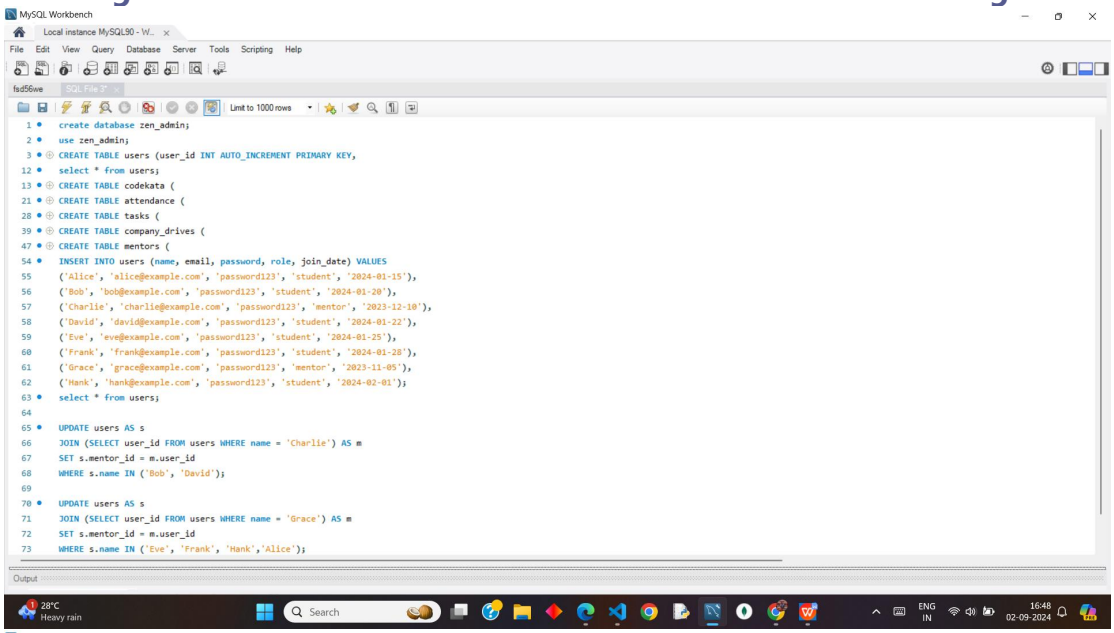
Design DB model for Guvi Zen class

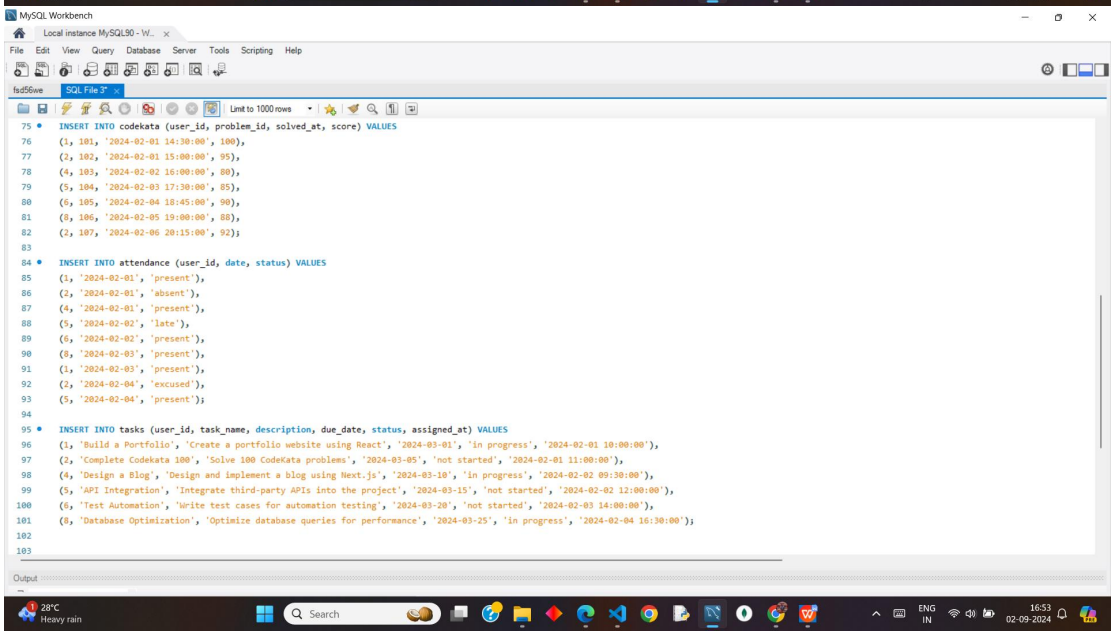
Relationships:

