

Nama : Andi Sibwayiq Abi Mahmud

Nim : F1D022002

Kelas : C

WeatherNow

Aplikasi Cuaca *Real-Time* Berbasis PyQt5 dengan Sentuhan Visual Sederhana

A. Deskripsi Singkat Aplikasi

Aplikasi ini dirancang untuk memberikan informasi cuaca terkini berdasarkan nama kota yang dimasukkan oleh pengguna. Aplikasi ini memanfaatkan OpenWeatherMap API untuk mendapatkan data suhu, kelembaban, serta deskripsi cuaca secara *real-time*. Dibangun menggunakan PyQt5, aplikasi ini menampilkan antarmuka yang intuitif dan responsif, serta mendukung fitur seperti ekspor riwayat pencarian, penghapusan data, dan penggantian tema (terang atau gelap). Aplikasi ini cocok digunakan oleh masyarakat umum maupun pelajar yang ingin mendapatkan informasi cuaca harian dengan cepat dan mudah.

B. Langkah-Langkah Pembuatan

1. Perancangan Struktur Dasar Aplikasi:

- Membuat kelas utama `WeatherApp` yang mewarisi dari `QMainWindow`.
- Menyusun antarmuka pengguna secara manual menggunakan `QVBoxLayout`.

2. Komponen GUI:

- Menambahkan `QLineEdit`, `QPushButton`, `QLabel`, `QListWidget`, dan `QToolBar`.
- Menata elemen-elemen tersebut dengan layout yang responsive.

3. Integrasi API Cuaca:

- Menggunakan pustaka `requests` untuk melakukan HTTP GET ke OpenWeatherMap.
- *Parsing* data JSON untuk mengambil informasi suhu, kelembaban, dan deskripsi cuaca.

4. Fitur Penyimpanan dan Ekspor Data:

- Riwayat pencarian ditambahkan ke `QListWidget`.
- Data riwayat dapat diekspor ke file `.csv` melalui `QFileDialog`.

5. Menu Bar dan Status Bar:
 - Menyediakan menu *File*, *Help*, dan *Delete* melalui `QMenuBar`
 - `QStatusBar` menampilkan nama dan NIM secara permanen menggunakan `QLabel`.
6. Pengaturan Tema Aplikasi
 - Tema terang dan gelap dapat diubah menggunakan tombol `toolbar`.
 - Warna *background* dinamis berdasarkan deskripsi cuaca yang ditampilkan.

C. Penjelasan Fungsi-Fungsi Utama

1. `__init__(self)`

Fungsi konstruktor utama dari kelas `WeatherApp`. Di sinilah semua komponen antarmuka diinisialisasi, seperti `QLineEdit`, tombol-tombol, *layout* utama (`QVBoxLayout`), menu bar, status bar, toolbar, serta pemanggilan awal untuk menampilkan data 5 kota terpanas. Fungsi ini memastikan seluruh komponen terpasang dan siap digunakan saat aplikasi dibuka.

2. `create_menu_bar(self)`

Membuat `QMenuBar` dan menambahkan tiga menu utama: *File*, *Delete*, dan *Help*.

- *File*, berisi opsi ekspor data ke CSV dan keluar dari aplikasi.
- *Delete*, untuk menghapus semua riwayat atau hanya yang dipilih.
- *Help*, menampilkan informasi pembuat aplikasi.

3. `create_top_toolbar(self)`

Fungsi ini membuat `QToolBar` yang ditempatkan di atas aplikasi, berisi satu tombol Ubah Tema. Saat tombol diklik, aplikasi akan berpindah antara mode terang dan gelap secara dinamis. Toolbar ini bersifat tidak dapat dipindahkan (`setMovable(False)`) dan memberikan kenyamanan pengguna.

4. `show_about(self)`

Menampilkan *pop-up* informasi yang memuat nama pembuat aplikasi dan NIM. Menggunakan `QMessageBox.information`, fungsi ini memberi konteks kepada pengguna terkait siapa yang mengembangkan aplikasi ini.

