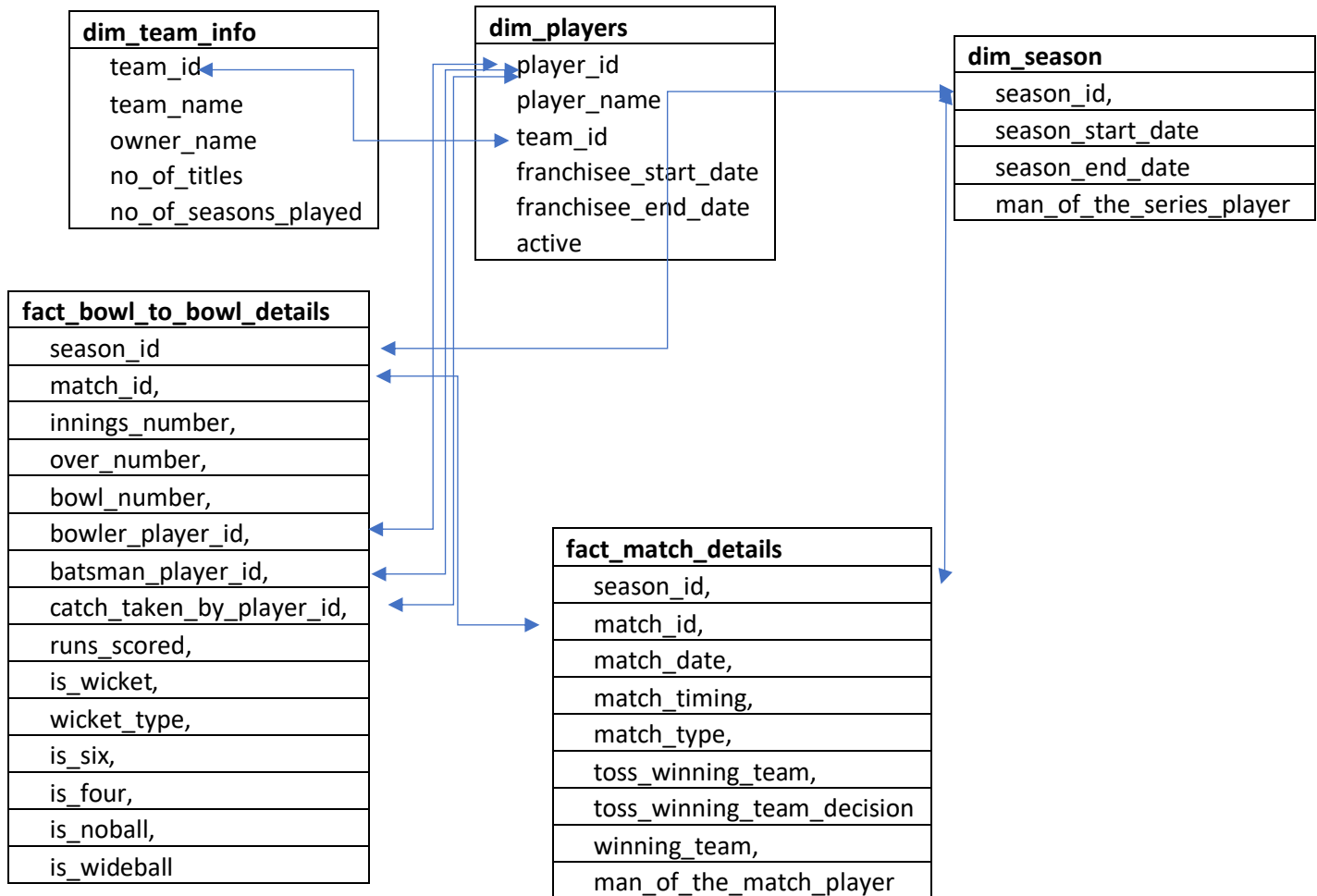


Data Warehousing Design for the IPL Tournament



Queries:

Q1: Find out the top 5 run scorers in the season held in the year 2010.

