Database

Tutorial 1

1: Table Creation and Constraints:

Create the following five tables:

1. students:
    - student_id INTEGER PRIMARY KEY
    - name VARCHAR(50) NOT NULL
    - email VARCHAR(100) UNIQUE NOT NULL
    - dob DATE NOT NULL
    - gender CHAR(1) NOT NULL CHECK (gender IN ('M','F','O'))
    - dept_name VARCHAR(30) NOT NULL

2. Teacher:
    - teacher_id INTEGER PRIMARY KEY
    - name VARCHAR(50) NOT NULL
    - email VARCHAR(100) UNIQUE NOT NULL
    - dept_name VARCHAR(30) NOT NULL
    - salary NUMERIC(8,2) NOT NULL CHECK (salary > 0)

3. subject:
    - subject_code VARCHAR(10) PRIMARY KEY
    - title VARCHAR(100) NOT NULL
    - credits INTEGER NOT NULL CHECK (credits BETWEEN 1 AND 6)
    - dept_name VARCHAR(30) NOT NULL
    - teacher_id INTEGER NOT NULL REFERENCES teacher(teacher_id)

4. course:
    - course_id INTEGER PRIMARY KEY
    - subject_code VARCHAR(10) NOT NULL REFERENCES subject(subject_code)
    - semester VARCHAR(10) NOT NULL
    - year INTEGER NOT NULL
    - section CHAR(1) NOT NULL
    - UNIQUE(subject_code, semester, year, section)

5. registration:
    - reg_id INTEGER PRIMARY KEY
    - student_id INTEGER NOT NULL REFERENCES students(student_id)

- course_id INTEGER NOT NULL REFERENCES course(course_id)
- reg_date DATE NOT NULL
- grade CHAR(2) CHECK (grade IN ('A','B','C','D','E','F','NA'))
- UNIQUE(student_id, course_id)

2: DDL Practice Queries:

1. Add a new column mobile (VARCHAR(15)) to the students table.
2. Drop the salary column from the teacher table and recreate it.
3. Modify the email column in students to increase its length to 120.
4. Add a unique constraint on (name, dob) in students.
5. Drop the unique constraint added above and recreate it.
6. Add a foreign key constraint on teacher.dept_name referencing students.dept_name.
7. Drop the registration table and recreate it.
8. Add a check constraint to registration.reg_date to disallow future dates.
9. Rename course.section to section_id and recreate it.
10. Restrict subject.credits to be between 2 and 5 and recreate it.

3: Insert the Following Data into the Table:

students -

(1001, 'Alice Johnson', 'alice.johnson@email.com', '2000-05-15', 'F', 'Computer Sci'),
(1002, 'Bob Smith', 'bob.smith@email.com', '1999-08-20', 'M', 'Mathematics'),
(1003, 'Charlie Lee', 'charlie.lee@email.com', '2001-11-02', 'M', 'Physics'),
(1004, 'Daisy Clark', 'daisy.clark@email.com', '2000-03-30', 'F', 'Biology'),
(1005, 'Ethan Brown', 'ethan.brown@email.com', '2002-01-17', 'M', 'History'),
(1006, 'Fiona Davis', 'fiona.davis@email.com', '1998-09-12', 'F', 'Computer Sci'),
(1007, 'George Wilson', 'george.wilson@email.com', '1999-12-24', 'M', 'Mathematics'),
(1008, 'Hannah Moore', 'hannah.moore@email.com', '2001-07-06', 'F', 'Physics'),
(1009, 'Irene Scott', 'irene.scott@email.com', '2000-10-19', 'F', 'Biology'),
(1010, 'Jack Miller', 'jack.miller@email.com', '2001-06-21', 'M', 'Computer Sci')

teacher -

(2001, 'Dr. Alan Turing', 'alan.turing@email.com', 'Computer Sci', 90000.00),
(2002, 'Prof. Jane Doe', 'jane.doe@email.com', 'Mathematics', 85000.00),
(2003, 'Dr. Marie Curie', 'marie.curie@email.com', 'Physics', 87000.00),
(2004, 'Prof. Isaac Newton', 'isaac.newton@email.com', 'Physics', 91000.00),
(2005, 'Dr. Rosalind Franklin', 'rosalind.franklin@email.com', 'Biology', 88000.00),
(2006, 'Prof. John Nash', 'john.nash@email.com', 'Mathematics', 86000.00),
(2007, 'Dr. Ada Lovelace', 'ada.lovelace@email.com', 'Computer Sci', 92000.00),
(2008, 'Prof. Richard Feynman', 'richard.feynman@email.com', 'Physics', 89000.00),

