Q. **Design Data Warehouses For Given Below Products:**

**Note :** While designing any Data Warehouse make sure to cover given below points. a. Design Fact & Dimension tables

b. Create meaningful Primary & Foreign keys

c. Try to follow Star/SnowFlake Schema Design

d. Try to write few SQL queries to generate insightful business metrics (This is the critical point because you need to understand the Data & Business both)

1. Design a Data Warehouse for IPL Cricket Tournament **(Asked in Flipkart Interview for Senior Data Engineer role)**

2. Design a Data Warehouse for Food delivery app like Swiggy, Zomato **(Asked in Grab for Data Engineer role)**

3. Design a Data Warehouse for cab ride service like Uber, Lyft **(Asked in Google for Data Engineer role)**

4. Design a Data Warehouse for Restaurent table booking app like Dineout **(Asked in McKinsey for Consultant Data Engineer role)**

5. Design a Data Warehouse for Covid Vaccination Application **(Asked in Livsapce for Data Engineer role)**

A)

**1. Designing a Data Warehouse for IPL Cricket Tournament:**

To design a Data Warehouse for the IPL Cricket Tournament, we need to consider the following points:

a. Design Fact & Dimension tables:

- Fact Table: We can have a "Match" fact table that includes information about each match played in the tournament. It can contain attributes such as match ID, date, venue, teams, scores, player performance, and other relevant statistics.

- Dimension Tables: We can have dimension tables for players, teams, venues, and dates. These tables would store additional information about each dimension, such as player details (name, age, nationality), team details (name, coach, home city), venue details (name, city, capacity), and date details (day, month, year).

b. Primary & Foreign keys:

- In the fact table, the primary key would be the match ID, which uniquely identifies each match.

- Foreign keys would link the fact table to dimension tables. For example, the team ID and player ID would be foreign keys in the fact table, referencing the team and player dimension tables.

c. Star/Snowflake Schema Design:

- The design can follow a star schema where the fact table is at the center, and dimension tables are connected to it.

- Alternatively, a snowflake schema can be used where dimension tables are further normalized by splitting them into additional tables to reduce redundancy.

d. SQL queries for business metrics:

- To generate insightful business metrics, we can write SQL queries to calculate statistics such as average score per match, player performance rankings, team rankings, highest run-scorer, highest wicket-taker, and other relevant metrics that provide insights into the performance of teams and players.

**2. Designing a Data Warehouse for Food delivery app like Swiggy, Zomato:**

For designing a Data Warehouse for a food delivery app, we should consider the following aspects:

a. Design Fact & Dimension tables:

- Fact Table: We can have an "Order" fact table that includes information about each food order placed through the app. It can contain attributes such as order ID, customer details, restaurant details, order items, total amount, delivery details, and timestamps.

- Dimension Tables: We can have dimension tables for customers, restaurants, menu items, delivery agents, and dates. These tables would store additional information about each dimension, such as customer details (name, address, contact), restaurant details (name, location, cuisine), menu item details (name, price, ingredients), delivery agent details (name, contact), and date details (day, month, year).

b. Primary & Foreign keys:

- The primary key in the fact table would be the order ID, uniquely identifying each order.

- Foreign keys would establish relationships between the fact table and dimension tables. For example, customer ID, restaurant ID, and delivery agent ID would be foreign keys in the fact table, referencing the respective dimension tables.

c. Star/Snowflake Schema Design:

- The design can follow a star schema, with the order fact table at the center and dimension tables connected to it, representing the different aspects of the order.

- Alternatively, a snowflake schema can be used if there is a need for further normalization of the dimension tables.

d. SQL queries for business metrics:

- SQL queries can be written to generate business metrics such as total revenue, popular cuisines, customer preferences, average delivery time, customer satisfaction ratings, top-selling menu items, and other relevant metrics that provide insights into the performance and trends of the food delivery app.

**3. Designing a Data Warehouse for a cab ride service like Uber, Lyft:**

To design a Data Warehouse for a cab ride service, considering platforms like Uber or Lyft, the following points should be taken into account:

a. Design Fact & Dimension tables:

- Fact Table: We can have a "Trip" fact table that includes information about each trip taken through the service. It can contain attributes such as trip ID, driver details, passenger details, pickup location, drop-off location, distance, fare, duration, and timestamps.

