Kurzbeschreibung

In diesem Projekt werden die Resultate von vergangenen Formel 1 Rennen ausgewertet. Die Daten sollen gesammelt werden und mit der Hilfe von verschiedenen Visualisierungen werden die Unterschiede zwischen den Fahrern und Teams dargestellt.

Datenbeschaffung

In der ersten Phase dieses Projekts werden Formel-1-Daten aus einer öffentlich zugänglichen API, in diesem Fall der Ergast API (https://ergast.com/mrd/) abgerufen. Diese Daten umfassen Rennergebnisse, Fahrerdaten, Teaminformationen und Renndetails. Das Ziel ist es, diese Daten regelmässig zu sammeln und in einem strukturierten Format zu speichern, dass für weitere Verarbeitungsschritte geeignet ist.

Da die API ein Rate-Limit von 4 Anfragen pro Sekunde hat, muss das Import-Skript entsprechend implementiert werden.

Ziele der Datenbeschaffung:

- Abrufen von Renndaten (Positionen, Zeiten, Runden) für jeden Grand Prix.
- Sammeln von Informationen über Fahrer und Konstrukteure.

Technologien:

- Python
- «requests» Bibliothek
- «json» Bibliothek

Persistieren der Daten

Die zweite Phase besteht darin, die gesammelten Daten in die Datenbank zu importieren. Hierfür wird MSSQL verwendet. Bei der Datenbankstruktur wird sich stark am Schema der Quell-API orientiert. Einige Daten werden aber nicht persistiert, da sie für die Analyse irrelevant sind.

Ziele der Archivierung:

- Erstellen der notwendigen Datenbanktabellen für die Speicherung der Formel-1-Daten.
- Importieren der JSON-Daten in die Datenbank.

Technologien:

- Python
- «SQLAlchemy» Bibliothek
- MSSQL

Auswertung und Analyse

Die dritte Phase des Projekts konzentriert sich auf die Analyse der in der Datenbank gespeicherten Daten. Verschiedene Visualisierungen werden erstellt, um die Daten verständlich und anschaulich darzustellen und miteinander zu vergleichen.

Ziele der Auswertung:

- Auswertung der Daten aus der Datenbank.
- Erstellen von Visualisierungen.
- Identifizierung von Mustern und interessanten Statistiken.

Technologien:

- Python
- «Pandas» Bibliothek
- «Matplotlib» Bibliothek

Mögliche Beispiele:

- **Durchschnittliche Platzierungen:** Bestimmung der durchschnittlichen Platzierungen von Fahrern und Teams, um ihre Leistung über eine Saison hinweg zu bewerten.
- **Anzahl der Siege:** Die Anzahl der Siege zeigt, welche Fahrer und Teams besonders dominant sind.
- **Performance-Trends:** Untersuchung von Trends, wie z.B. Verbesserung oder Verschlechterung der Performance über die Saison hinweg. Kann auf Änderungen in der Teamstrategie oder Fahrerform hinweisen.

Zusammenfassung

Dieses Projekt bietet eine umfassende Analyse der Formel-1-Daten, von der Beschaffung über die Archivierung bis hin zur detaillierten Auswertung.

Zielsetzung

Übersicht

Das Ziel der Arbeit ist es die verschiedenen Fahrer und Hersteller/Teams miteinander zu vergleichen. Mit Hilfe von Grafiken sollen anschaulich die Unterschiede zweier Fahrer in einem Team, sowie der Entwicklungsfortschritt eines Herstellers während einem Jahr aufgezeigt werden.

Muss-Ziele

Datenbeschaffung:

- **API-Verbindung**: Das Skript muss eine Verbindung zur Ergast API herstellen.
- **Rate-Limitierung**: Die Anzahl Anfragen muss limitiert werden, um nicht von der API blockiert zu werden.

Datenarchivierung:

- **Datenbank**: Eine MSSQL Datenbank mit allen benötigten Tabellen wird in der dritten Normalform erstellt.
- **Datenimport**: Die Daten werden korrekt in die Datenbank importiert.

Datenanalyse:

- **Grundlegende Analysen**: Grundlegende Analysen über die Ergebnisse können gemacht werden.
- **Diagramme**: Verschiedene Diagramme werden per Skript erstellt.

Dokumentation:

- **Help Command**: Die Verwendung der jeweiligen Skripts wird mit einem Help Command erklärt.
- **Inline-Dokumentation**: Die einzelnen Code-Abschnitte werden direkt im Code dokumentiert.

Kann-Ziele

Mehrere Saisons: Die Daten können für mehrere Saisons abgerufen und analysiert werden.

Web-Interface: Einfache Web-Oberfläche zur Anzeige der Ergebnisse und Diagramme.

Abgrenzung

In diesem Abschnitt wird erwähnt, was nicht Teil der Arbeit ist. Dies hauptsächlich, da diese Punkte zu zeitaufwändig wären.

Vorhersagen und fortgeschrittene KI-Modelle: Solche Modelle erfordern umfangreiche Datenmengen, Rechenressourcen und Zeit, die den Rahmen dieses Projekts sprengen würden.

Planung

1. **Projektsetup und Grundstruktur erstellen**
   - **Beschreibung**: Einrichtung der Entwicklungsumgebung, Erstellung der Verzeichnisstruktur und Initialisierung eines Versionskontrollsystems (Git).
   - **Dauer**: 0.5 Stunden
2. **API-Verbindung und Datenabruf**
   - **Beschreibung**: Implementierung eines Skripts zum Abrufen der Formel-1-Daten von der Ergast API.
   - **Dauer**: 0.5 Stunden
3. **Datenbankdesign**
   - **Beschreibung**: Entwurf des Datenbankschemas für Rennen, Fahrer und Teams.
   - **Dauer**: 1 Stunde
4. **Datenbank einrichten**
   - **Beschreibung**: Erstellung der Datenbank und Tabellen basierend auf dem entworfenen Schema.
   - **Dauer**: 1 Stunde
5. **Datenimport-Skript**
   - **Beschreibung**: Implementierung eines Skripts zum Importieren der JSON-Daten in die Datenbank.
   - **Dauer**: 1 Stunde
6. **Grundlegende Analysen implementieren**
   - **Beschreibung**: Implementierung grundlegender Datenanalysen wie Durchschnittsplatzierungen und Anzahl der Siege pro Fahrer und Team.
   - **Dauer**: 1-2 Stunden
7. **Visualisierungen erstellen**
   - **Beschreibung**: Erstellung von Diagrammen zur Visualisierung der Analyseergebnisse.
   - **Dauer**: 2-3 Stunden
8. **Projektdokumentation**
   - **Beschreibung**: Erstellung von Command- und Inline Dokumentation, sowie README-Datei.
   - **Dauer**: 2-3 Stunden

Hilfsmittel

Datengrundlage: Ergast API (https://ergast.com/mrd/)

Entwicklungsumgebung: IntelliJ Idea inkl. Python Plugin

Datenbankumgebung: SQL Server Management Studio

Datenbank: MSSQL Server 16

Scripting Sprache: Python 3

Python Bibliotheken (keine abschliessende Liste, nur eine Aufzählung der wichtigsten):

- requests
- pyodbc
- SQLAlchemy
- pandas
- matplotlib
- json

Versionskontrollsystem: Git