### Phase 5 Relational Algebra

#### 1. List all students registered for training along with their mentor

π Student.Student\_ID, User.Name AS Student\_Name, Uni\_Mentor.Uni\_Name AS Uni\_Mentor, Company\_Mentor.Company\_Name AS Company\_Mentor

(Student ⋈ (User ⋈ (Internship\_Application ⋈ (Uni\_Mentor ⋈ Mentor) ⋈ (Company\_Mentor ⋈ Mentor))))

## 2. Which students are assigned to a particular company mentor (given a specific mentor ID)?

π Student.Student\_ID, User.Name, Student.Major, Student.CGPA

(σ Company\_Mentor.Company\_Mentor\_ID='specific\_id'(Company\_Mentor) ⋈ Internship\_Application ⋈ Student ⋈ User)

#### 3. How many students are assigned to each company mentor?

γ Company\_Mentor.Company\_Mentor\_ID, User.Name AS Mentor\_Name; COUNT(Student.Student\_ID)→Assigned\_Students

(Company\_Mentor ⋈ Mentor ⋈ User ⋈ Internship\_Application ⋈ Student)

#### 4. List all trainings and the number of enrolled students

γ Company.Name, Company.Industry; COUNT(DISTINCT Student.Student ID)→Enrolled Students

(Company ⋈ Has\_a\_Relation ⋈ Internship\_Application ⋈ Student)

# 5. Which students have registered for training but do not have a supervisor assigned yet?

π Student.Student\_ID, User.Name, Student.Major

( $\sigma$  Internship\_Application.Uni\_Mentor\_ID=NULL(Internship\_Application)  $\bowtie$  Student  $\bowtie$  User)

#### 6. List names and emails of all users who live in Riyadh

π Name, Email (σ Address LIKE '%Riyadh%' (User))

#### 7. How many students with GPA higher than 3?

γ COUNT(Student\_ID)→HighGPAStudents
(σ GPA > 3(Student))

#### 8. How many students were approved?

γ COUNT(Student\_ID)→ApprovedStudents
(σ Application\_State='Approved'(Student))

#### 9. Show students with specific majors

π Student.Student\_ID, User.Name, Student.CGPA, Student.Academic\_Level (σ Major='Computer Science'(Student ⋈ User))

#### 10. Show students according to the nearest addresses

τ Address(π Student\_ID, User.Name, User.Address(Student ⋈ User))

#### 11. Find Students with Highest Performance Score

π Student.Student\_ID, User.Name, Evaluation\_Report.Performance\_Score
 (Student ⋈ User ⋈ Internship\_Application ⋈
 σ Performance\_Score = MAX(Performance\_Score)(Evaluation\_Report))

#### 12. How many companies accepted students?

γ COUNT(DISTINCT Company.Company\_Logo)→Companies\_Accepting\_Students

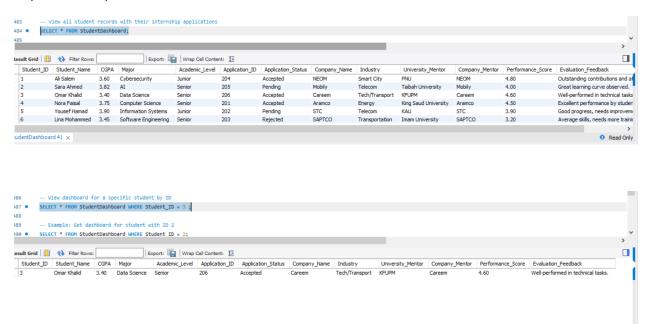
(Company ⋈ Has\_a\_Relation ⋈ σ Status='Accepted'(Internship\_Application))

#### 1. View: StudentDashboard

Purpose: To provide a comprehensive overview of each student's academic details, internship application status, and evaluations in a single view.

Functionality: This view joins several tables related to the student, including user information, internship applications, mentors, company info, and evaluation reports.

#### **Expected output:**

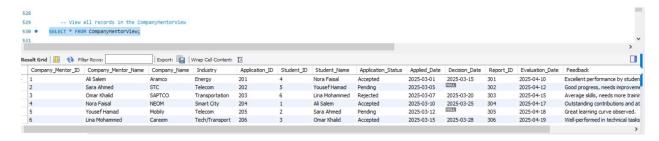


#### 2. View: CompanyMentorView

Purpose: To give company mentors a dashboard summarizing their assigned students, applications, and evaluations.

Functionality: Joins data from mentors, users, students, applications, and evaluation reports. Designed to help mentors monitor student progress and performance.

#### Expected output:

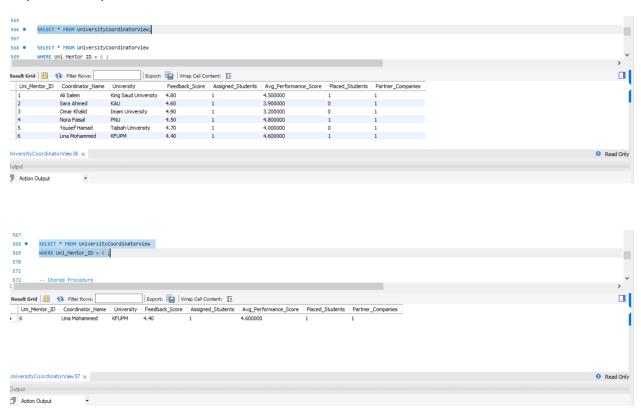


#### 3. View: UniversityCoordinatorView

Purpose: To summarize the performance of university mentors, showing how many students are under their supervision, how many are placed, and average performance.

Functionality: Aggregates data using GROUP BY to show KPIs (Key Performance Indicators) per university coordinator.

#### Expected output:

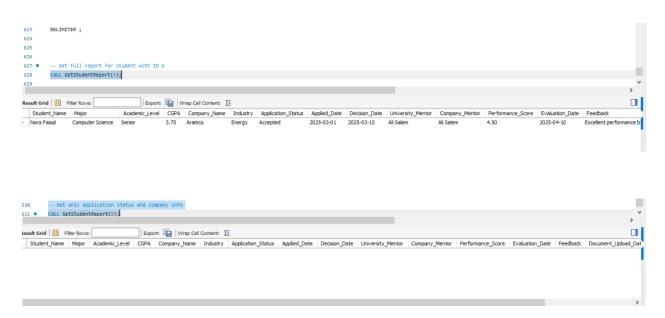


### 4. Stored Procedure: GetStudentReport(student\_id)

Purpose:To generate a detailed report for a specific student, including personal info, academic records, internship applications, mentor names, evaluations, and documents.

Functionality: -Accepts student\_id as input.

- -Pulls together detailed records using joins.
- -Allows filtering on the front-end if only partial data is needed.



(Because we have no id=8)