**REPORT ON INTERIM PROJECT**

**PROJECT USING SQL SERVER INTEGRSTION SERVICES**

**AND MICROSOFT POWER BI**

BY

**MOHAMMAD IMRAZ UDDIN**

**EMPLOYEE ID.-2265033**

**INTRODUCTION**

**Microsoft SQL Server Integration Services** (**SSIS**)

Microsoft SQL Server Integration Services (SSIS) is a component of the Microsoft SQL server database software that can be used to perform a broad range of data migration tasks.

SSIS is a platform for data integration and workflow application. It features a data warehouse tool used for data the ETL (Extract, transform and Load)  The tool may also be used to automate maintenance of SQL Server databases and updates to multidimensional.

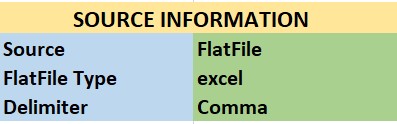
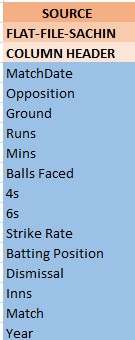
SQL Server Integration Services is a platform for building enterprise-level data integration and data transformations solutions. Use Integration Services to solve complex business problems by copying or downloading files, loading data warehouses, cleansing and mining data, and managing SQL Server objects and data.

**MICROSOFT POWER BI**

Power BI is a powerful cloud based and On-premises suit of business analytics tools developed by Microsoft to make it easy to combine data from multiple sources, analyze and visualize information and share insights. It also provides self-service [business intelligence](https://en.wikipedia.org/wiki/Business_intelligence) capabilities, where end users can create reports and dashboards by themselves, without having to depend on [information technology](https://en.wikipedia.org/wiki/Information_technology) staff or [database administrators.](https://en.wikipedia.org/wiki/Database_administrator) Power BI allows us to share dashboards and reports with the right people.

**Power BI** is an interactive data visualisation software product developed by Microsoft with a primary focus on business intelligence. It is part of the  Power BI is a collection of software services, apps, and connectors that work together to turn unrelated sources of data into coherent, visually immersive, and interactive insights. Data may be input by reading directly from a [database](https://en.wikipedia.org/wiki/Database), webpage, or structured files such as spreadsheets, csv, xml and json.

**SOURCE TAKEN-FLAT FILE**

 ****

**SOURCE-FLAT FILE**

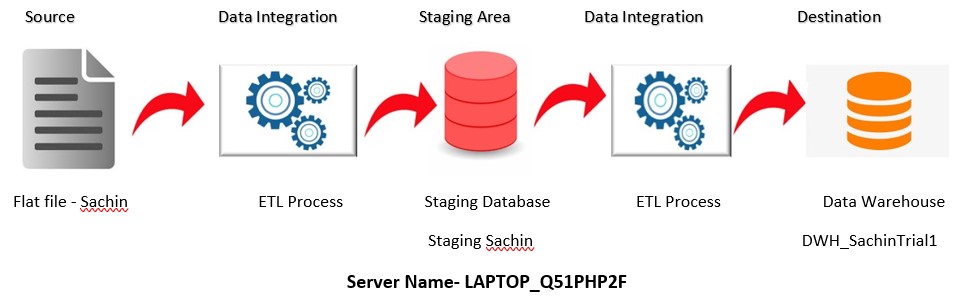
A Flat File source is a data flow component that uses metadata defined by a Flat File connection manager. This metadata specifies the format and structure of the data to be extracted from the flat file by a transform process. The Flat File source extracts data from a single flat file, using the format definitions in the Flat File connection manager.

**STAGING DATABASE(AREA)**

A **staging area**, or **landing zone**, is an intermediate storage area used for data processing during the ETL (Extract, transform and Load) process. The data staging area sits between the data source(s) and the data target(s), which are often Data warehouse, Data marts or other data repositories.

Data staging areas are often transient in nature, with their contents being erased prior to running an ETL process or immediately following successful completion of an ETL process. There are staging area architectures, however, which are designed to hold data for extended periods of time for archival or troubleshooting purposes.

**PROJECT ARCHITECTURE**

****

The image is the Architecture of the Project.

--- The Source Flat File -Sachin is taken to perform ETL Process and Storing it into a

Staging Database called “Staging Sachin”.

--- Then the again the database taken to perform the ETL Process and load it in a

Data warehouse called “DWH\_sachintrial1”.