- **Users to Codekata: One-to-many relationship, as each user (student) can solve many CodeKata problems.**
- **Users to Attendance: One-to-many relationship, as each user has multiple attendance records.**
- **Users to Tasks: One-to-many relationship, as each user can be assigned multiple tasks.**
- **Users to Mentors: Self-referencing one-to-many relationship, as a mentor can have multiple mentees, and a student can have one mentor.**
- **Company Drives: This table is independent but can be linked to users if you want to track which students participated.**

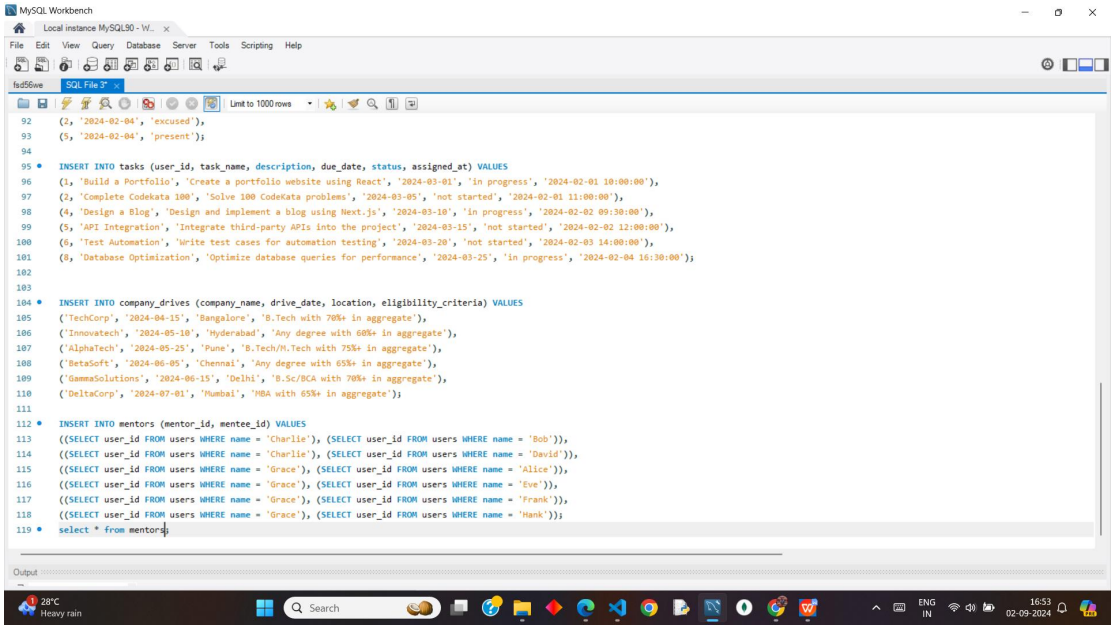
Creating database and tables for different datas and then inserting datas



```
create database zen_admin;
use zen_admin;
CREATE TABLE users (user_id INT AUTO_INCREMENT PRIMARY KEY,
select * from users;
CREATE TABLE codekata (
CREATE TABLE attendance (
CREATE TABLE tasks (
CREATE TABLE company_drives (
CREATE TABLE mentors (
INSERT INTO users (name, email, password, role, join_date) VALUES
('Alice', 'alice@example.com', 'password123', 'student', '2024-01-15'),
('Bob', 'bob@example.com', 'password123', 'student', '2024-01-20'),
('Charlie', 'charlie@example.com', 'password123', 'mentor', '2023-12-10'),
('David', 'david@example.com', 'password123', 'student', '2024-01-22'),
('Eve', 'eve@example.com', 'password123', 'student', '2024-01-25'),
('Frank', 'frank@example.com', 'password123', 'student', '2024-01-28'),
('Grace', 'grace@example.com', 'password123', 'mentor', '2023-11-05'),
('Hank', 'hank@example.com', 'password123', 'student', '2024-02-01');
select * from users;

UPDATE users AS s
JOIN (SELECT user_id FROM users WHERE name = 'Charlie') AS m
SET s.mentor_id = m.user_id
WHERE s.name IN ('Bob', 'David');

UPDATE users AS s
JOIN (SELECT user_id FROM users WHERE name = 'Grace') AS m
SET s.mentor_id = m.user_id
WHERE s.name IN ('Eve', 'Frank', 'Hank','Alice');
```