Select

b.player_name,

sum(runs_scored)

From **fact_bowl_to_bowl_details** a

Join dim_players b on a.batsman_player_id=b.player_id

Join dim_season c on a.season_id= c.season_id and year(c. season_start_date)=2010

Where a.runs_scored > 0

Group by b.player_name

Order by sum(runs_scored) desc

Limit 5;

Data Warehousing Design for the IPL Tournament

Q2: Find out the who won purple cap[The bowler who took most no of wickets] in year 2016. .

Select

b.player_name,

count(1)

From **fact_bowl_to_bowl_details** a

Join dim_players b on a. bowler_player_id =b.player_id

Join dim_season c on a.season_id= c.season_id and year(c. season_start_date)=2016

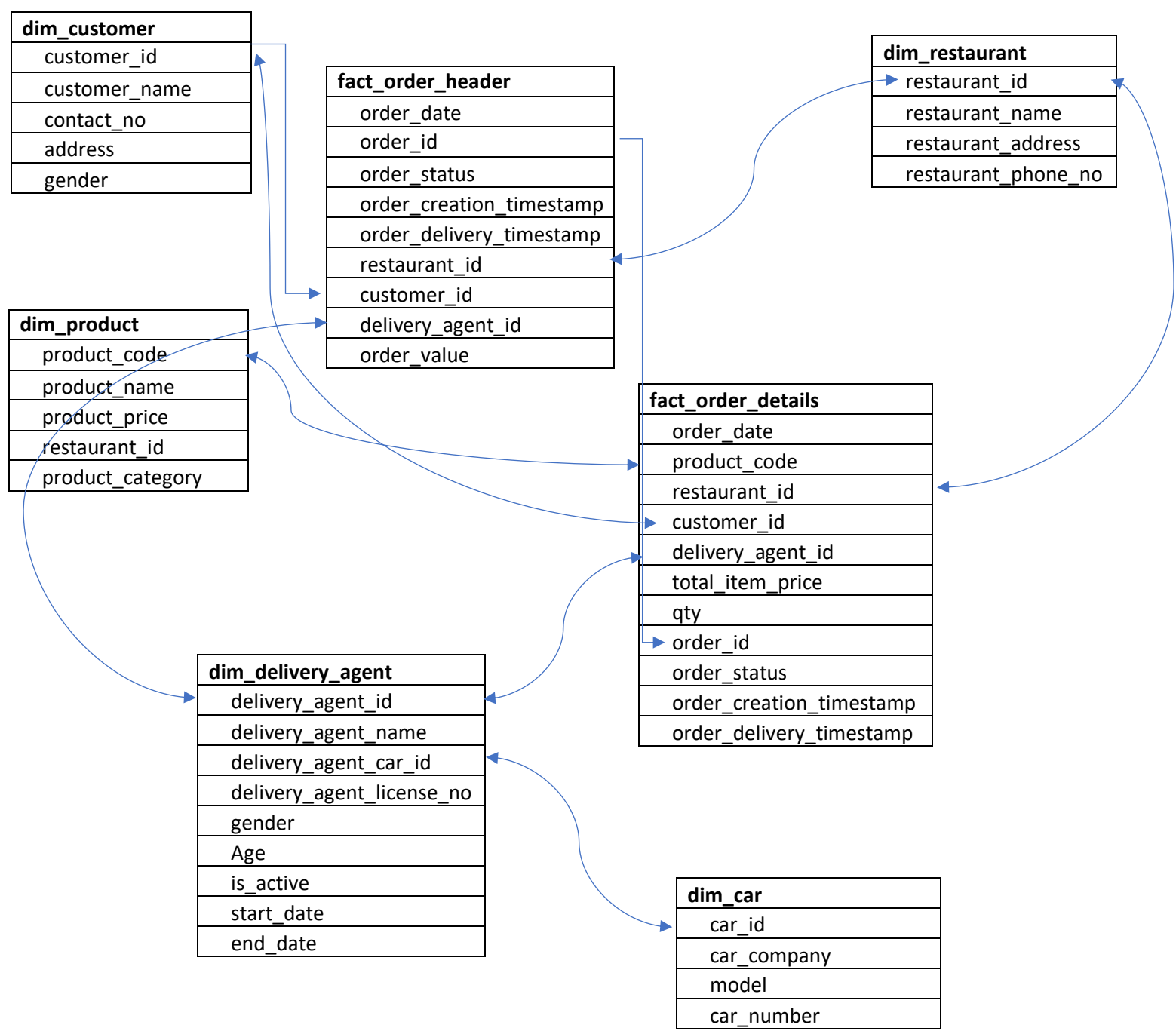
Where is_wicket=1 and wicket_type<>'Run Out' –since run out is not counted as bowlers wicket

