CREATE TABLE hava\_durumu (

sehir TEXT,

tarih DATE,

sicaklik INTEGER,

nem INTEGER,

yagis INTEGER,

ruzgar INTEGER,

);

INSERT INTO hava\_durumu VALUES (

'İstanbul',

'2024-01-02',

10,

75,

0,

15,

);

INSERT INTO hava\_durumu VALUES (

'İzmir',

'2022-03-06',

20,

45,

1,

35,

);

SELECT \* FROM hava\_durumu WHERE sehir = 'İstanbul';

SELECT \* FROM hava\_durumu WHERE sehir = 'İzmir';

import sqlite3

# Veritabanına bağlanma komutu olarak yazdım.

baglanti = sqlite3.connect("hava\_durumu.db")

# İmleç oluşturma komutum.

imlec = baglanti.cursor()

# Veri okuma fonksiyonunu oluşturdum.

def veri\_oku(sehir):

imlec.execute("SELECT \* FROM hava\_durumu WHERE sehir = ?", (sehir,))

sonuc = imlec.fetchall()

return sonuc

# Veri yazma fonksiyonum.

def veri\_yaz(sehir, tarih, sicaklik, nem, yagis, ruzgar,durum):

imlec.execute("INSERT INTO hava\_durumu VALUES (?, ?, ?, ?, ? ,?)", (sehir, tarih, sicaklik, nem, yagis, ruzgar,))

baglanti.commit()

# Veri güncelleme fonkiyonum.

def veri\_guncelle(sehir, tarih, sicaklik, nem, yagis, ruzgar,durum):

imlec.execute("UPDATE hava\_durumu SET sicaklik = ?, nem = ?, yagis = ?, ruzgar = ? WHERE sehir = ? AND tarih = ?", (sicaklik, nem, yagis, ruzgar, sehir, tarih))

baglanti.commit()

# Veri silme fonksiyonumu oluşturdum.

def veri\_sil(sehir, tarih):

imlec.execute("DELETE FROM hava\_durumu WHERE sehir = ? AND tarih = ?", (sehir, tarih))

baglanti.commit()

# Data katmanındaki fonksiyonlarımı içe aktardım.

from data\_katmani import veri\_oku, veri\_yaz, veri\_guncelle, veri\_sil

# Servis katmanında fonksiyonlar tanımlıyorum.

# Hava durumu raporu oluşturma fonksiyonum:

def hava\_raporu(sehir, tarih):

veriler = veri\_oku(sehir)

for veri in veriler:

if veri[1] == tarih:

rapor = f"{sehir} şehrinin {tarih} tarihli hava durumu raporu:\n"

rapor += f"Sıcaklık: {veri[2]} °C\n"

rapor += f"Nem: {veri[3]} %\n"

rapor += f"Yağış: {veri[4]} mm\n"

rapor += f"Rüzgar: {veri[5]} km/s\n"

return rapor

return f"{sehir} şehrinin {tarih} tarihli hava durumu verisi bulunamadı."

# Hava durumu verisi ekleme fonksiyonum:

def veri\_ekle(sehir, tarih, sicaklik, nem, yagis, ruzgar):

veri\_yaz(sehir, tarih, sicaklik, nem, yagis, ruzgar)

return f"{sehir} şehrinin {tarih} tarihli hava durumu verisi başarıyla eklendi."

# Hava durumu verisi güncelleme fonksiyonunu oluşturdum:

def veri\_guncelle(sehir, tarih, sicaklik, nem, yagis, ruzgar):

veri\_guncelle(sehir, tarih, sicaklik, nem, yagis, ruzgar)

return f"{sehir} şehrinin {tarih} tarihli hava durumu verisi başarıyla güncellendi."

# Hava durumu verisi silme fonksiyonum:

def veri\_sil(sehir, tarih):

veri\_sil(sehir, tarih)

return f"{sehir} şehrinin {tarih} tarihli hava durumu verisi başarıyla silindi."

# Flask modülünü içe aktardım.

from flask import Flask, request, jsonify

# Servis katmanındaki fonksiyonları içe aktarıyorum.

from servis\_katmani import hava\_raporu, veri\_ekle, veri\_guncelle, veri\_sil

# Flask uygulaması oluşturdum ve kaydettim.

app = Flask(\_\_name\_\_)

# Hava raporu servisi tanımladım.

@app.route("/hava\_raporu", methods=["GET"])

def hava\_raporu\_servis():

# Parametreleri almak için yazdım

sehir = request.args.get("sehir")

tarih = request.args.get("tarih")

# Hava raporu oluşturdum.

rapor = hava\_raporu(sehir, tarih)

# Raporu JSON formatında döndürmek için yazdım.

return jsonify({"rapor": rapor})

# Veri ekleme servisini tanımladım.

@app.route("/veri\_ekle", methods=["POST"])

def veri\_ekle\_servis():

# Parametreleri alıyorum.

sehir = request.form.get("sehir")

tarih = request.form.get("tarih")

sicaklik = request.form.get("sicaklik")

nem = request.form.get("nem")

yagis = request.form.get("yagis")

ruzgar = request.form.get("ruzgar")

# Veri ekleme fonksiyonumu çağırdım işleme aldım.

mesaj = veri\_ekle(sehir, tarih, sicaklik, nem, yagis, ruzgar)

# Mesajı JSON formatında döndürdüm.

return jsonify({"mesaj": mesaj})

# Veri güncelleme servisi tanımladım.

@app.route("/veri\_guncelle", methods=["PUT"])

def veri\_guncelle\_servis():

# Parametreleri aldım.

sehir = request.form.get("sehir")

tarih = request.form.get("tarih")

sicaklik = request.form.get("sicaklik")

nem = request.form.get("nem")

yagis = request.form.get("yagis")

ruzgar = request.form.get("ruzgar")

# Veri güncelleme fonksiyonunu çağırmak için yazdım.

mesaj = veri\_guncelle(sehir, tarih, sicaklik, nem, yagis, ruzgar)

# Mesajı JSON formatında döndürmek için yazdım.

return jsonify({"mesaj": mesaj})

# Veri silme servisi tanımladım.

@app.route("/veri\_sil", methods=["DELETE"])

def veri\_sil\_servis():

# Parametreleri almak için yazdım.

sehir = request.form.get("sehir")

tarih = request.form.get("tarih")

# Veri silme fonksiyonunu çağırmak için kullandım.

mesaj = veri\_sil(sehir, tarih)

# Mesajı JSON formatında döndürmek için yazdım.

return jsonify({"mesaj": mesaj})

# Uygulamayı çalıştırma fonksiyonum.

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

app.run()