", (sporcu\_id,))

antrenmanlar = self.cursor.fetchall()

for antrenman in antrenmanlar:

self.antrenman\_listesi.addItem(antrenman[0])

def ilerleme\_kaydet(self):

sporcu\_adi = self.sporcu\_input.text().strip()

ilerleme\_detayi = self.ilerleme\_input.toPlainText().strip()

if sporcu\_adi and ilerleme\_detayi:

self.cursor.execute("SELECT id FROM Sporcular WHERE ad = ?", (sporcu\_adi,))

sporcu\_id = self.cursor.fetchone()[0]

self.cursor.execute("INSERT INTO Takip (sporcu\_id, ilerleme\_detayi) VALUES (?, ?)", (sporcu\_id, ilerleme\_detayi))

self.veritabani\_baglantisi.commit()

QMessageBox.information(self, "Başarılı", "İlerleme kaydedildi!")

else:

QMessageBox.warning(self, "Uyarı", "Lütfen sporcu adı ve ilerleme detayları girin.")

def closeEvent(self, event):

self.veritabani\_baglantisi.close()

event.accept()

if \_\_name\_\_ == "\_\_main\_\_":

uygulama = QApplication(sys.argv)

pencere = SporTakipUygulamasi()

pencere.show()

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

**proje 8**

import sys

import sqlite3

from PyQt5.QtWidgets import QApplication, QWidget, QVBoxLayout, QHBoxLayout, QLabel, QLineEdit, QPushButton, QTextEdit, QMessageBox, QComboBox

class StokTakipSistemi(QWidget):

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

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

self.setWindowTitle("Stok Takip Sistemi")

self.arayuz\_olustur()

self.veritabani\_baglantisi\_olustur()

def arayuz\_olustur(self):

self.duzen = QVBoxLayout()

# Product (Ürün) Section

self.urun\_label = QLabel("Ürün Adı:")

self.urun\_input = QLineEdit()

self.duzen.addWidget(self.urun\_label)

self.duzen.addWidget(self.urun\_input)

self.stok\_label = QLabel("Stok Miktarı:")

self.stok\_input = QLineEdit()

self.duzen.addWidget(self.stok\_label)

self.duzen.addWidget(self.stok\_input)

self.urun\_ekle\_button = QPushButton("Ürün Ekle")

self.urun\_ekle\_button.clicked.connect(self.urun\_ekle)

self.duzen.addWidget(self.urun\_ekle\_button)

# Stock (Stok) Section

self.stok\_listesi\_label = QLabel("Stok Durumu:")

self.stok\_combobox = QComboBox()

self.stok\_combobox.addItems(["Var", "Az", "Bitmek Üzere", "Tükenmiş"])

self.duzen.addWidget(self.stok\_listesi\_label)

self.duzen.addWidget(self.stok\_combobox)

self.stok\_guncelle\_button = QPushButton("Stok Güncelle")

self.stok\_guncelle\_button.clicked.connect(self.stok\_guncelle)

self.duzen.addWidget(self.stok\_guncelle\_button)

# Order (Sipariş) Section

self.siparis\_label = QLabel("Sipariş Numarası:")

self.siparis\_input = QLineEdit()

self.duzen.addWidget(self.siparis\_label)

self.duzen.addWidget(self.siparis\_input)

self.siparis\_detay\_label = QLabel("Sipariş Detayları:")

self.siparis\_detay\_input = QTextEdit()

self.duzen.addWidget(self.siparis\_detay\_label)

self.duzen.addWidget(self.siparis\_detay\_input)

self.siparis\_olustur\_button = QPushButton("Sipariş Oluştur")

self.siparis\_olustur\_button.clicked.connect(self.siparis\_olustur)

self.duzen.addWidget(self.siparis\_olustur\_button)

self.setLayout(self.duzen)

def veritabani\_baglantisi\_olustur(self):

self.veritabani\_baglantisi = sqlite3.connect('stok\_takip.db')

self.cursor = self.veritabani\_baglantisi.cursor()

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Urunler

(id INTEGER PRIMARY KEY,

ad TEXT,

stok TEXT)''')

self.cursor.execute('''CREATE TABLE IF NOT EXISTS Siparisler

(id INTEGER PRIMARY KEY,

siparis\_no TEXT,

detay TEXT)''')

self.veritabani\_baglantisi.commit()

def urun\_ekle(self):

urun\_adi = self.urun\_input.text().strip()

stok\_durumu = self.stok\_combobox.currentText()

if urun\_adi and stok\_durumu:

self.cursor.execute("INSERT INTO Urunler (ad, stok) VALUES (?, ?)", (urun\_adi, stok\_durumu))

self.veritabani\_baglantisi.commit()

QMessageBox.information(self, "Başarılı", "Ürün başarıyla eklendi!")

else:

QMessageBox.warning(self, "Uyarı", "Lütfen ürün adı ve stok durumu seçin.")

def stok\_guncelle(self):

QMessageBox.warning(self, "Uyarı", "Stok güncelleme işlemi henüz eklenmedi.")

def siparis\_olustur(self):

siparis\_no = self.siparis\_input.text().strip()

siparis\_detay = self.siparis\_detay\_input.toPlainText().strip()

if siparis\_no and siparis\_detay:

self.cursor.execute("INSERT INTO Siparisler (siparis\_no, detay) VALUES (?, ?)", (siparis\_no, siparis\_detay))

self.veritabani\_baglantisi.commit()

QMessageBox.information(self, "Başarılı", "Sipariş başarıyla oluşturuldu!")

else:

QMessageBox.warning(self, "Uyarı", "Lütfen sipariş numarası ve detayları girin.")

def closeEvent(self, event):

self.veritabani\_baglantisi.close()

event.accept()

if \_\_name\_\_ == "\_\_main\_\_":

uygulama = QApplication(sys.argv)

pencere = StokTakipSistemi()

pencere.show()

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