-- Drop tables

DROP TABLE Coursera\_Certificates CASCADE CONSTRAINTS;

DROP TABLE Coursera\_Enrollments CASCADE CONSTRAINTS;

DROP TABLE Coursera\_Courses CASCADE CONSTRAINTS;

DROP TABLE Coursera\_Users CASCADE CONSTRAINTS;

DROP TABLE COURSERA\_INSTRUCTORS CASCADE CONSTRAINTS;

DROP PROCEDURE Get\_Enrollment\_Status;

DROP PROCEDURE Get\_Student\_Courses;

-- Drop sequences

DROP SEQUENCE seq\_users;

DROP SEQUENCE seq\_courses;

DROP SEQUENCE seq\_enrollments;

DROP SEQUENCE seq\_certificates;

CREATE TABLE Coursera\_Users (

user\_id NUMBER PRIMARY KEY,

full\_name VARCHAR2(100) NOT NULL,

email VARCHAR2(100) UNIQUE NOT NULL CHECK (email LIKE '%@%'),

pass VARCHAR2(255) NOT NULL,

created\_at DATE DEFAULT SYSDATE

);

CREATE TABLE COURSERA\_INSTRUCTORS (

instructor\_id NUMBER PRIMARY KEY,

full\_name VARCHAR2(100) NOT NULL,

email VARCHAR2(100) UNIQUE NOT NULL CHECK (email LIKE '%@%'),

hired\_date DATE DEFAULT SYSDATE,

bio VARCHAR2(2000)

);

CREATE TABLE Coursera\_Courses (

course\_id NUMBER PRIMARY KEY,

title VARCHAR2(100) NOT NULL,

description VARCHAR2(4000),

price NUMBER(10,2) DEFAULT 0,

duration\_weeks NUMBER(2),

instructor\_id NUMBER,

CONSTRAINT fk\_course\_instructor FOREIGN KEY (instructor\_id)

REFERENCES COURSERA\_INSTRUCTORS (instructor\_id)

ON DELETE SET NULL

);

CREATE TABLE Coursera\_Enrollments (

enrollment\_id NUMBER PRIMARY KEY,

user\_id NUMBER NOT NULL,

course\_id NUMBER NOT NULL,

enrollment\_date DATE DEFAULT SYSDATE,

completion\_status VARCHAR2(15) DEFAULT 'not\_started',

CONSTRAINT fk\_enrollment\_user FOREIGN KEY (user\_id) REFERENCES Coursera\_Users(user\_id),

CONSTRAINT fk\_enrollment\_course FOREIGN KEY (course\_id) REFERENCES Coursera\_Courses(course\_id),

CONSTRAINT uk\_user\_course UNIQUE (user\_id, course\_id)

);

CREATE TABLE Coursera\_Certificates (

certificate\_id NUMBER PRIMARY KEY,

enrollment\_id NUMBER NOT NULL,

issue\_date DATE DEFAULT SYSDATE,

CONSTRAINT fk\_certificate\_enrollment FOREIGN KEY (enrollment\_id) REFERENCES Coursera\_Enrollments(enrollment\_id)

);

CREATE SEQUENCE seq\_users START WITH 1001 INCREMENT BY 1;

CREATE SEQUENCE seq\_courses START WITH 2001 INCREMENT BY 1;

CREATE SEQUENCE seq\_enrollments START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE seq\_certificates START WITH 3001 INCREMENT BY 1;

-- Triggers for auto-incrementing IDs

CREATE OR REPLACE TRIGGER trg\_users\_ai

BEFORE INSERT ON Coursera\_Users

FOR EACH ROW

BEGIN

:NEW.user\_id := seq\_users.NEXTVAL;

END;

/

CREATE OR REPLACE TRIGGER trg\_courses\_ai

BEFORE INSERT ON Coursera\_Courses

FOR EACH ROW

BEGIN

:NEW.course\_id := seq\_courses.NEXTVAL;

END;

/

CREATE OR REPLACE TRIGGER trg\_enrollments\_ai

BEFORE INSERT ON Coursera\_Enrollments

FOR EACH ROW

BEGIN

:NEW.enrollment\_id := seq\_enrollments.NEXTVAL;

END;