--- In DWH\_SachinTrial1 the Dimension Tables And Fact Table Created and then with

The help of Power Bi the report is created and save as .Pbix file

**DATA WAREHOUSE**

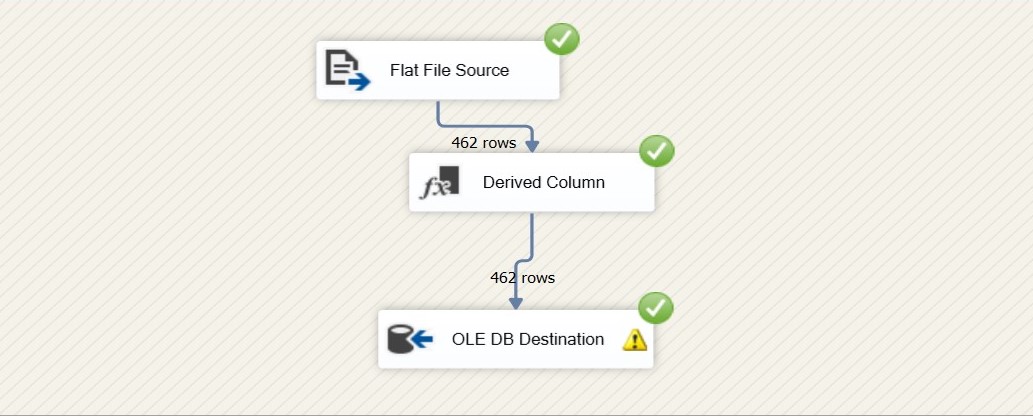
A **data warehouse** (**DW** or **DWH**), also known as an **enterprise data warehouse** (**EDW**), is a system used for reporting and data analysis and is considered a core component of business intelligence. Data warehouses are central of integrated data from one or more disparate sources. They store current and historical data in one single place that are used for creating analytical reports for workers throughout the enterprise. This is beneficial for companies as it enables them to interrogate and draw insights from their data and make decisions.

The data stored in the warehouse is uploaded from the operational system (such as marketing or sales). The data may pass through an operational data store and may require data cleansing for additional operations to ensure data quality before it is used in the data warehouse for reporting.

 Extract, transform and Load (ETL) and Extract, Load and transform (ELT) are the two main approaches used to build a data warehouse system.

**CREATING STAGING DATABASE**

**Loading the Flat File Source into OLE DB (Database) Destination for Staging Table**



--In the above figure the source flat file is loaded into derived column where the empty places are replaced by NULL values and then loaded into the OLE database.

**SCRIPT COMPONENT (ETL TOOL USED)**

The Script component extends the data flow capabilities of Microsoft Integration Services packages with custom code written in Microsoft Visual Basic or Microsoft Visual C# that is compiled and executed at package run time. The Script component simplifies the development of a custom data flow source, transformation, or destination when the sources, transformations, and destinations included with Integration Services do not fully satisfy your requirements. After you configure the component with the expected inputs and outputs, it writes all the required infrastructure code for you, letting you focus exclusively on the code that is required for your custom processing

**FACT TABLE AND DIMENSION TABLES**

Fact tables are the core tables of a data warehouse. They contain quantitative information, commonly associated with points in time. They are used in trends, comparisons, aggregations, and groupings. They feed analysis and visualization tools to allow insights to be discovered about the functional area.

Dimensions, on the other hand, are collections of reference information about the facts in a data warehouse. Dimensions categorize and describe the facts recorded in a data warehouse to provide meaningful, categorized, and descriptive answers to business questions.

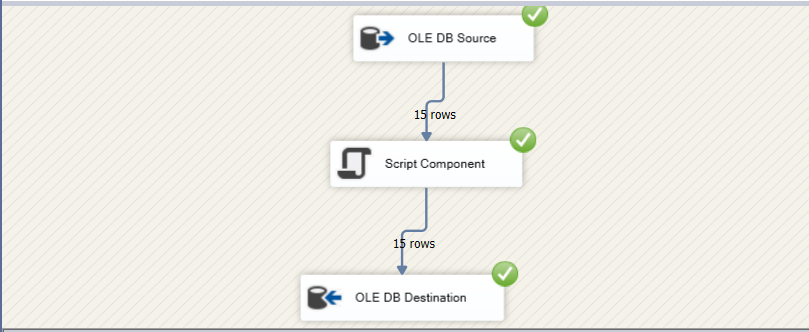
In the design of a data warehouse, it is common to create dimension tables first and then create fact tables by relating them to the dimension tables through foreign keys

**CREATING DIMENSION TABLE**

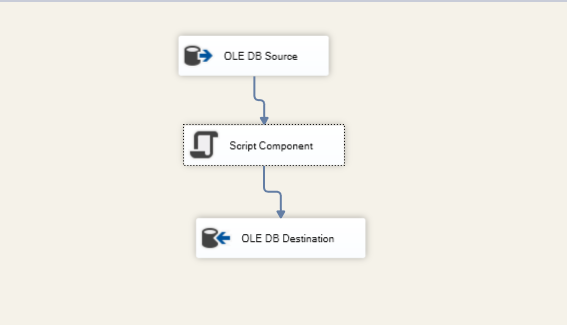
**-- OPPOSITION as (Dim\_Opposition)**

We are using **Script Component** to add keys (ID) for Dim\_Opposition Dimension Table

In Script Component with the help of C# we write a code to create keys of Auto Increment for Dimension Table as Opposition\_ID

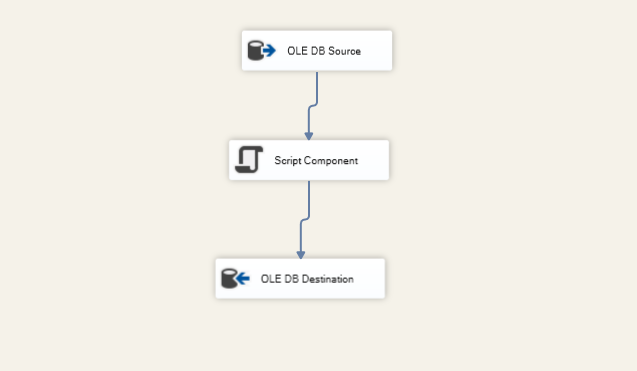


**-- CREATING DIMENSION TABLE FOR GROUND**



Taking Staging Sachin as source and from that creating Ground/ region Dimension table and using Script component for generating keys/ID for Ground/Region dimension Table

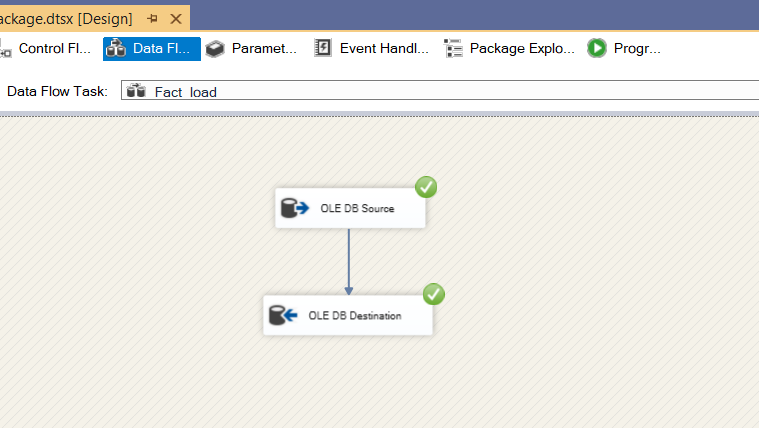
**---CREATING DIMENSION TABLE FOR MATCH**



Taking Staging Sachin as source and from that creating Match Dimension table and using

Script component for generating keys/ID for Match dimension Table

**--CREATING FACT TABLE**



The fact table is created by taking source as staging database and removing columns which are not

Needed loading it into destination

**CONTROL FLOW**

The below figure, using the execute SQL Task is used to insert the keys in the staging table from each dimension table and fact table