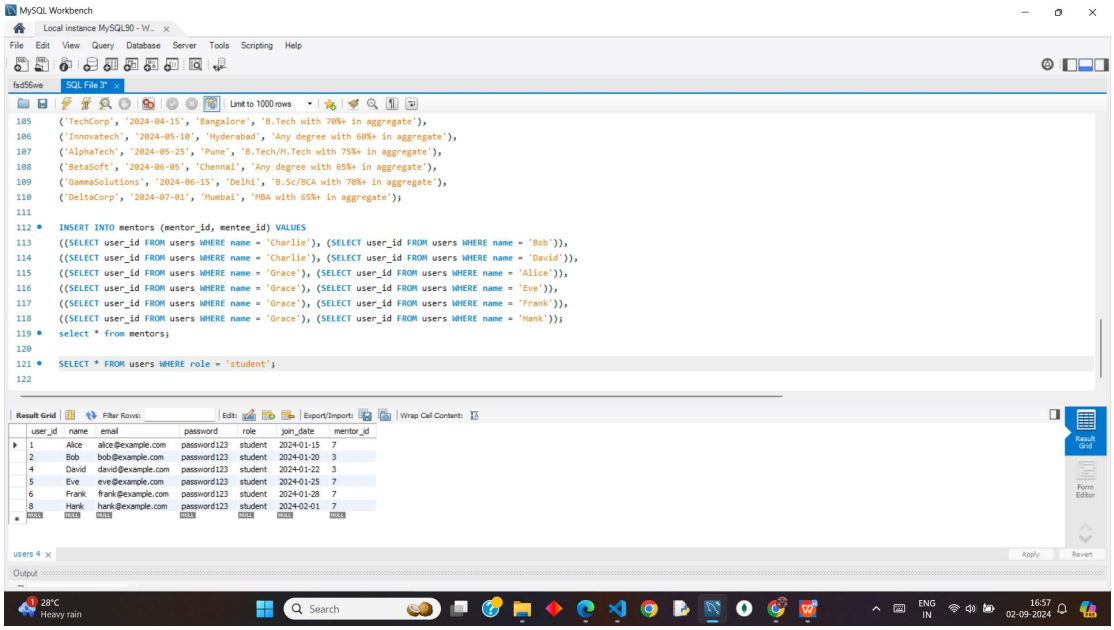
```
INSERT INTO codekata (user_id, problem_id, solved_at, score) VALUES
(1, 101, '2024-02-01 14:30:00', 100),
(2, 102, '2024-02-01 15:00:00', 95),
(4, 103, '2024-02-02 16:00:00', 80),
(5, 104, '2024-02-03 17:30:00', 85),
(6, 105, '2024-02-04 18:45:00', 90),
(8, 106, '2024-02-05 19:00:00', 88),
(2, 107, '2024-02-06 20:15:00', 92);

INSERT INTO attendance (user_id, date, status) VALUES
(1, '2024-02-01', 'present'),
(2, '2024-02-01', 'absent'),
(4, '2024-02-01', 'present'),
(5, '2024-02-02', 'late'),
(6, '2024-02-02', 'present'),
(8, '2024-02-03', 'present'),
(1, '2024-02-03', 'present'),
(2, '2024-02-04', 'excused'),
(5, '2024-02-04', 'present');

INSERT INTO tasks (user_id, task_name, description, due_date, status, assigned_at) VALUES
(1, 'Build a Portfolio', 'Create a portfolio website using React', '2024-03-01', 'in progress', '2024-02-01 10:00:00'),
(2, 'Complete Codekata 100', 'Solve 100 CodeKata problems', '2024-03-05', 'not started', '2024-02-01 11:00:00'),
(4, 'Design a Blog', 'Design and implement a blog using Next.js', '2024-03-10', 'in progress', '2024-02-02 09:30:00'),
(5, 'API Integration', 'Integrate third-party APIs into the project', '2024-03-15', 'not started', '2024-02-02 12:00:00'),
(6, 'Test Automation', 'Write test cases for automation testing', '2024-03-20', 'not started', '2024-02-03 14:00:00'),
(8, 'Database Optimization', 'Optimize database queries for performance', '2024-03-25', 'in progress', '2024-02-04 16:30:00');
```



```
(2, '2024-02-04', 'excused'),
(5, '2024-02-04', 'present');

INSERT INTO tasks (user_id, task_name, description, due_date, status, assigned_at) VALUES
(1, 'Build a Portfolio', 'Create a portfolio website using React', '2024-03-01', 'in progress', '2024-02-01 10:00:00'),
(2, 'Complete Codekata 100', 'Solve 100 CodeKata problems', '2024-03-05', 'not started', '2024-02-01 11:00:00'),
(4, 'Design a Blog', 'Design and implement a blog using Next.js', '2024-03-10', 'in progress', '2024-02-02 09:30:00'),
(5, 'API Integration', 'Integrate third-party APIs into the project', '2024-03-15', 'not started', '2024-02-02 12:00:00'),
(6, 'Test Automation', 'Write test cases for automation testing', '2024-03-20', 'not started', '2024-02-03 14:00:00'),
(8, 'Database Optimization', 'Optimize database queries for performance', '2024-03-25', 'in progress', '2024-02-04 16:30:00');


INSERT INTO company_drives (company_name, drive_date, location, eligibility_criteria) VALUES
('TechCorp', '2024-04-15', 'Bangalore', 'B.Tech with 70%+ in aggregate'),
('Innovatech', '2024-05-10', 'Hyderabad', 'Any degree with 60%+ in aggregate'),
('AlphaTech', '2024-05-25', 'Pune', 'B.Tech/M.Tech with 75%+ in aggregate'),
('BetaSoft', '2024-06-05', 'Chennai', 'Any degree with 65%+ in aggregate'),
('GammaSolutions', '2024-06-15', 'Delhi', 'B.Sc/BCA with 70%+ in aggregate'),
('DeltaCorp', '2024-07-01', 'Mumbai', 'MBA with 65%+ in aggregate');

INSERT INTO mentors (mentor_id, mentee_id) VALUES
((SELECT user_id FROM users WHERE name = 'Charlie'), (SELECT user_id FROM users WHERE name = 'Bob')),
((SELECT user_id FROM users WHERE name = 'Charlie'), (SELECT user_id FROM users WHERE name = 'David')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Alice')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Eve')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Frank')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Hank'));
select * from mentors;
```