- Dimension Tables: We can have dimension tables for drivers, passengers, locations, and dates. These tables would store additional information about each dimension, such as driver details (name, rating, vehicle details), passenger details (name, contact), location details (name, coordinates, city), and date details (day, month, year).

b. Primary & Foreign keys:

- The primary key in the fact table would be the trip ID, uniquely identifying each trip.

- Foreign keys would link the fact table to dimension tables. For example, driver ID and passenger ID would be foreign keys in the fact table, referencing the driver and passenger dimension tables.

c. Star/Snowflake Schema Design:

- The design can follow a star schema, where the trip fact table is at the center, and dimension tables are connected to it.

- Alternatively, a snowflake schema can be used if further normalization is required, such as splitting location details into separate tables.

d. SQL queries for business metrics:

- SQL queries can be written to generate business metrics such as total trips, average trip fare, driver ratings, popular pickup/drop-off locations, busiest days/times, driver utilization rates, customer satisfaction ratings, and other relevant metrics that provide insights into the performance and efficiency of the cab ride service.

**4. Designing a Data Warehouse for a restaurant table booking app like Dineout:**

To design a Data Warehouse for a restaurant table booking app like Dineout, the following considerations should be taken into account:

a. Design Fact & Dimension tables:

- Fact Table: We can have a "Reservation" fact table that includes information about each table reservation made through the app. It can contain attributes such as reservation ID, customer details, restaurant details, reservation date and time, table size, and special requests.

- Dimension Tables: We can have dimension tables for customers, restaurants, table details, and dates. These tables would store additional information about each dimension, such as customer details (name, contact), restaurant details (name, location, cuisine), table details (table number, seating capacity), and date details (day, month, year).

b. Primary & Foreign keys:

- The primary key in the fact table would be the reservation ID, uniquely identifying each reservation.

- Foreign keys would link the fact table to dimension tables. For example, customer ID and restaurant ID would be foreign keys in the fact table, referencing the customer and restaurant dimension tables.

c. Star/Snowflake Schema Design:

- The design can follow a star schema, with the reservation fact table at the center and dimension tables connected to it.

- Alternatively, a snowflake schema can be used if there is a need for further normalization, such as splitting restaurant details into separate tables.

d. SQL queries for business metrics:

- SQL queries can be written to generate business metrics such as total reservations, popular restaurants, customer preferences, peak reservation times, average table turnover time, customer feedback ratings, and other relevant metrics that provide insights into the performance and trends of the restaurant table booking app.

**5. Designing a Data Warehouse for a Covid Vaccination Application:**

To design a Data Warehouse for a Covid Vaccination Application, the following aspects should be considered:

a. Design Fact & Dimension tables:

- Fact Table: We can have a "Vaccination" fact table that includes information about each vaccination administered through the application.

It can contain attributes such as vaccination ID, patient details, healthcare provider details, vaccination date and time, vaccine type, dosage, and location.

- Dimension Tables: We can have dimension tables for patients, healthcare providers, vaccine details, locations, and dates. These tables would store additional information about each dimension, such as patient details (name, age, contact), healthcare provider details (name, location), vaccine details (name, manufacturer, dosage schedule), location details (name, city, state), and date details (day, month, year).

b. Primary & Foreign keys:

- The primary key in the fact table would be the vaccination ID, uniquely identifying each vaccination.

- Foreign keys would link the fact table to dimension tables. For example, patient ID and healthcare provider ID would be foreign keys in the fact table, referencing the patient and healthcare provider dimension tables.

c. Star/Snowflake Schema Design:

- The design can follow a star schema, with the vaccination fact table at the center and dimension tables connected to it.

- Alternatively, a snowflake schema can be used if there is a need for further normalization, such as splitting location details into separate tables.

d. SQL queries for business metrics:

- SQL queries can be written to generate business metrics such as total vaccinations administered, vaccine distribution by manufacturer, vaccination coverage by location, vaccination rates over time, average waiting times, adverse event reporting, and other relevant metrics that provide insights into the vaccination program's effectiveness and performance.