Group by b.player_name

Order by count(1) desc

Limit 1;

Data Warehousing Design for the Swiggy /Zomato Order System



Data Warehousing Design for the Swiggy /Zomato Order System

Explanation:

We have created below tables as a part of DWH design:

Table Name	Purpose
Dim_customer:	Stores the details at customer level.
Dim_product:	Product master. Stores the details at product level. Also stores the foreign key of the dim_restaurant so that we will come to know that product belongs to which restaurant.
Dim_Restaurant:	Stores details of the restaurants listed on our app.
dim_delivery_agent	Stores the details of our deliver agents/partners. We will get active records based on is_active flag.
Dim_Car	Stores the details of the cars/bike used by delivery agnts.
fact_order_header	Fact table which is at order_id level stores the details of each order.
fact_order_details	Fact table which stores the data at order + product level. Gives us the entire info related or order such as order_date, product ordered, qty Etc.

Sample Queries:

Q1. Find out the total sale ,count of orders during Jan-2021.

Select

sum(order_value) as total_sales,

count(order_id)

From fact_order_header a

Where a. order_date between '2021-01-01' and '2021-01-31' and order_status='completed'

Data Warehousing Design for the Swiggy /Zomato Order System

Q2. Find out the top 5 delivery partner's based on number of deliveries done in year 2022

Select

b. delivery_agent_id,

b. delivery_agent_name,

total_orders

from (

 Select

 b. delivery_agent_id,

 b. delivery_agent_name,

 count(order_id) as total_orders,

 row_number() over (partition by delivery_agent_id order by count(order_id) desc) as "rank"

 From fact_order_header a

 join **dim_delivery_agent b** on a. delivery_agent_id=b. delivery_agent_id and b.is_active=1