Basic Queries

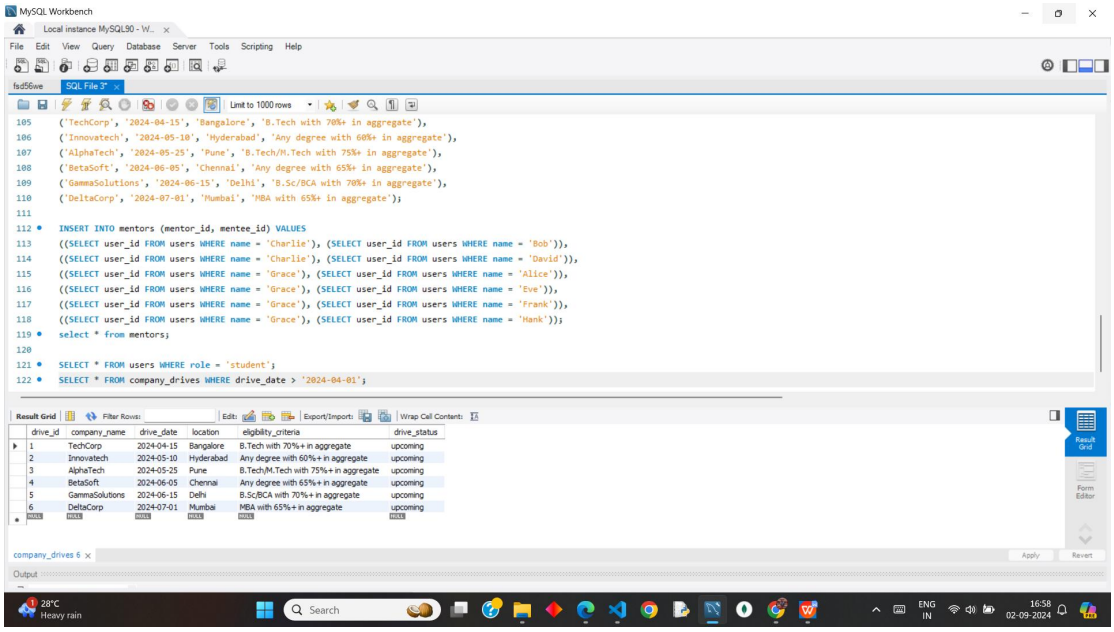
a. Retrieve all students' details:



```
('TechCorp', '2024-04-15', 'Bangalore', 'B.Tech with 70%+ in aggregate'),
('Innovatech', '2024-05-10', 'Hyderabad', 'Any degree with 60%+ in aggregate'),
('AlphaTech', '2024-05-25', 'Pune', 'B.Tech/M.Tech with 75%+ in aggregate'),
('BetaSoft', '2024-06-05', 'Chennai', 'Any degree with 65%+ in aggregate'),
('GammaSolutions', '2024-06-15', 'Delhi', 'B.Sc/BCA with 70%+ in aggregate'),
('DeltaCorp', '2024-07-01', 'Mumbai', 'MBA with 65%+ in aggregate');

INSERT INTO mentors (mentor_id, mentee_id) VALUES
((SELECT user_id FROM users WHERE name = 'Charlie'), (SELECT user_id FROM users WHERE name = 'Bob')),
((SELECT user_id FROM users WHERE name = 'Charlie'), (SELECT user_id FROM users WHERE name = 'David')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Alice')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Eve')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Frank')),
((SELECT user_id FROM users WHERE name = 'Grace'), (SELECT user_id FROM users WHERE name = 'Hank'));
select * from mentors;

SELECT * FROM users WHERE role = 'student';
```

