1. Repeat Customers problem

--Count of new and repeat customers by date

--Date, new\_cust\_count, repeat\_cust\_count

Table code:  
create table customer\_orders ( order\_id integer, customer\_id integer, order\_date date, order\_amount integer)

insert into customer\_orders values(1,100,cast('2022-01-01' as date),2000),(2,200,cast('2022-01-01' as date),2500),(3,300,cast('2022-01-01' as date),2100) ,(4,100,cast('2022-01-02' as date),2000),(5,400,cast('2022-01-02' as date),2200),(6,500,cast('2022-01-02' as date),2700) ,(7,100,cast('2022-01-03' as date),3000),(8,400,cast('2022-01-03' as date),1000),(9,600,cast('2022-01-03' as date),3000) ;

1. Uber Trip cancellation rate

--Cancellation rate for unbanned users by date

--No of rides cancelled for unbanned users/total no. of rides requested for unbanned users

--Date, cancelled\_trip count, total\_rides, cancellation\_rate

--cancellation rate = cancelled trip count/total\_rides \* 100

Table code:  
Create table Trips (id int, client\_id int, driver\_id int, city\_id int, status varchar(50), request\_at varchar(50)); Create table Users (users\_id int, banned varchar(50), role varchar(50));

insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('1', '1', '10', '1', 'completed', '2013-10-01'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('2', '2', '11', '1', 'cancelled\_by\_driver', '2013-10-01'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('3', '3', '12', '6', 'completed', '2013-10-01'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('4', '4', '13', '6', 'cancelled\_by\_client', '2013-10-01'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('5', '1', '10', '1', 'completed', '2013-10-02'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('6', '2', '11', '6', 'completed', '2013-10-02'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('7', '3', '12', '6', 'completed', '2013-10-02'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('8', '2', '12', '12', 'completed', '2013-10-03'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('9', '3', '10', '12', 'completed', '2013-10-03'); insert into Trips (id, client\_id, driver\_id, city\_id, status, request\_at) values ('10', '4', '13', '12', 'cancelled\_by\_driver', '2013-10-03'); insert into Users (users\_id, banned, role) values ('1', 'No', 'client'); insert into Users (users\_id, banned, role) values ('2', 'Yes', 'client'); insert into Users (users\_id, banned, role) values ('3', 'No', 'client'); insert into Users (users\_id, banned, role) values ('4', 'No', 'client'); insert into Users (users\_id, banned, role) values ('10', 'No', 'driver'); insert into Users (users\_id, banned, role) values ('11', 'No', 'driver'); insert into Users (users\_id, banned, role) values ('12', 'No', 'driver'); insert into Users (users\_id, banned, role) values ('13', 'No', 'driver');

1. Repeat Customers problem :
2. Ans:

WITH customer\_summary AS (

SELECT

customer\_id,

order\_date,

CASE

WHEN COUNT(\*) OVER (PARTITION BY customer\_id ORDER BY order\_date) = 1 THEN 'new'

ELSE 'repeat'

END AS customer\_type

FROM

customer\_orders

)

SELECT

order\_date,

COUNT(CASE WHEN customer\_type = 'new' THEN 1 END) AS new\_cust\_count,

COUNT(CASE WHEN customer\_type = 'repeat' THEN 1 END) AS repeat\_cust\_count

FROM

customer\_summary

GROUP BY

order\_date

ORDER BY

order\_date;

1. 2.. Uber Trip cancellation rate

--Cancellation rate for unbanned users by date

--No of rides cancelled for unbanned users/total no. of rides requested for unbanned users

--Date, cancelled\_trip count, total\_rides, cancellation\_rate

--cancellation rate = cancelled trip count/total\_rides \* 100

ANS:

SELECT

request\_at AS "Date",

COUNT(CASE WHEN status LIKE '%cancelled%' AND banned = 'No' THEN id END) AS "cancelled\_trip\_count",

COUNT(CASE WHEN banned = 'No' THEN id END) AS "total\_rides",

COUNT(CASE WHEN status LIKE '%cancelled%' AND banned = 'No' THEN id END) / COUNT(CASE WHEN banned = 'No' THEN id END) \* 100 AS "cancellation\_rate"

FROM

Trips

JOIN Users ON Trips.client\_id = Users.users\_id

GROUP BY

request\_at;