5. `tampilkan_5_kota_terpanas(self)`

Mengambil cuaca dari 5 kota besar di Indonesia seperti Jakarta, Surabaya, dan Bandung, lalu mengurutkannya berdasarkan suhu tertinggi. Informasi tersebut ditampilkan pada `QLabel` utama saat aplikasi pertama kali dibuka. Fungsi ini membuat aplikasi langsung terasa informatif tanpa interaksi awal dari pengguna

6. `set_weather_color(self, description)`

Mengatur warna *background* aplikasi secara dinamis sesuai deskripsi cuaca (cerah, hujan, berawan, dll). Hal ini memberi nuansa visual yang mencerminkan kondisi cuaca, sehingga pengguna bisa merasakannya secara intuitif.

7. `get_weather(self)`

Fungsi inti dari aplikasi. Mengambil input kota dari pengguna, memanggil API OpenWeatherMap, dan menampilkan hasil berupa suhu, kelembaban, cuaca, dan waktu ke dalam `QLabel`. Data juga ditambahkan ke `QListWidget` sebagai riwayat. Selain itu, fungsi ini juga memanggil `set_weather_color` untuk menyesuaikan warna tampilan berdasarkan cuaca.

8. `export_to_csv(self)`

Menulis semua item dari `QListWidget` ke dalam file `.csv` yang dipilih pengguna melalui `QFileDialog`. Fungsi ini menggunakan pustaka `csv` untuk menyimpan data, memungkinkan pengguna menyimpan rekam jejak pencarian cuaca secara permanen.

9. `Hapus_riwayat(self)`

Menghapus seluh riwayat pencarian di `QListWidget`. Setelah dihapus, pengguna akan mendapat notifikasi sukses lewat `QMessageBox`.

10. `hapus_riwayat_terpilih(self)`

Menghapus hanya item riwayat yang sedang dipilih oleh pengguna. Tindakan ini dilindungi dengan dialog konfirmasi (`QMessageBox.question`) agar tidak terjadi penghapusan tidak disengaja.

11. `toggle_theme(self)`

Memanggil fungsi `set_dark_theme()` atau `set_light_theme()` secara bergantian berdasarkan status boolean `self.dark_mode`. Fungsi ini digunakan oleh tombol “Ubah Tema” untuk mengganti mode tampilan aplikasi.