 where year(a.order_date)=2022

 group by

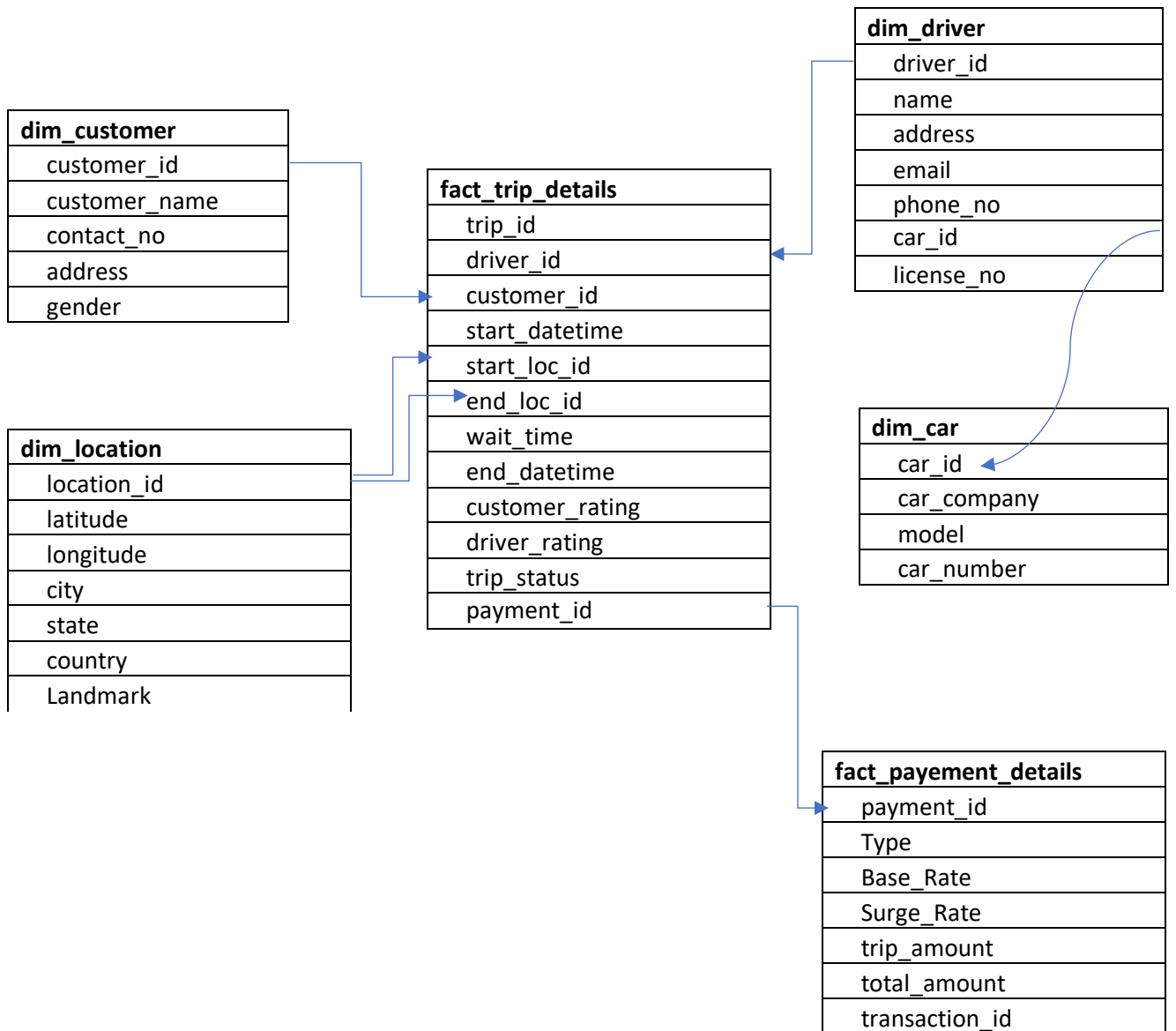
 b. delivery_agent_id,

 b. delivery_agent_name,

) inner_table

WHERE inner_table.rank<=5;

Data Warehousing Design for the Car Services like Uber



Data Warehousing Design for the Car Services like Uber

Explanation:

We have created below tables as a part of DWH design:

Table Name	Purpose
Dim_customer:	Stores the details at customer level.
Dim_location:	Location Master which stores the location details such as latitude, longitude, city, state, country, landmark etc.
Dim_Restaurant:	Stores details of the restaurants listed on our app.
Dim_Driver	Stores the Driver Details such as Name, Email, Phone, License_id, car_id etc.
Dim_Car	Stores the details of the cars used by driver.
Fact_trip_Details	Fact tables which stores the details of the trip at trip_id level.
Fact_Payment_Details	Fact table which stores the data of payment made by the customer for their trips.

Sample Queries:

Q1. Find out the total trips and avg trip price during 2021.

