Задание 4

4.1 CREATE TABLE Users (

    Users\_Id INT PRIMARY KEY,

    Banned VARCHAR(3),

    Role VARCHAR(10)

);

CREATE TABLE Trips (

    Id INT PRIMARY KEY,

    Client\_Id INT,

    Driver\_Id INT,

    City\_Id INT,

    Status VARCHAR(20),

    Request\_at DATE,

    FOREIGN KEY (Client\_Id) REFERENCES Users(Users\_Id),

    FOREIGN KEY (Driver\_Id) REFERENCES Users(Users\_Id)

);

INSERT INTO Users (Users\_Id, Banned, Role) VALUES

(1, 'No', 'client'),

(2, 'Yes', 'client'),

(3, 'No', 'client'),

(4, 'No', 'client'),

(10, 'No', 'driver'),

(11, 'No', 'driver'),

(12, 'No', 'driver'),

(13, 'No', 'driver');

INSERT INTO Trips (Id, Client\_Id, Driver\_Id, City\_Id, Status, Request\_at) VALUES

(1, 1, 10, 1, 'completed', '2021-10-01'),

(2, 2, 11, 1, 'cancelled\_by\_driver', '2021-10-01'),

(3, 3, 12, 6, 'completed', '2021-10-01'),

(4, 4, 13, 6, 'cancelled\_by\_client', '2021-10-01'),

(5, 1, 10, 1, 'completed', '2021-10-02'),

(6, 2, 11, 6, 'completed', '2021-10-02'),

(7, 3, 12, 6, 'completed', '2021-10-02'),

(8, 2, 12, 12, 'completed', '2021-10-03'),

(9, 3, 10, 12, 'completed', '2021-10-03'),

(10, 4, 13, 12, 'cancelled\_by\_driver', '2021-10-03');

SELECT

    Request\_at AS Day,

    ROUND(

        SUM(CASE WHEN Status LIKE 'cancelled%' THEN 1 ELSE 0 END) \* 1.0 / COUNT(\*),

        2

    ) AS 'Cancellation Rate'

FROM Trips t

JOIN Users uc ON t.Client\_Id = uc.Users\_Id

JOIN Users ud ON t.Driver\_Id = ud.Users\_Id

WHERE uc.Banned = 'No'

    AND ud.Banned = 'No'

    AND Request\_at BETWEEN '2021-10-01' AND '2021-10-03'

GROUP BY Request\_at

ORDER BY Request\_at;

4.2

CREATE TABLE Employee2 (

    Id INT PRIMARY KEY,

    Name VARCHAR(50),

    Salary INT,

    DepartmentId INT

);

INSERT INTO Employee2 (Id, Name, Salary, DepartmentId) VALUES

(1, 'Joe', 70000, 1),

(2, 'Henry', 80000, 2),

(3, 'Sam', 60000, 2),

(4, 'Max', 90000, 1),

(5, 'Janet', 69000, 1),

(6, 'Randy', 85000, 1);

CREATE TABLE Department (

    Id INT PRIMARY KEY,

    Name VARCHAR(50)

);

INSERT INTO Department (Id, Name) VALUES

(1, 'IT'),

(2, 'Sales');

WITH RankedEmployees AS (

    SELECT

        d.Name AS Department,

        e.Name AS Employee,

        e.Salary,

        DENSE\_RANK() OVER (PARTITION BY e.DepartmentId ORDER BY e.Salary DESC) as salary\_rank

    FROM Employee2 e

    JOIN Department d ON e.DepartmentId = d.Id

)

SELECT

    Department,

    Employee,

    Salary

FROM RankedEmployees

WHERE salary\_rank <= 3

ORDER BY Department, Salary DESC;