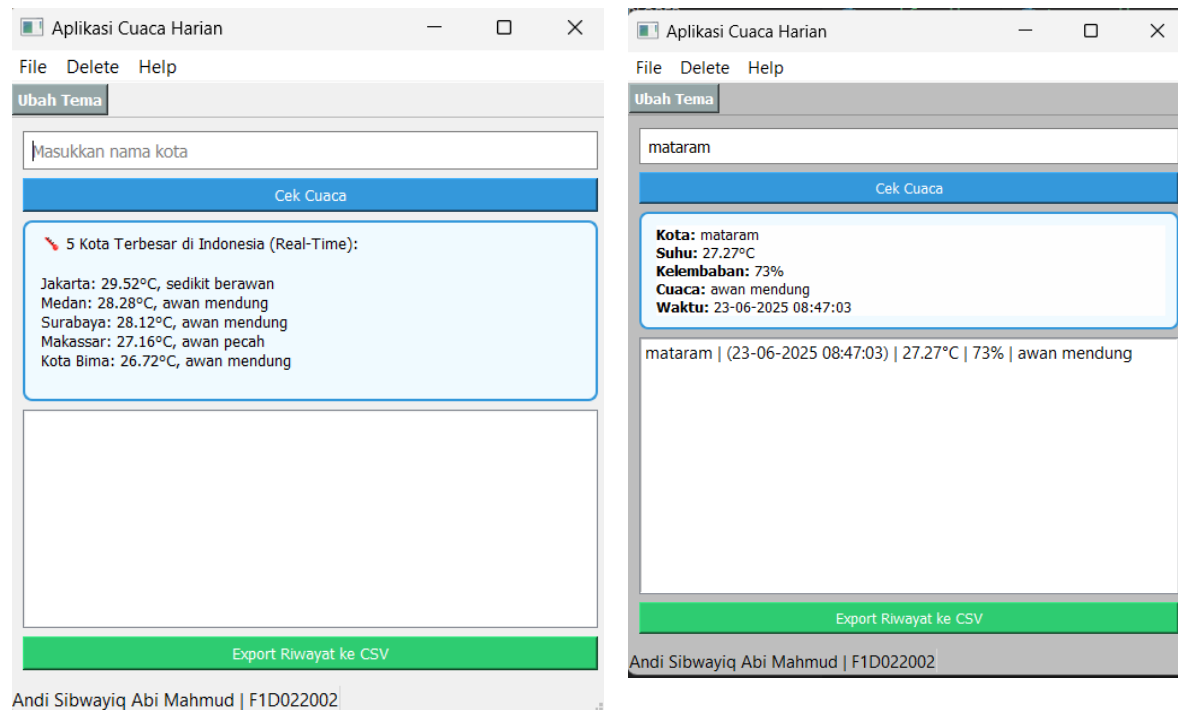
12. `set_dark_theme(self)`

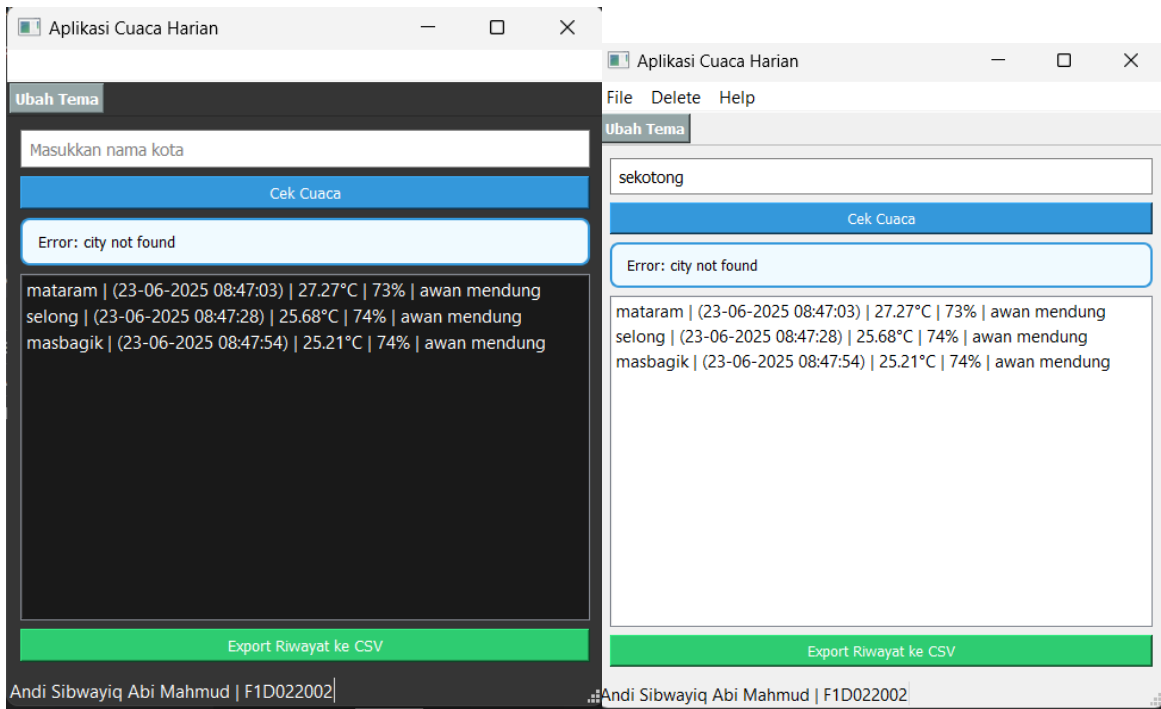
Mengubah palet warna aplikasi menjadi dominan gelap (*dark mode*), dengan teks putih dan latar abu gelap. Ini cocok digunakan pada malam hari atau untuk kenyamanan visual