(2009, 'Dr. Jane Goodall', 'jane.goodall@email.com', 'Biology', 87000.00),
(2010, 'Prof. Stephen Jay', 'stephen.jay@email.com', 'History', 73000.00)

subject -
('CS101', 'Intro to CS', 3, 'Computer Sci', 2001),
('MATH201', 'Calculus II', 4, 'Mathematics', 2002),
('PHY101', 'Physics I', 4, 'Physics', 2003),
('BIO110', 'General Biology', 3, 'Biology', 2005),
('HIST201', 'Ancient Civilizations', 4, 'History', 2010),
('CS102', 'Data Structures', 4, 'Computer Sci', 2007),
('MATH101', 'Algebra', 3, 'Mathematics', 2006),
('PHY102', 'Physics II', 3, 'Physics', 2004),
('BIO210', 'Molecular Biology', 4, 'Biology', 2009),
('CS201', 'Operating Systems', 3, 'Computer Sci', 2001)

course -
(3001, 'CS101', 'Spring', 2023, 'A'),
(3002, 'MATH201', 'Spring', 2023, 'A'),
(3003, 'PHY101', 'Spring', 2023, 'B'),
(3004, 'BIO110', 'Spring', 2023, 'A'),
(3005, 'HIST201', 'Fall', 2023, 'A'),
(3006, 'CS102', 'Fall', 2023, 'A'),
(3007, 'MATH101', 'Fall', 2023, 'B'),
(3008, 'PHY102', 'Fall', 2023, 'A'),
(3009, 'BIO210', 'Fall', 2023, 'B'),
(3010, 'CS201', 'Fall', 2023, 'B')

registration -
(4001, 1001, 3001, '2023-01-10', 'A'),
(4002, 1002, 3002, '2023-01-11', 'B'),
(4003, 1003, 3003, '2023-01-12', 'C'),
(4004, 1004, 3004, '2023-01-13', 'A'),
(4005, 1005, 3005, '2023-08-20', 'B'),
(4006, 1006, 3006, '2023-08-21', 'A'),
(4007, 1007, 3007, '2023-08-22', 'C'),
(4008, 1008, 3008, '2023-08-23', 'D'),
(4009, 1009, 3009, '2023-08-24', 'B'),
(4010, 1010, 3010, '2023-08-25', 'A')

4: Data Manipulation Queries:

1. Write a query to delete all registrations with grade 'F'.
2. Write a query to update a student's email using their student_id. Ensure email remains unique.
3. Write a query to list all student names with their department names.
4. Write a query to list all teachers who belong to the 'Computer Sci' department.
5. Write a query to find subjects that have more than 3 credits.
6. Write a query to display all courses offered in 'Fall' 2023.
7. Write a query to find students registered for courses in 'Spring' 2023.
8. Write a query to get all teachers whose salary is above the average salary.

5: Join and Advanced Queries:

1. Write a query to display students and the subjects they are registered for.
2. Write a query to show course title, teacher name, and number of registrations per course.
3. Write a query to list all teachers and their departments, including those not teaching any subjects.
4. Write a query to list students registered in courses taught by Physics teachers.
5. Write a query to list all courses with credits >= 4, including course title and teacher.
6. Write a query to list all teachers who are not teaching any subject.

6: Aggregates and Grouping:

1. Write a query to count the number of registrations per semester in 2023.
2. Write a query to find the average salary of teachers grouped by department.
3. Write a query to find the maximum credits assigned per department.
4. Write a query to get student count per course.
5. Write a query to list departments with average salary greater than ₹80,000.
6. Write a query to list courses from departments where the total credits exceed 15.
7. Write a query to count how many courses each teacher is teaching.

7: Updates, Deletes, and Advanced Constraints:

1. Write a query to update teacher salary by 7% if it’s less than ₹75,000, otherwise by 3%.
2. Write a query to delete registrations older than 2 years from today.
3. Write a query to set NULL grades to 'NA'.
4. Write a query to increase all Mathematics subjects’ credits by 1.
5. Write a query to set dept_name to 'Undeclared' where it is NULL in students table.
6. Write a query to delete courses that have no registrations.