TABLE CREATION

Students Table

Create a table `students` with:

- `student_id` (integer, primary key, auto-increment)
- `full_name` (text, NOT NULL)
- `email` (text, UNIQUE and NOT NULL)
- `date_of_birth` (date, NOT NULL)
- `gender` (text, CHECK gender IN ('Male','Female','Other'))
- `created_at` (timestamp, DEFAULT current timestamp)

Departments Table

Create a table `departments` with:

- `dept_id` (integer, primary key)
- `dept_name` (varchar, UNIQUE and NOT NULL)
- `building` (varchar)
- `budget` (integer CHECK budget > 0)

Courses Table

Create a `courses` table with:

- `course_id` (integer, primary key)
- `course_code` (varchar, UNIQUE and NOT NULL)

- `course_title` (varchar, NOT NULL)
- `credit` (integer CHECK credit BETWEEN 1 AND 4)
- `dept_id` (integer, NOT NULL, FK → departments.dept_id)

Teachers Table

Create a `teachers` table with:

- `teacher_id` (integer, primary key)
- `teacher_name` (varchar NOT NULL)
- `email` (varchar UNIQUE NOT NULL)
- `salary` (integer CHECK salary >= 10000)
- `dept_id` (FK → departments.dept_id)

Course_Registrations Table

Students register for courses.

Create a `course_registrations` table with:

- `reg_id` (primary key)
- `student_id` (FK → students.student_id)
- `course_id` (FK → courses.course_id)
- `semester` (varchar NOT NULL CHECK semester IN ('Spring','Summer','Fall'))
- `year` (integer CHECK year >= 2000)
- UNIQUE(student_id, course_id, semester, year) → No duplicate registration

---

TABLE CREATION COMMANDS:

```
-- ============================
-- 1. STUDENTS TABLE
-- ============================
CREATE TABLE students (
    student_id      SERIAL PRIMARY KEY,
    full_name       VARCHAR(100) NOT NULL,
    email           VARCHAR(100) NOT NULL UNIQUE,
    date_of_birth   DATE NOT NULL,
    gender          VARCHAR(10) CHECK (gender IN ('Male', 'Female', 'Other')),
    created_at      TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);


-- ============================
-- 2. DEPARTMENTS TABLE
-- ============================
CREATE TABLE departments (
    dept_id     INT PRIMARY KEY,
    dept_name   VARCHAR(100) NOT NULL UNIQUE,
    building    VARCHAR(50),
    budget      INT CHECK (budget > 0)
);
```

```
-- ============================
-- 3. COURSES TABLE
-- ============================
CREATE TABLE courses (
    course_id       INT PRIMARY KEY,
    course_code     VARCHAR(20) NOT NULL UNIQUE,
    course_title    VARCHAR(150) NOT NULL,
    credit          INT CHECK (credit BETWEEN 1 AND 4),
    dept_id         INT NOT NULL,
    CONSTRAINT fk_dept_id FOREIGN KEY (dept_id) REFERENCES departments(dept_id)
);

-- ============================
-- 4. TEACHERS TABLE
-- ============================
CREATE TABLE teachers (
    teacher_id      INT PRIMARY KEY,
    teacher_name    VARCHAR(100) NOT NULL,
    email           VARCHAR(100) NOT NULL UNIQUE,
    salary          INT CHECK (salary >= 10000),
    dept_id         INT,
    CONSTRAINT fk_dept_id FOREIGN KEY (dept_id) REFERENCES departments(dept_id)

);
```

```
-- ============================
-- 5. COURSE REGISTRATIONS
-- ============================
CREATE TABLE course_registrations (
    reg_id      SERIAL PRIMARY KEY,
    student_id  INT NOT NULL,
    course_id   INT NOT NULL,
    semester    VARCHAR(10) NOT NULL CHECK (semester IN ('Spring', 'Summer', 'Fall')),
    year        INT CHECK (year >= 2000),
    CONSTRAINT uniq_student_id UNIQUE (student_id, course_id, semester, year),
    CONSTRAINT fk_student_id FOREIGN KEY (student_id) REFERENCES students(student_id),
    CONSTRAINT fk_course_id FOREIGN KEY (course_id) REFERENCES courses(course_id)

);
```