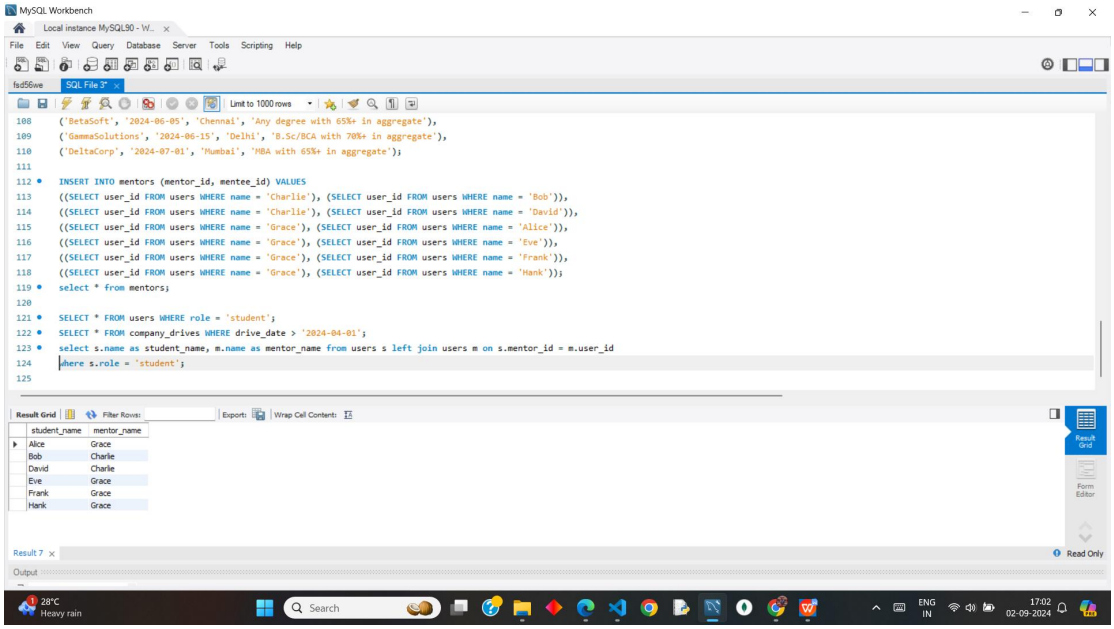
user_id	name	email	password	role	join_date	mentor_id
1	Alice	alice@example.com	password123	student	2024-01-15	7
2	Bob	bob@example.com	password123	student	2024-01-20	3
4	David	david@example.com	password123	student	2024-01-22	3
5	Eve	eve@example.com	password123	student	2024-01-25	7
6	Frank	frank@example.com	password123	student	2024-01-28	7
8	Hank	hank@example.com	password123	student	2024-02-01	7

b. List all company drives happening after a certain date:

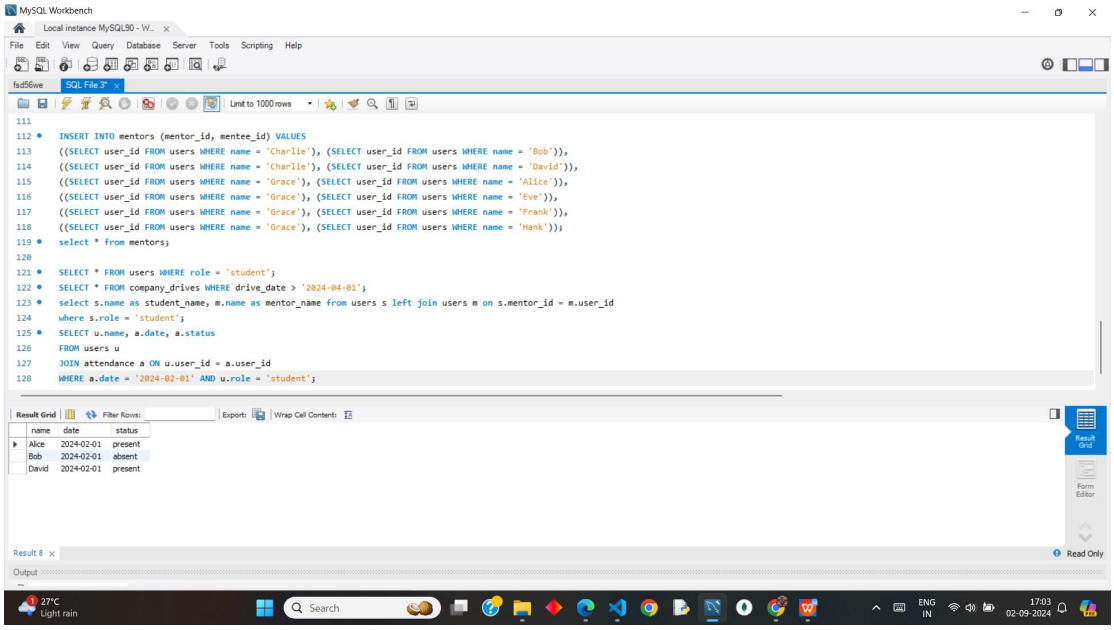


Joins

a. Retrieve all students along with their mentor's name:

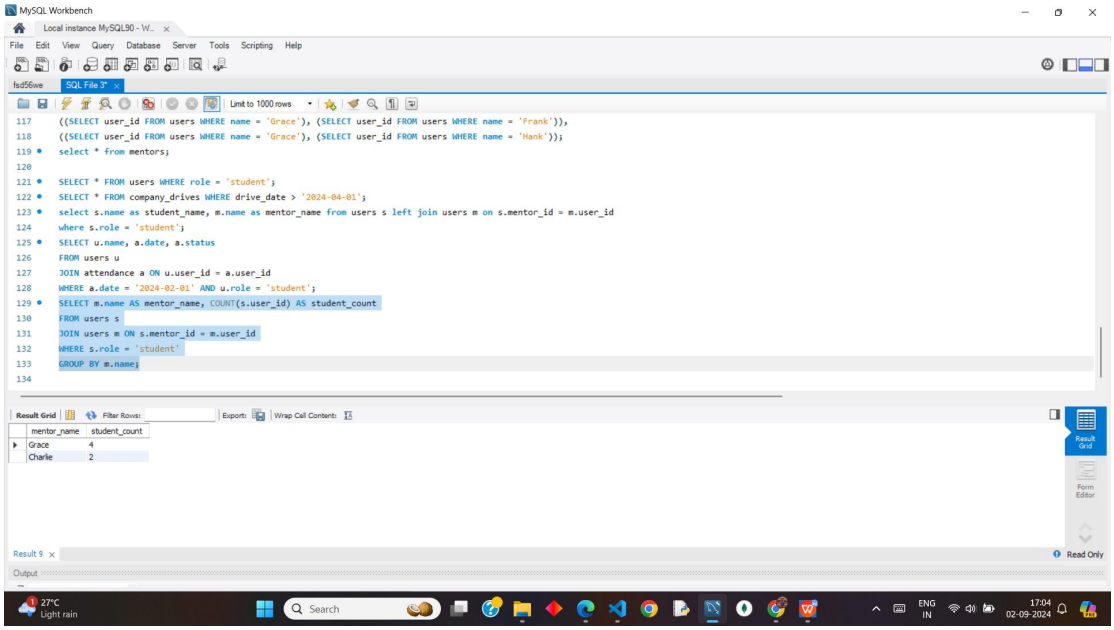


b. List students with their attendance status for a specific date:

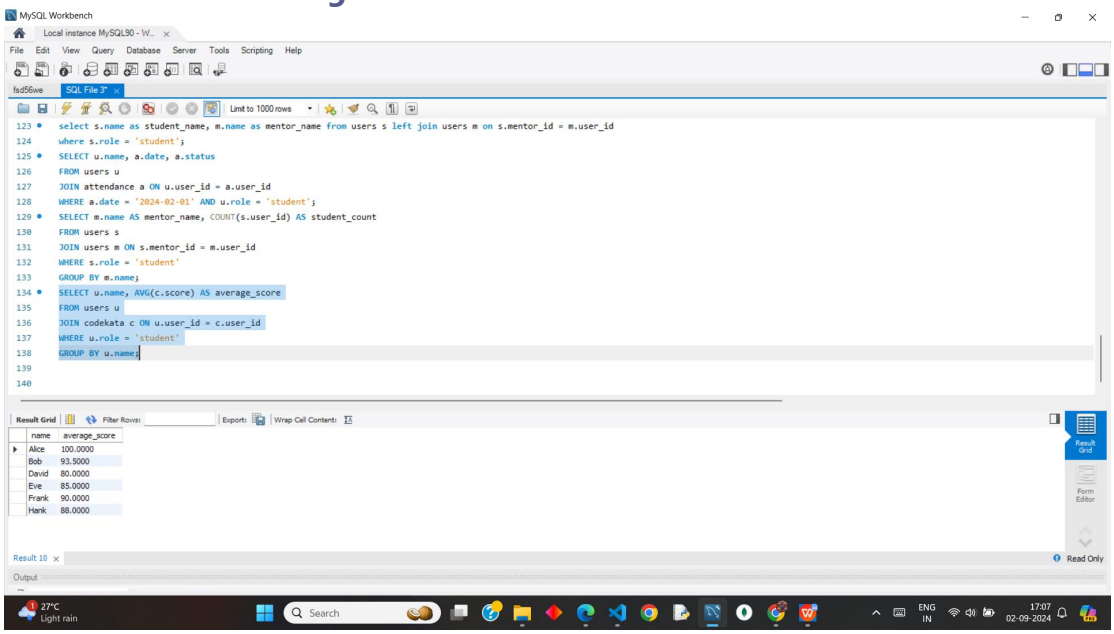


Aggregate Functions

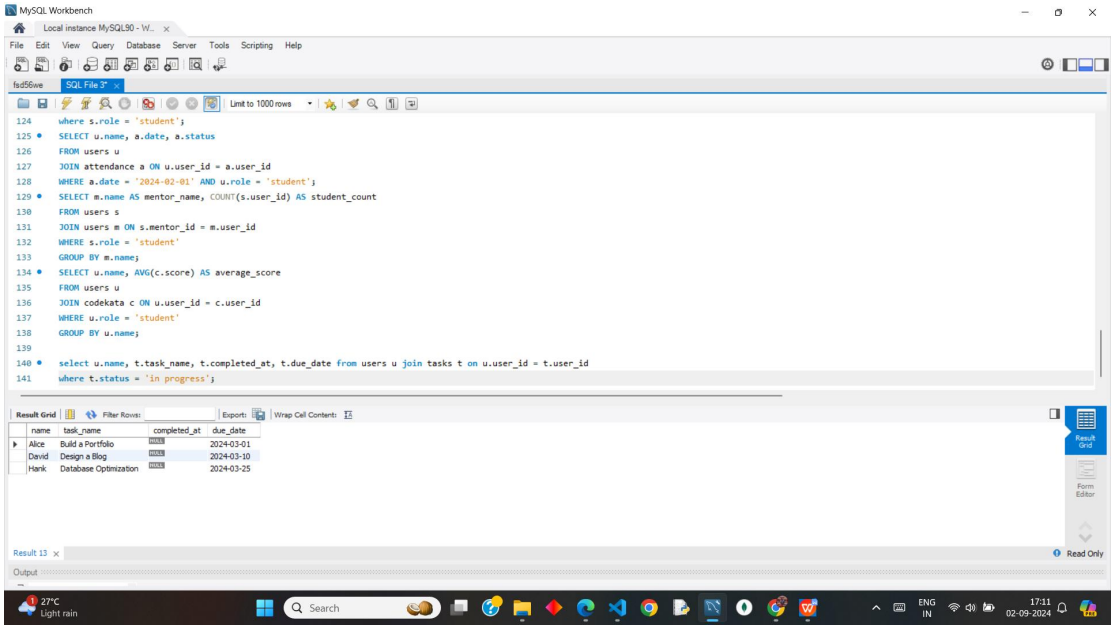
a. Count the number of students assigned to each mentor:



b. Calculate the average Codekata score for each student:

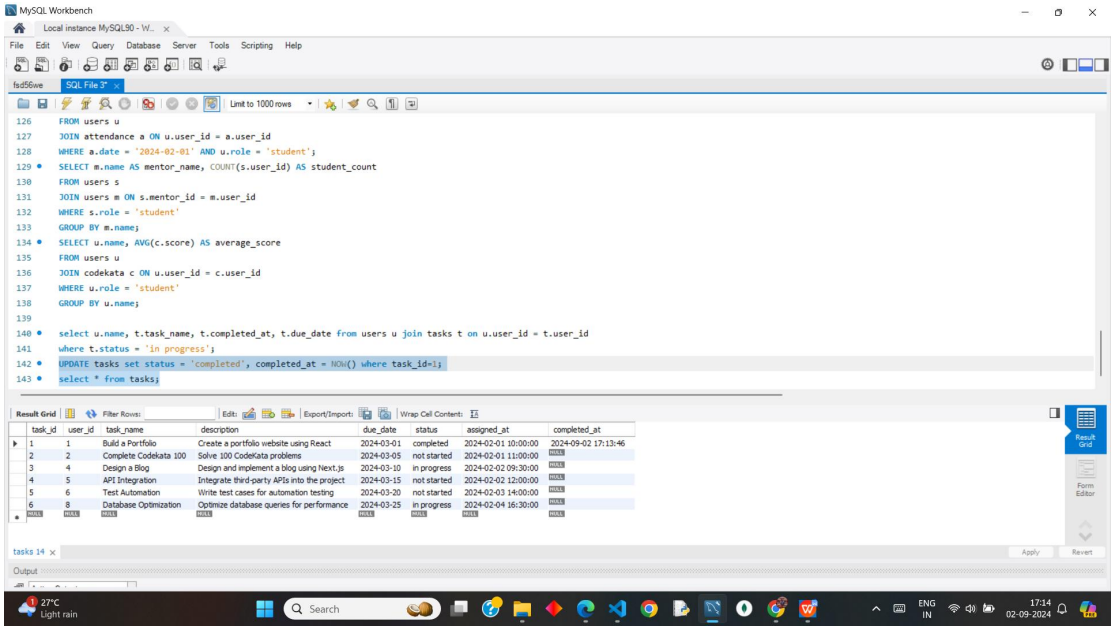


Get the details of students who have status of ‘in progress’ in their tasks:



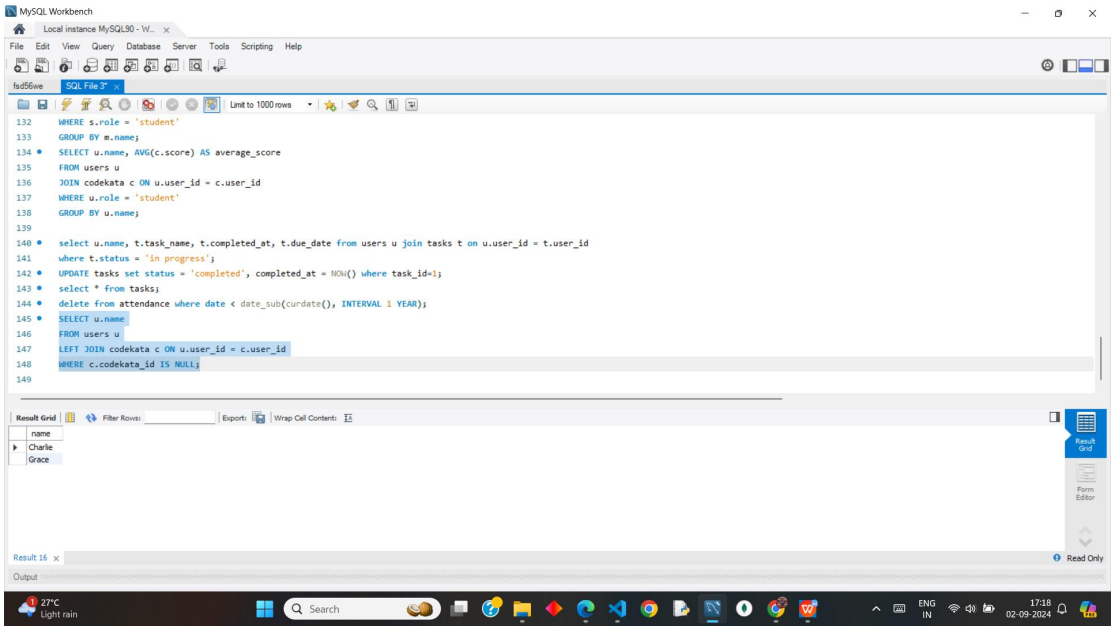
Updating and Deleting Data

a. Update the status of a task when it's completed:



Advanced Joins

a. Find data who haven't solved any Codekata problems:



b. Retrieve all students who attended a specific company drive: