DS210 - Final Project: Bank Transaction Analysis with Rust

A. Project Overview

Goal: Analyze over 1 million bank transaction records to uncover regional trends, time series patterns, and detect anomalies, then visualize the results via an interactive HTML dashboard.

Dataset:

- Source: Kaggle - Massive Bank dataset (1 Million+ rows)
- Size: ~1 million transactions
- Fields: Date, Domain, Location, Transaction Value

B. Data Processing

How it was loaded into Rust:

- Loaded the CSV using the csv crate.
- Initialized a Reader object to parse each record.

Cleaning and transformations applied:

- Skipped the header row.
- Trimmed whitespace from the domain and location string fields.
- Manually fixed known typos (e.g., "RESTRAUNT" ➔ "RESTAURANT").
- Parsed each row into a Transaction struct.
- Appended valid rows into a vector for downstream processing.

C. Code Structure

Modules:

- parser.rs: Handles CSV reading and cleaning.
- models.rs: Defines custom Rust structs for storing data.
- analysis.rs: Processes and aggregates data, detects anomalies.
- main.rs: Runs the full pipeline and generates visualizations.

Purpose of Each Module:

parser.rs - Load, clean, and parse CSV data.

models.rs - Define Transaction, RegionStats, and MonthStats types for structured data handling.

analysis.rs - Aggregate statistics, compute medians/percentiles, detect anomalies.

main.rs - Coordinate loading, analysis, and HTML visualization generation.

Key Functions & Types:

- Transaction struct (models.rs)
    - Purpose: Represent a single transaction
    - Inputs: Date, domain, location, transaction value.
    - Outputs: Structured transaction entry.
    - Core logic: Simple container for each CSV row.

- RegionStats struct (models.rs)
    - Purpose: Aggregate statistics per city.
    - Inputs: Vector of transactions grouped by city.
    - Outputs: Total, average, median, and count.

- MonthStats struct (models.rs)
    - Purpose: Aggregate statistics per month.
    - Inputs: Transactions grouped by month.
    - Outputs: Monthly total and transaction count.

- parse_csv() function (parser.rs)
    - Purpose: Load and clean the CSV dataset.
    - Inputs: File path to the CSV.
    - Outputs: Vec<Transaction>.
    - Core logic: Reader + cleaner + typo fixer.

- compute_region_stats() (analysis.rs)
    - Purpose: Aggregate city-level statistics.
    - Inputs: Vec<Transaction>.
    - Outputs: Vec<RegionStats>.

- compute_month_stats() (analysis.rs)
    - Purpose: Aggregate month-level statistics.
    - Inputs: Vec<Transaction>.
    - Outputs: Vec<MonthStats>.

- calculate_median() function (analysis.rs)
    - Purpose: Find median from a vector of floats.

    - Inputs: Vec<f64>.
    - Outputs: Single median value.

- detect_anomalies() function (analysis.rs)
    - Purpose: Use IQR method to find outliers.
    - Inputs: Vector of transaction values.
    - Outputs: List of anomalous transactions.

(Note: Anomalies were computed but not visualized.)

D. Tests

- Test parsing: Ensures a sample transaction row is parsed correctly. Catches CSV parsing errors early.
- Test median: Verifies correct computation of median from sample data. Ensures the accuracy of statistical outputs.
- Test percentile: Confirms correct percentile calculation. Important for detecting anomalies and understanding distributions.

```
warning: `bank-analysis` (bin "bank-analysis" test) generated 2 warnings
    Finished `test` profile [unoptimized + debuginfo] target(s) in 52.77s
     Running unittests src\main.rs (target\debug\deps\bank_analysis-321a539db2b89579.exe)

running 3 tests
test analysis::test_percentile ... ok
test analysis::test_median ... ok
test parser::test_load_file ... ok

test result: ok. 3 passed; 0 failed; 0 ignored; 0 measured; 0 filtered out; finished in 0.00s

PS C:\Users\wilso\Documents\PycharmProjects\ds210-final-project\bank-analysis>
```

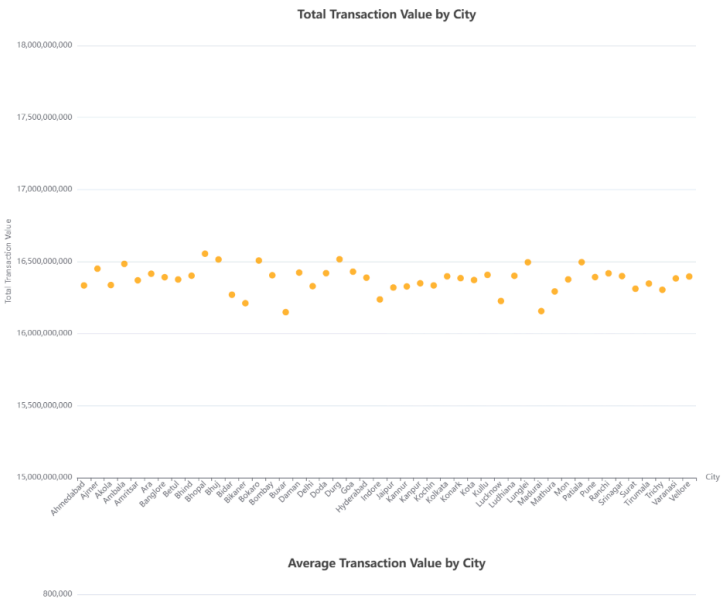
E. Results

Generated Outputs: Interactive HTML file (scatter plots + line charts).

Charts Created:

- Total Transaction Value by City
- Average and Median Transaction Value by City
- Total Number of Transactions by City
- Monthly Trends in Transaction Value
- Monthly Trends in Transaction Count

Interpretation for monthly trends: Month-to-month transaction counts fluctuate, likely reflecting seasonality or certain events. (One of the graphs, this html file is auto-generated by running the program, so it has been gitignor'd). Also, the anomaly detection output shows 0 outliers.



F. Usage Instructions

A detailed version of how to run the program is in the readme file. Please look at that instead.

G. AI Assistance & Usage

Asked ChatGPT for:

- Custom date parsing helps with chrono::NaiveDate.
- Guidance for using the charming Rust crate (for visualization).
- Small debugging assistance for iterators and string trimming.