Contents

[**Project Title** 1](#_Toc187273131)

[**1.** **Skills Takeaway From This Project** 1](#_Toc187273132)

[**2.** **Domain** 1](#_Toc187273133)

[**3.** **Problem Statement** 1](#_Toc187273134)

[**4.** **Business Use Cases** 1](#_Toc187273135)

[**5.** **Approach** 1](#_Toc187273136)

[**6.** **Results** 1](#_Toc187273137)

[**7.** **Technical Tags** 1](#_Toc187273138)

### **Project Title**

**Cricsheet Match Data Analysis**

### **Skills Takeaway From This Project**

1. **Web Scraping using Selenium**
   * Automate browsing and data extraction from dynamic websites.
2. **Data Processing with Python**
   * Work with JSON files and transform data into structured formats using Pandas.
3. **Database Management with SQL**
   * Create tables, insert data, and write optimized SQL queries.
4. **Data Analysis**
   * Use SQL to derive insights from cricket data through analytical queries.
5. **Visualization with Power BI**
   * Build interactive dashboards for data-driven storytelling and insights.
6. **Data Preprocessing**
   * Clean and organize raw JSON data into meaningful structures.
7. **Automation**
   * Streamline data collection using Selenium for efficiency.

### **Domain**

**Sports Analytics / Data Analysis**

### **Problem Statement**

The objective of this project is to scrape, process, analyze, and visualize cricket match data available at [**Cricsheet**](https://cricsheet.org/matches/). Learners will use **Selenium** to scrape JSON files of different cricket match types (ODI, T20, Test), store the data in **SQL tables**, and create a **Power BI dashboard** to analyze key performance metrics. The project also includes writing **20 SQL queries** to uncover insights such as top-performing players, team statistics, and match outcomes.And an EDA using python libraries such as matplotlib,seaborn,plotly.

### 

### **Business Use Cases**

1. **Player Performance Analysis**: Analyze players' performance across different match formats (Test, ODI, T20).
2. **Team Insights**: Compare team performance over time and across match types.
3. **Match Outcomes**: Understand win/loss patterns, margin of victories, and trends.
4. **Strategic Decision-Making**: Assist analysts, coaches, and management in making informed decisions.
5. **Fan Engagement**: Present interactive dashboards for fans to explore match data and statistics.

### **Approach**

1. **Data Scraping Using Selenium**
   1. Automate navigation to [**https://cricsheet.org/matches/**](https://cricsheet.org/matches/).
   2. Use Selenium to scrape and download all the JSON files available on the page.
2. **Data Transformation**
   1. Parse JSON files using Python’s pandas library.
   2. Create separate DataFrames for **Test**, **ODI**, and **T20** match types.
3. **Database Management**
   1. Create an SQL database (e.g., MySQL or SQLite).
   2. Design separate tables for each match type: test\_matches, odi\_matches, t20\_matches.

#### **Test Match Tables**

* + - 1. **testmatchinfo**

**Table fields and datatype:**

|  |  |
| --- | --- |
| **Field** | **Type** |
| match\_id | int |
| file\_id | int |
| balls\_per\_over | int |
| dates\_0 | date |
| event\_name | varchar(255) |
| gender | varchar(10) |
| match\_type | varchar(55) |
| season | varchar(10) |
| team\_type | varchar(55) |
| teams\_0 | varchar(55) |
| teams\_1 | varchar(55) |
| toss\_winner | varchar(55) |
| toss\_decision | varchar(55) |
| winner | varchar(55) |
| by\_runs | decimal(5,0) |
| by\_innings | decimal(3,0) |
| by\_wickets | decimal(3,0) |
| outcome | varchar(55) |
| outcome\_by | varchar(55) |

* + - 1. **Testinningdata:**

**Contains following fields and datatype:**

|  |  |
| --- | --- |
| **Field** | **Type** |
| match\_id | bigint |
| batter | text |
| bowler | text |
| non\_striker | text |
| runs\_batter | bigint |
| runs\_extras | bigint |
| runs\_total | bigint |
| file\_id | bigint |
| inning | bigint |
| Over | bigint |
| delivery | bigint |
| team | text |
| season | text |
| team\_type | text |
| gender | text |
| wickets\_0\_kind | text |

* 1. **ODI Match Tables**
     1. **odimatchinfo**

**Contains following fields and datatype:**

|  |  |
| --- | --- |
| **Field** | **Type** |
| match\_id | int |
| file\_id | int |
| balls\_per\_over | int |
| dates\_0 | date |
| event\_name | varchar(255) |
| gender | varchar(10) |
| match\_type | varchar(55) |
| season | varchar(10) |
| team\_type | varchar(55) |
| teams\_0 | varchar(55) |
| teams\_1 | varchar(55) |
| toss\_winner | varchar(55) |
| toss\_decision | varchar(55) |
| winner | varchar(55) |
| by\_wickets | decimal(2,0) |
| by\_runs | decimal(3,0) |
| outcome | varchar(55) |

* + 1. **odiinningdata**

**Contains following fields and datatype:**

|  |  |
| --- | --- |
| **Field** | **Type** |
| match\_id | bigint |
| batter | text |
| bowler | text |
| non\_striker | text |
| runs\_batter | bigint |
| runs\_extras | bigint |
| runs\_total | bigint |
| file\_id | bigint |
| inning | bigint |
| Over | bigint |
| delivery | bigint |
| team | text |
| season | text |
| team\_type | text |
| gender | text |
| extras\_wides | double |
| wickets\_0\_kind | text |

* + 1. **t20matchinfo**

**Contains following fields and datatype:**

|  |  |
| --- | --- |
| **Field** | **Type** |
| match\_id | int |
| file\_id | int |
| balls\_per\_over | int |
| dates\_0 | date |
| event\_name | varchar(255) |
| gender | varchar(10) |
| match\_type | varchar(55) |
| season | varchar(10) |
| team\_type | varchar(55) |
| teams\_0 | varchar(55) |
| teams\_1 | varchar(55) |
| toss\_winner | varchar(55) |
| toss\_decision | varchar(55) |
| winner | varchar(55) |
| by\_wickets | decimal(2,0) |
| by\_runs | decimal(3,0) |
| outcome | varchar(55) |

* + 1. **t20inningdata**

**Contains following fields and datatype:**

|  |  |
| --- | --- |
| **Field** | **Type** |
| match\_id | bigint |
| batter | text |
| bowler | text |
| non\_striker | text |
| runs\_batter | bigint |
| runs\_extras | bigint |
| runs\_total | bigint |
| file\_id | bigint |
| inning | bigint |
| Over | bigint |
| delivery | bigint |
| team | text |
| season | text |
| team\_type | text |
| gender | text |
| extras\_wides | double |
| wickets\_0\_kind | text |

1. **SQL Queries for Data Analysis**SQL queries for creating tables and extracting data is written in the below python file.

**🡪 SQL\_TABLE\_&\_QUERRY.ipynb**

1. **EDA using Python**

different visualizations using libraries like matplotlib,seaborn,plotly

All visualizations are created in the query file stated above and visuals. A powerpoint presentation sing the visuals are prepared for the Test Team of Australia. Both visuals and presentation are saved in the file named: **Pandas Plots and Presentation**

1. **Power BI Dashboard**PowerBI dash board is created for better data visualization and analysis.

**04 numbers of Power BI files are created.**

* + - 1. For Players
         1. For Test Matches - test\_cricsheet\_inning .pbix
         2. For ODI Matches - odi\_cricsheet\_inning.pbix
         3. For T20 Matches - t20\_cricsheet\_inning.pbix
      2. For Matches
         1. For Match Summary of all formats - match\_info\_all\_format.pbix

All .pbix files are saved in the folder **Power BI Files** along with the screenshots of some dashboards.

### **Results**

* Automated scraping of JSON files from Cricsheet.
* Structured SQL database with separate tables for Test, ODI, and T20 matches.
* SQL queries to analyze player and team performance metrics.
* A dynamic Power BI dashboard to present insights visually.

### **Technical Tags**

* **Matplotlib,Seaborn**
* **Web Scraping**
* **Selenium**
* **Python**
* **Pandas**
* **SQL (MySQL/SQLite)**
* **Power BI**
* **JSON**
* **Data Analysis**