

PRO Szkolenie Techniczne 4- Dokumentacja

22.06.2025

Spis treści

Wstęp	1
Zrzuty Ekranu	1
Klasy	5
Klasa „Filtr”	5
Metaklasa „Metaklasa”	5
Dekoratory	5
Funkcje	6
„Main”	6

Wstęp

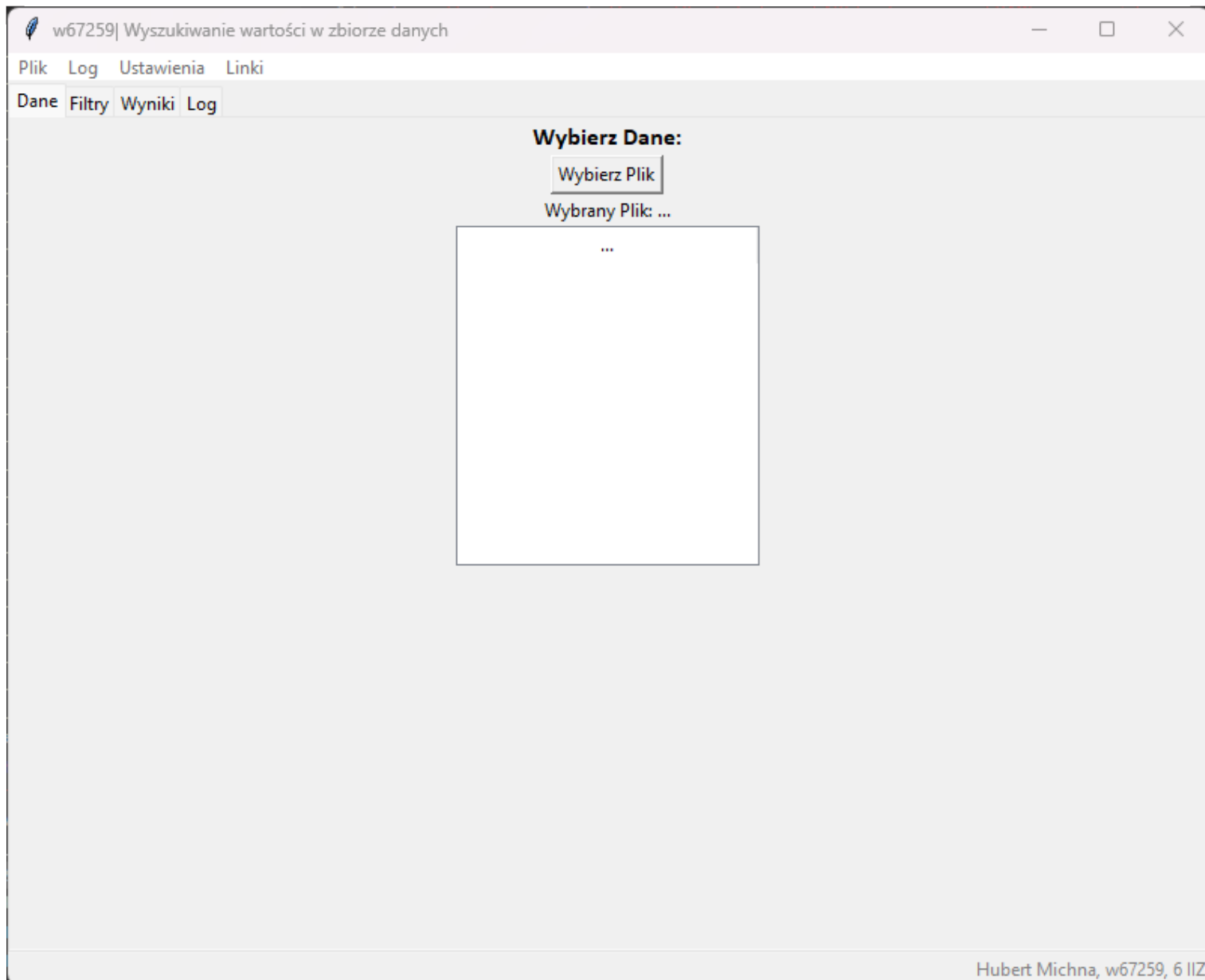
W ramach tego projektu przygotowałem aplikację desktopową która pozwala na wybranie pliku „.csv” lub „.db” z bazą danych SQLite. W przypadku bazy danych, dane zostaną wzięte z tabeli/widoku „Tabela_1”. Po wybraniu pliku można tworzyć i usuwać filtry, które pozwalają na przefiltrowanie danych w taki sposób, że zostaną wyświetlone tylko takie dane, które spełniają co najmniej jeden filtr; Pomiędzy wszystkimi filtrami jest logiczny OR.

Bardziej szczegółowe informacje dotyczące działania programu znajdują się kodzie źródłowym, wszystkie funkcje są odpowiednio nazwane w stosunku do tego co robią, podobnie ze zmiennymi. Każda funkcja zawiera bardziej szczegółowe komentarze.

Zrzuty Ekranu

Aplikacja zawiera MenuBar, główne okno aplikacji zawiera cztery zakładki:

- Dane – Tutaj użytkownik wybiera plik.
- Filtry – Tutaj użytkownik tworzy i/lub usuwa filtry.
- Wyniki – Tutaj użytkownik może przefiltrować dane według utworzonych przez niego filtrów.
- Log – Tutaj znajdują się dodatkowe, pomocnicze komunikaty tekstowe.



Rysunek 1 - Zakładka "Dane", przed wybraniem pliku

w67259 | Wyszukiwanie wartości w zbiorze danych

PlikLogUstawieniaLinki

DaneFiltryWynikiLog

Wybierz Dane:

Wybierz Plik

Wybrano plik: 'D:/WIP_2.csv'

locationIdentifier	locationType	locationName	address1
LT-1	OPERATIONS	LightTree Plant 1 Birmingham	451 24th St N
LT-2	OPERATIONS	LightTree Plant 2 Tuscaloosa	2436 University Blvd
Alumi-lux - Nashua	SUPPLIER	Alumi-lux Metal Products - Nashua	5 Main St
Amos - Philadelphia	SUPPLIER	Amos Wire Products Inc - Philadelph	1500 PA-3
Autonomic - Morgantown	SUPPLIER	Autonomic Specialties - Morgantow	341-349 Oakland St
Benchmark - Wheaton	SUPPLIER	Benchmark Lighting - Wheaton	2100 Willow Run
Brewers - Phoenix	SUPPLIER	Brewers Manufacturing - Phoenix	22 W Washington St
Core - Saranac Lake	SUPPLIER	Core Components of North America	272 Forest Hill Ave
Corporal - Tulsa	SUPPLIER	Corporal Brass - Tulsa	199 South Boulder Ave W
EM - Hemphill	SUPPLIER	Engineered Materials - Hemphill	110 Market St

Hubert Michna, w67259, 6 IIZ

Rysunek 2 - Zakładka "Dane", po wybraniu pliku

w67259 | Wyszukiwanie wartości w zbiorze danych

Plik

Log

Ustawienia

Linki

Dane

Filtry

Wyniki

Log

Dodaj Nowy Filtr:

Kolumna:

locationType

Treść:

OPERATIONS

Rodzaj Danych:

Tekst

Dodaj Filtr

Obejdnę Filtry:

	ID:	Treść:	Rodzaj:	Kolumna:
1		OPERATIONS	Tekst	locationType

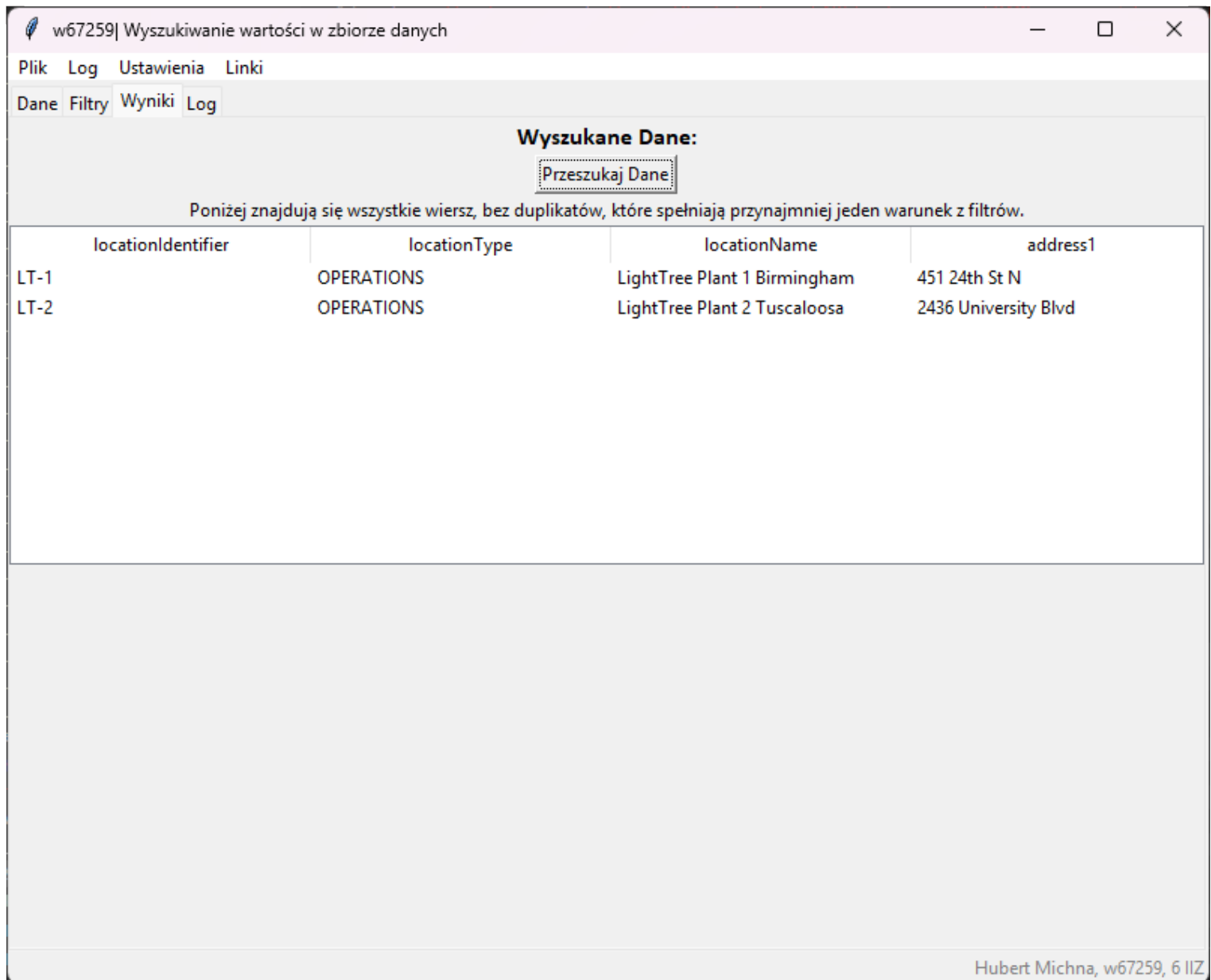
Usuń Wszystkie Filtr

Usuń Zaznaczony Filtr

Hubert Michna, w67259, 6 IIZ

Rysunek 3 - Zakładka "Filtry"

Hubert Michna; w67259.docx; 4



Rysunek 4 - Zakładka "Wyniki"

Klasy

Aplikacja zawiera jedną metaklasę i jedną klasę.

Klasa „Filtr”

Klasa będąca reprezentacją filtrów. Posiada właściwość klasy „Licznik” służący jako licznik tego, ile razy obiekt tego typu został utworzony. Jej metaklasą jest „Metaklasa”.

Konstruktor przyjmuje parametry:

- Kolumna (string) – W jakiej kolumnie dane mają być sprawdzane
- Treść (string) – Treść jaka ma zostać znaleziona
- Rodzaj (string) – Trzeba podać „Tekst”, „Liczba” lub „RegEx”

Metaklasa „Metaklasa”

Klasa będąca metaklasą. Wymusza, aby właściwości klasy były rozdzielane znakiem podłogi i żeby każde słowo było kapitalizowane, nie mogą również zawierać polskich znaków.

Właściwości tworzone podczas korzystania z konstruktora nie muszą spełniać tego warunku.

Dekoratory

Aplikacja wykorzystuje cztery dekoratory, można je podzielić następująco:

- Dopisujące do Loga:
 - „Dekorator_Print_Czas” – Kiedy funkcja została wykonana.

○ „Dekorator_Print_Czas_Trwania” – Kiedy funkcja została wykonana i ile czasu to zajęło.

- Wyświetlające w konsoli:

- „Dekorator_Log_Czas” – Kiedy funkcja została wykonana.

- „Dekorator_Log_Czas_Trwania” – Kiedy funkcja została wykonana i ile czasu to zajęło.

Funkcje

Poniżej zostały opisane najważniejsze funkcje wykorzystywane w tej aplikacji:

- „Dane_Wybierz_Plik()” – Otwiera okno wyboru odpowiedniego typu pliku, czyta jego zawartość i wyświetla jako tabelka.
- „Filtr_Dodanie()” – Dodaje nowy filtr na podstawie danych podanych przez użytkownika, waliduje poprawne dane i dodaje nowy filtr do tabelki.
- „Dane_Filtrowanie()” – Dla każdego z filtrów, tworzy proces, który realizuje dany filtr poprzez funkcję „Dane_Filtrowanie_Proces”
- „Dane_Filtrowanie_Proces(_Filtr: Filtr, _Dane: pandas.DataFrame, _Kolejka: multiprocessing.Queue)” – Ta funkcja jest wywoływana w procesach. Przyjmuje obiekt typu „Filtr”, i na podstawie jego właściwości iteruje przez wszystkie wiersze z DataFrame przekazanego jako dane, te z nich które spełniają filtr są dodawane do kolejki. Dane są sprawdzane tylko w kolumnie o nazwie z „Filtr.Kolumna”. Jeśli „Filtr.Rodzaj” to „Tekst”, dane z kolumny są interpretowane jako string, jeśli „Filtr.Rodzaj” to „RegEx”, to są interpretowane jako wyrażenie regularne, jeśli „Filtr.Rodzaj” to „Liczba” to jeśli dane są możliwe do interpretacji jako float, to są tak traktowane.

„Main”

Tutaj znajdują się wszystkie rzeczy związane z tworzeniem i umieszczaniem elementów interfejsu użytkownika.

Są tutaj również tworzone następujące zmienne globalne:

- „Filtry” – Pusta tablica, służąca do przechowywania filtrów.
- „Dane” – Pusty pandas.DataFrame, służący do przechowywania danych z pliku podanego przez użytkownika.
- „Dane_Przefiltrowane” – Pusty pandas.DataFrame, służący do przechowywania przefiltrowanych danych z pliku podanego przez użytkownika.