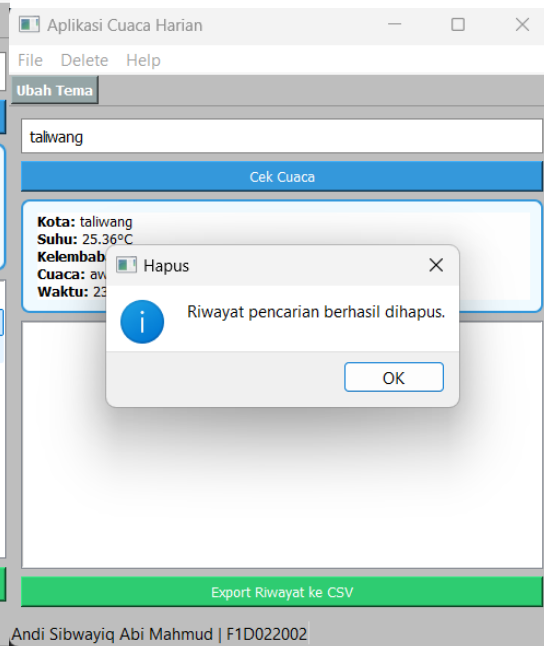
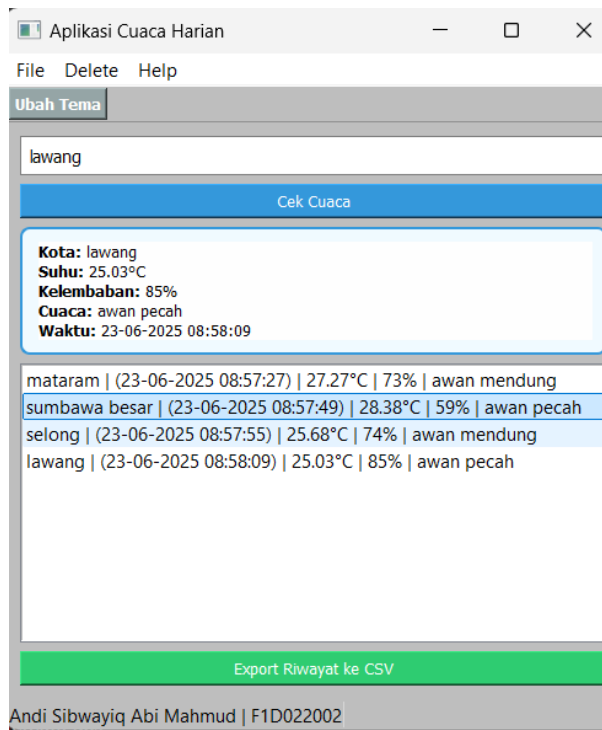
13. `set_light_theme(self)`

Mengembalikan tampilan aplikasi ke palet warna *default* (terang). Ini adalah tema standar PyQt5 dengan warna latar putih dan teks hitam.

D. Screenshot Antarmuka Aplikasi







E. Kesimpulan

Aplikasi **WeatherNow** yang dibangun dengan PyQt5 ini terbukti memberikan manfaat langsung bagi pengguna dalam memperoleh informasi cuaca secara cepat, akurat, dan interaktif. Melalui antarmuka yang sederhana namun menarik, pengguna dapat dengan mudah mengetahui kondisi cuaca terkini hanya dengan mengetikkan nama kota. Fitur seperti riwayat pencarian, ekspor data ke CSV, dan dukungan tema terang maupun gelap semakin meningkatkan kenyamanan dan fleksibilitas penggunaan.

Dengan integrasi API cuaca yang *real-time*, aplikasi ini tidak hanya relevan secara fungsional, tetapi juga praktis digunakan dalam kehidupan sehari-hari, baik oleh pelajar, pekerja, maupun masyarakat umum. Pengguna tidak perlu lagi membuka *browser* atau aplikasi pihak ketiga untuk mendapatkan informasi cuaca cukup melalui satu aplikasi ringan ini, semua dapat diakses secara langsung.

Dengan demikian, aplikasi ini bukan hanya hasil pembelajaran pemrograman visual, melainkan juga sebuah solusi yang memiliki nilai guna nyata dan berorientasi pada kebutuhan pengguna (*user-centered design*). Ke depannya,