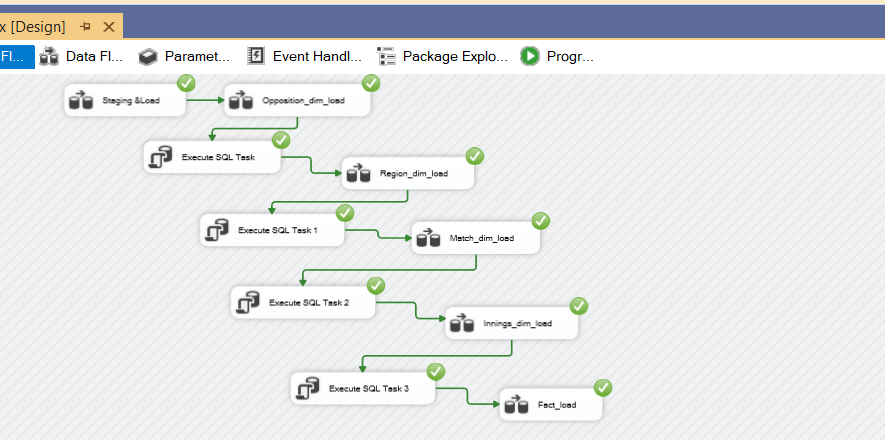
**EXECUTE SQL TASK (ETL TOOL USED)**

The Execute SQL task runs SQL statements or stored procedures from a package. The task can contain either a single SQL statement or multiple SQL statements that run sequentially. You can use the Execute SQL task for the following purposes:

----Truncate a table or view in preparation for inserting data.

----Create, alter, and drop database objects such as tables and views.

----Re-create fact and dimension tables before loading data into them.



Execute SQL Task --------- Updating Opposition ID into Staging Table

Execute SQL Task 1---------- Updating Ground ID into Staging Table

Execute SQL Task 2----------Updating Match ID into Staging Table

Execute SQL Task 3---------- Updating Innings ID into Staging Table

Then loading data into Fact table by removing some not needed columns

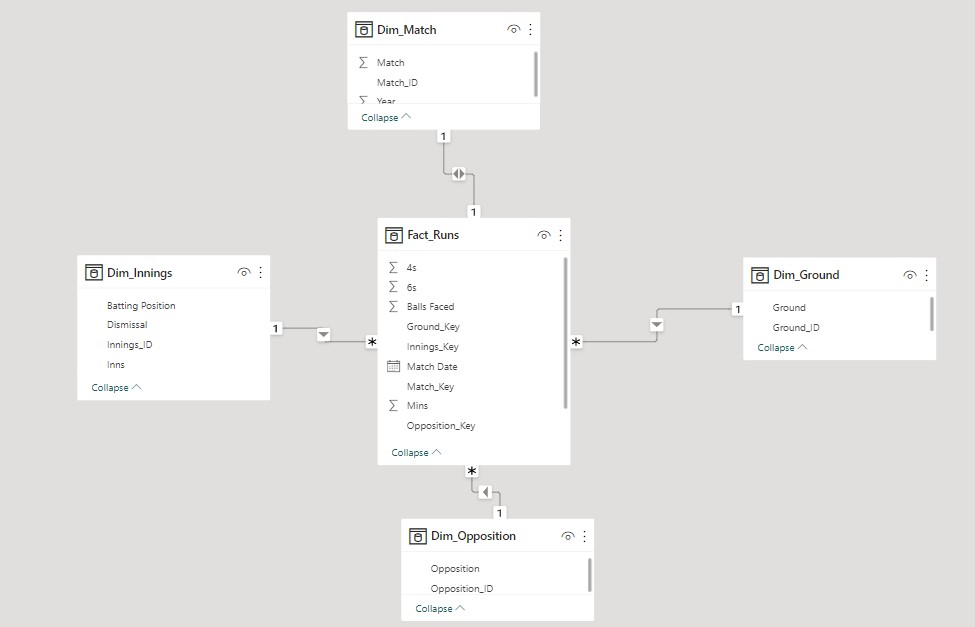
**POWER BI**

Power BI is a powerful cloud based and On-premises suit of business analytics tools developed by Microsoft to make it easy to combine data from multiple sources, analyze and visualize information and share insights. It also provides self-service [business intelligence](https://en.wikipedia.org/wiki/Business_intelligence) capabilities, where end users can create reports and dashboards by themselves, without having to depend on [information technology](https://en.wikipedia.org/wiki/Information_technology) staff or [database administrators.](https://en.wikipedia.org/wiki/Database_administrator) Power BI allows us to share dashboards and reports with the right people.

**STAR SCHEMA**

 the **star schema** is the simplest style of data mart schema and is the approach most widely used to develop data warehouses and dimensional data marts. The star schema consists of one or more fact tables referencing any number of dimension tables

\*The figure below is Star Schema model for Four Dimension Tables and One Fact Table



The above figure is the Star schema of the Dimension tables and fact table which we created using the Sql Server database with some ETL (Extract Transform and Load) in Visual Studio.

---The table in the middle called Fact Runs is the Fact Table

--- the Tables Around the Fact Table are Dimension Table they are

\*Dim\_Match

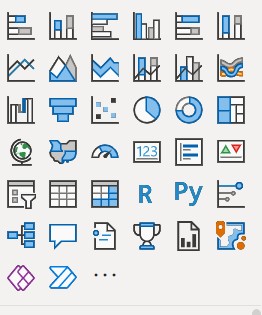
\*Dim\_Inning

\*Dim\_Ground

\*Dim\_Opposition

The Fact Table and Dimension Table are connected through Foreign Keys

**VISUALIZATIONS IN POWER BI**

****

The above shown image are some of Visualizations used in Power Bi according to the

Business Requirements like-

**AREA CHARTS**

The basic area chart is based on the line chart with the area between the axis and line filled in. Area charts emphasize the magnitude of change over time, and can be used to draw attention to the total value across a trend.

**GAUGE CHART**

A radial gauge chart has a circular arc and displays a single value that measures progress toward a goal. The goal, or target value, is represented by the line (needle). Progress toward that goal is represented by the shading. And the value that represents that progress is shown in bold inside the arc. All possible values are spread evenly along the arc, from the minimum (left-most value) to the maximum (right-most value).

**DOUGHTNUT CHART**

Doughnut charts are similar to pie charts. They show the relationship of parts to a whole. The only difference is that the center is blank and allows space for a label or icon.

**Stacked Bar Chart**

A stacked chart is a form of bar chart that shows the composition and comparison of a few variables, either relative or absolute, over time. Also called a stacked bar or column chart, they look like a series of columns or bars that are stacked on top of each other.

**Clustered Bar Chart**

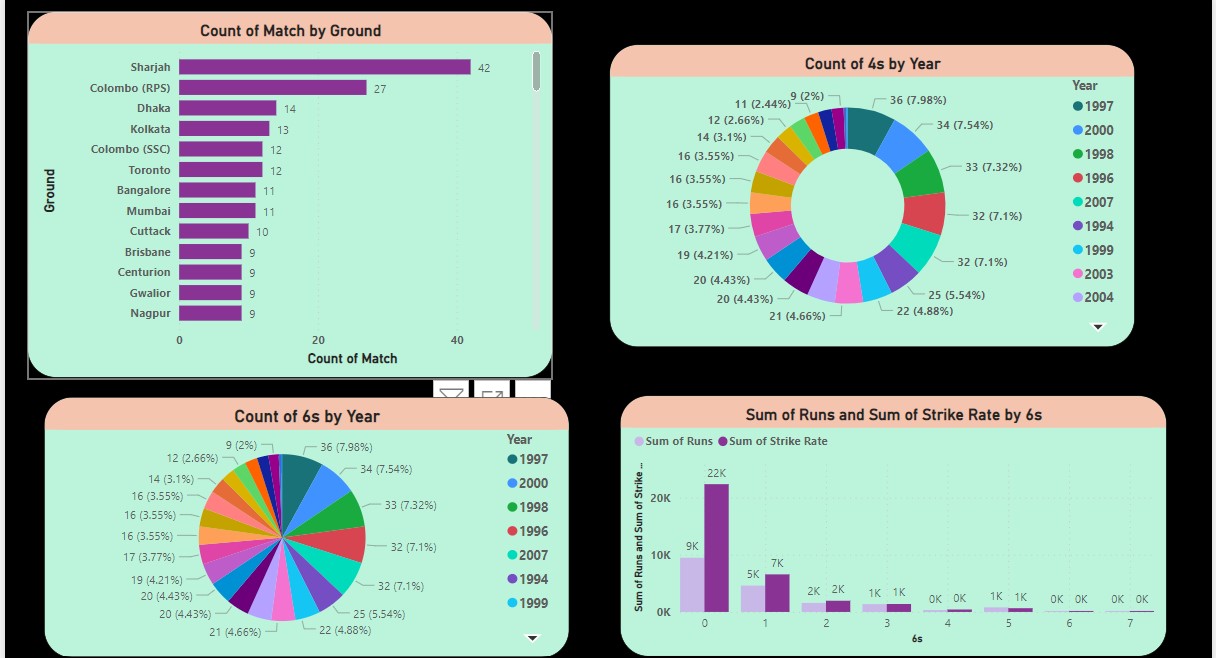
Power BI Clustered Bar charts display values (i.e. measures) where the length of the bar or column is proportional to the data. Clustered Bar charts are means to display a measure against a single dimension. A second dimension can break the primary dimension further more

**Pie Chart**

Pie Chart in Power BI is a built-in visualization chart available with all versions of Power BI. The pie chart is a round-shaped circle chart where each category data set is shown in a pie shape based on the value of each data label.