Select

avg(total_amount) as total_sales,

count(a.trip_id) as total_trips

From fact_trip_details a

Join fact_payment_details b on a.payment_id=b.payment_id

Where year(start_datetime)=2021 and a.trip_status='completed'

Data Warehousing Design for the Car Services like Uber

Q2. Find out the top 5 cities where most number of customer travelled in 2022

Select

city,

count_of_trips

from (

 Select b.city,

 Count(1) as count_of_trips,

 Row_number() over (order by count(1) desc) as "rank"

 From fact_trip_details a

 Join dim_location b on a.end_loc_id=b.location_id

 Where a.trip_status='completed' and year(start_datetime)=2022

 Group by b.city

) inner_table

Where inner_table.rank<=5

Q2. Find out the top 5 delivery partner's based on number of deliveries done in year 2022

Select

b. delivery_agent_id,

b. delivery_agent_name,

total_orders

from (

 Select

 b. delivery_agent_id,

 b. delivery_agent_name,

 count(order_id) as total_orders,

Data Warehousing Design for the Car Services like Uber

row_number() over (partition by delivery_agent_id order by count(order_id) desc) as "rank"

From fact_order_header a

join **dim_delivery_agent b on a.** delivery_agent_id=b. delivery_agent_id and b.is_active=1

where year(a.order_date)=2022

group by

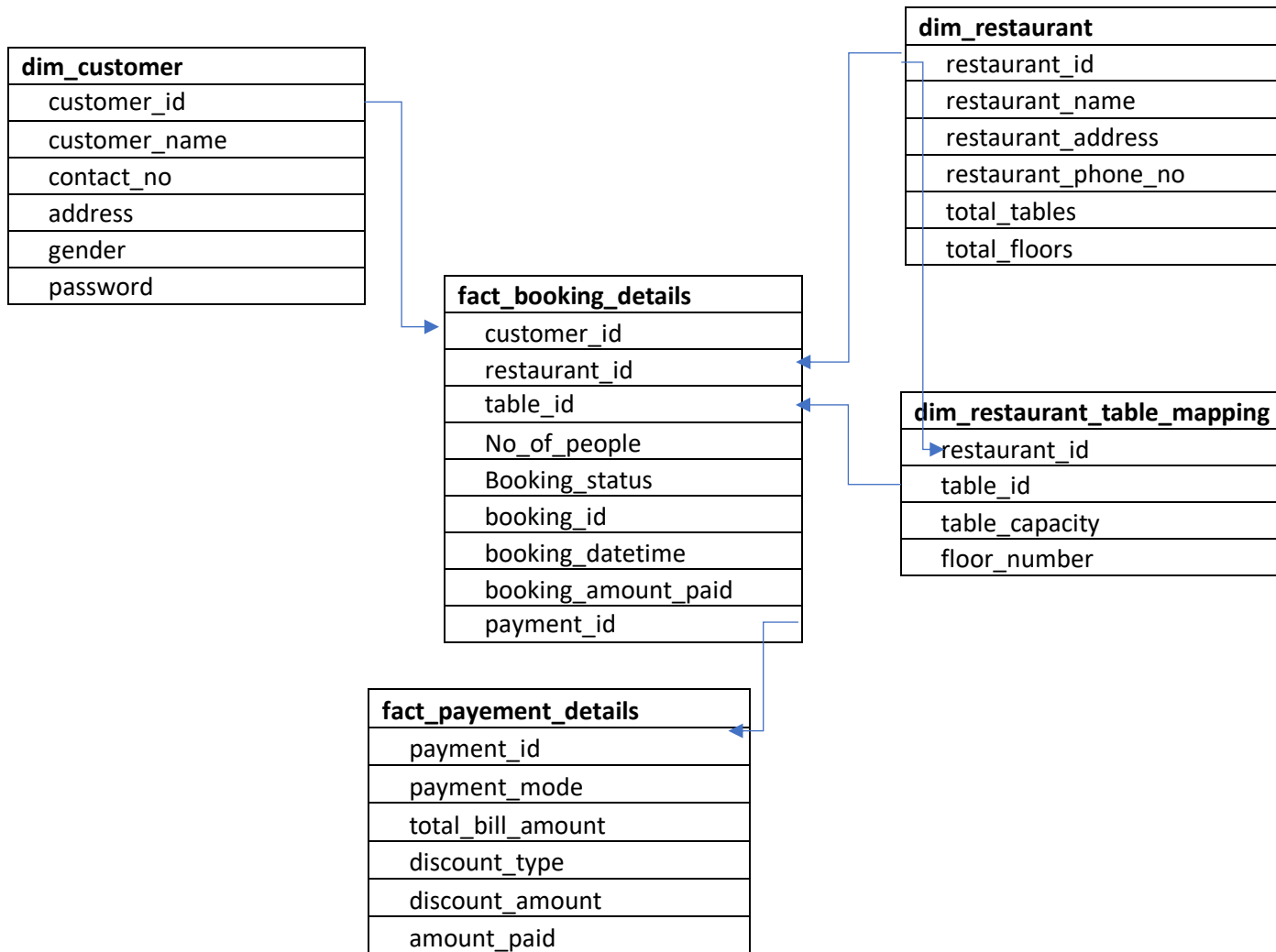
b. delivery_agent_id,

b. delivery_agent_name,

) inner_table

WHERE inner_table.rank<=5;

Data Warehousing Design for Restaurant Table Booking App



Explanation:

We have created below tables as a part of DWH design:

Table Name	Purpose
Dim_customer:	Stores the details at customer level.
Dim_Restaurant_Table_Mapping:	Dimension table which store the mapping between the table_id and restaurant_id's. Also gives us that the current_table is present at which floor in the given restaurant_id and tables seating capacity.
Dim_Restaurant:	Stores details of the restaurants listed on our app.
Fact_booking_details	Fact table that stores the booking details of the tables.
fact_payement_details	Fact tables that captures the details regarding the payment made by the customer against booking of the table

Data Warehousing Design for Restaurant Table Booking App

Questions:

Q1: Find out the top 5 restaurants which have been booked more than the 90 % capacity of their total seating capacity on 31st Dec 2022[Assuming today is 31st Dec].

Select

Restaurant_id,

Percentage_Booking_Done

From

(

Select

a.Restaurant_id,

a. total_seats_booked*100.0/b. total_capacity as "Percentage_Booking_Done",
row_number() over (order by "Percentage_Booking_Done" desc) as "Rank"

from

(

Select

Restaurant_id,

Sum(no_of_people) as total_seats_booked

From

fact_booking_details a

Where a.booking_status='completed' and date(a. booking_datetime)='2022-12-31'

Group by Restaurant_id

) a

join

(

select

restaurant_id,

sum(total_capacity) as total_capacity

From

dim_restaurant_table_mapping

group by

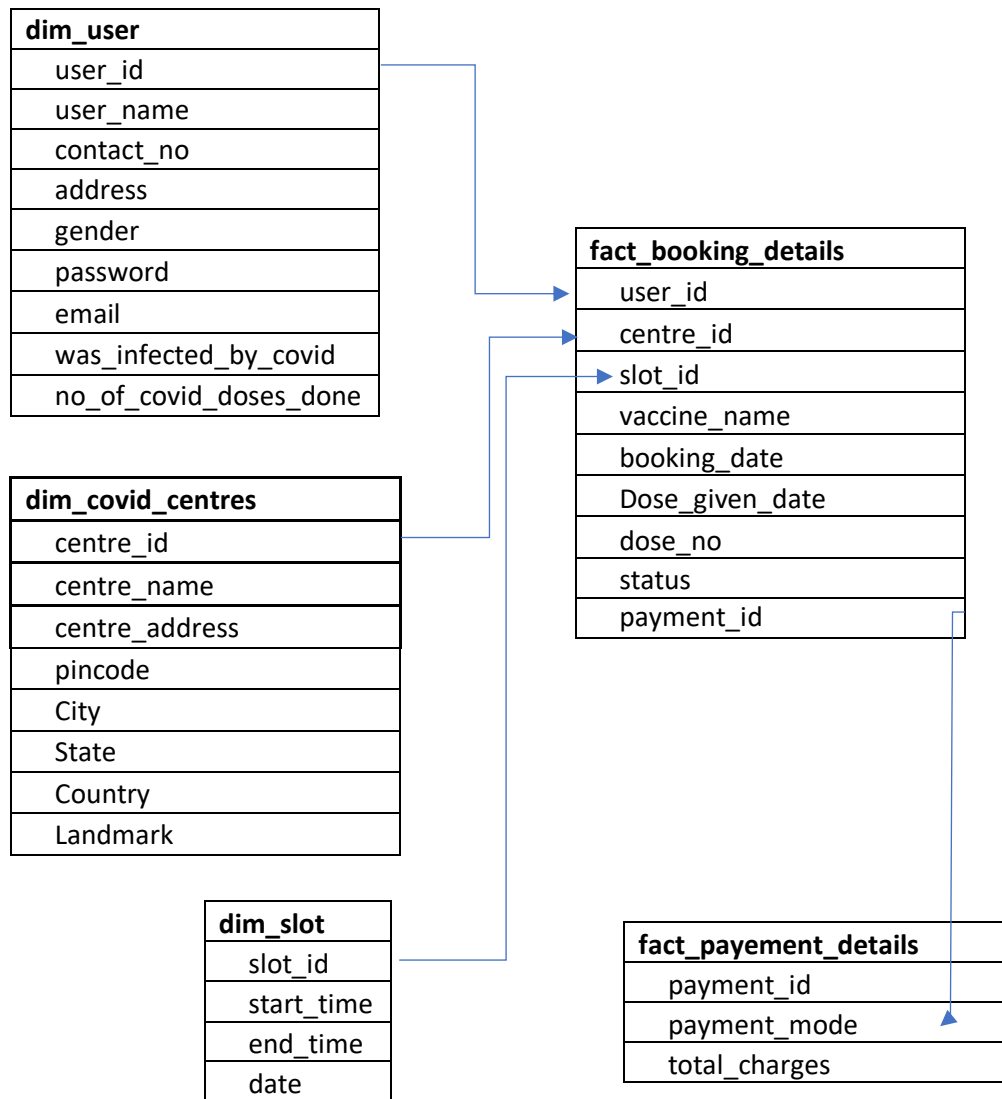
restaurant_id

) b on a. restaurant_id=b. restaurant_id

)

Where "Rank"<=5

Data Warehousing Design for Covid Vaccine Application



Explanation:

We have created below tables as a part of DWH design:

Table Name	Purpose
Dim_user:	Stores the details at user level.
Dim_Covid_Centres:	Dimension table which stores the details of the Covid Vaccine Centres
Dim_Slot:	Stores details of the various time slots available for taking vaccine.
Fact_booking_details	Fact table that stores the booking details of the vaccine. Also provides what is status[booked/cancelled/completed] of the user for given dose_no.

Data Warehousing Design for Covid Vaccine Application

fact_payement_details	Fact tables that captures the details regarding the payment made by the user against booking of the slot for vaccine
------------------------------	--

Questions:

Q1: Find out total vaccines dose administered/completed since inception till Dec-22 and ratio of how many people have taken 1st dose and 2nd dose.

Select

Dose_no,

Total_dose_given,

Sum(Total_dose_given) over () as "Overall_count",

Total_dose_given*1.0/ Overall_count*100 as Ratio

From

(

Select

Dose_no,

Count(1) as total_dose_given,

From fact_booking_details a

Where a.status='Completed' and Dose_given_date<='2022-12-31'

Group by dose_no

)

Q2: Find out total users who has been affected by covid before and have not completed a single dose.

Select

used_id,

user_name

From dim_user a

Where a. was_infected_by_covid='Y' and a. no_of_covid_doses_done=0