**VISUALS CREATED ARE**

**REPORT 1**



**In the above report there four visualisations**

**1.Stacked Bar Chart Is Used**

Count of Match by Ground—It shows that the total Matches by Sachin in each Ground

**2.Donut Chart Is Used**

Count of Fours(4s) by year----- It shows that the total fours(4s) hit by Sachin in each Year

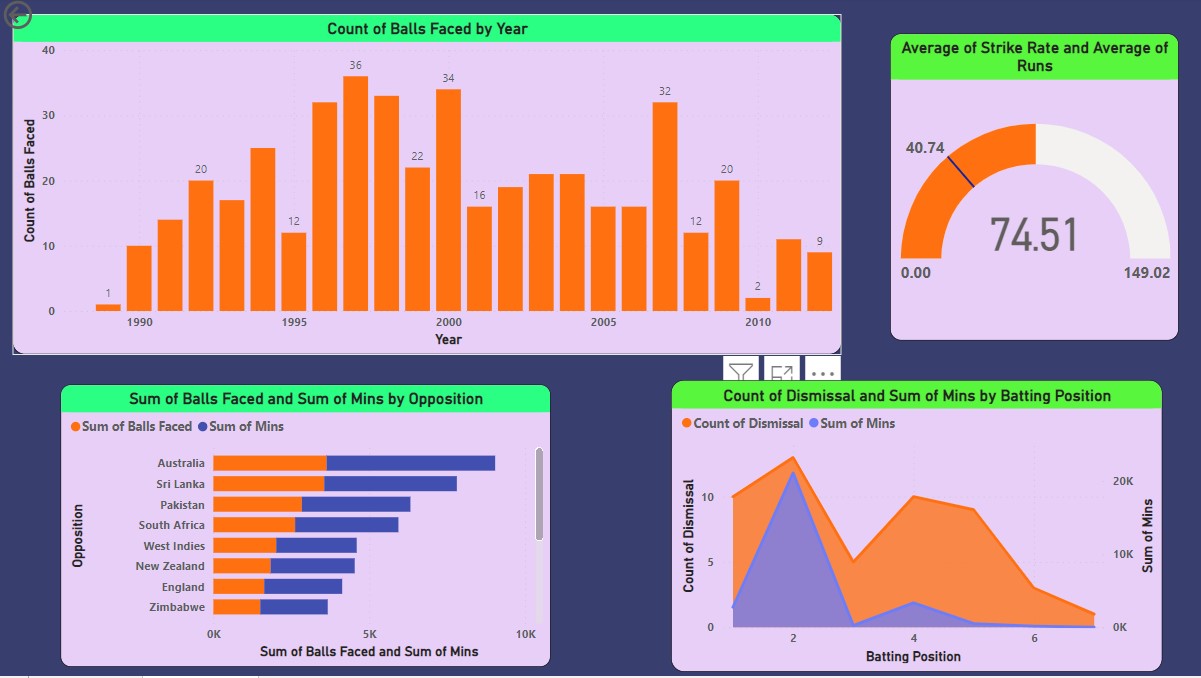
**3.Pie-Chart is Used**

Count of Sixes(6s) by Year----- It shows that the total Sixes(6s) hit by Sachin in each Year

**4.Clustered column Chart is Used**

Sum of Runs and Sum of Strike Rate by Sixes (6s) ---- It shows sum of Runs and Strike Rate of Sachin when he hit certain Number of Sixes

**REPORT 2**



**In the above report there four visualisations**

**1.Staked Column Chart is Used**

Count of Balls Faced by Year----- It shows that the Number of Balls Faced By Sachin Each Year

**2. Gauge is used**

Average Strike rate and Average Runs of Sachin

**3.Staked bar Chart is Used**

Sum of Balls Faced and Sum of Mins by Opposition ------- It shows the Sum of Balls faced and Sum Of Minutes spent on Ground against each Opponent Country By Sachin

**4.Area Chart is Used**

Count of Dismissal and Sum of Mins by Batting Position ----- It shows Number of Dismissals and Minutes Spent on Ground at Playing each Batting Position.