/

CREATE OR REPLACE TRIGGER trg\_certificates\_ai

BEFORE INSERT ON Coursera\_Certificates

FOR EACH ROW

BEGIN

:NEW.certificate\_id := seq\_certificates.NEXTVAL;

END;

/

-- Insert 10 users (students and instructors)

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Abdo Ghazy', 'ghazy@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Steven Amin', 'steven@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Mohamed Sameh', 'sameh@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Omar Mahmoud', 'omar.d@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Khalid', 'khalid@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Dr. Sarah Ali', 'sarah@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Prof. John Doe', 'john@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Dr. Emily Stone', 'emily@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Eng. Ahmed Fathy', 'ahmed@gmail.com', '123');

INSERT INTO Coursera\_Users (full\_name, email, pass) VALUES ('Dr. Mona Hassan', 'mona@gmail.com', '123');

-- Insert COURSERA\_INSTRUCTORS based on existing users who teach courses

INSERT INTO COURSERA\_INSTRUCTORS (instructor\_id, full\_name, email)

VALUES (1006, 'Dr. Sarah Ali', 'sarah@gmail.com');

INSERT INTO COURSERA\_INSTRUCTORS (instructor\_id, full\_name, email)

VALUES (1007, 'Prof. John Doe', 'john@gmail.com');

INSERT INTO COURSERA\_INSTRUCTORS (instructor\_id, full\_name, email)

VALUES (1008, 'Dr. Emily Stone', 'emily@gmail.com');

INSERT INTO COURSERA\_INSTRUCTORS (instructor\_id, full\_name, email)

VALUES (1009, 'Eng. Ahmed Fathy', 'ahmed@gmail.com');

INSERT INTO COURSERA\_INSTRUCTORS (instructor\_id, full\_name, email)

VALUES (1010, 'Dr. Mona Hassan', 'mona@gmail.com');

-- Insert a new instructor

INSERT INTO COURSERA\_INSTRUCTORS (instructor\_id, full\_name, email, bio)

VALUES (seq\_users.NEXTVAL, 'Dr. Alice Brown', 'alice.brown@example.com', 'Ph.D. in Data Science with 10+ years teaching experience.');

-- Insert 10 courses

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Introduction to SQL', 'Learn basic SQL commands and queries', 49.99, 4, 1006);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('OOP Basics', 'Handles the main 4 principles of the OOP', 49.99, 6, 1007);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Web Development', 'Build websites with HTML, CSS, JavaScript', 29.99, 8, 1008);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Machine Learning', 'Introduction to ML algorithms', 59.99, 10, 1009);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Data Structures', 'Learning how to use STLs', 39.99, 5, 1010);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Python for Beginners', 'Intro to Python programming', 19.99, 5, 1006);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Java Essentials', 'Java basics and syntax', 44.99, 6, 1007);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Cybersecurity Basics', 'Learn core concepts in cybersecurity', 39.99, 6, 1008);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Cloud Computing', 'Intro to AWS and cloud concepts', 54.99, 7, 1009);

INSERT INTO Coursera\_Courses (title, description, price, duration\_weeks, instructor\_id) VALUES ('Advanced Algorithms', 'Graph, DP, and complexity topics', 69.99, 8, 1010);

-- Assign new instructor to Web Development course

UPDATE Coursera\_Courses

SET instructor\_id = (SELECT instructor\_id FROM COURSERA\_INSTRUCTORS WHERE email = 'alice.brown@example.com')

WHERE course\_id = 2003;

-- Insert 10 enrollments

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1001, 2001, 'completed');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1002, 2002, 'in\_progress');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1003, 2003, 'not\_started');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1004, 2004, 'in\_progress');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1005, 2005, 'completed');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1001, 2006, 'not\_started');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1002, 2007, 'in\_progress');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1003, 2008, 'completed');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1004, 2009, 'not\_started');

INSERT INTO Coursera\_Enrollments (user\_id, course\_id, completion\_status) VALUES (1005, 2010, 'completed');

-- Insert certificates for completed enrollments only

INSERT INTO Coursera\_Certificates (enrollment\_id) VALUES (1);

INSERT INTO Coursera\_Certificates (enrollment\_id) VALUES (5);

INSERT INTO Coursera\_Certificates (enrollment\_id) VALUES (8);

INSERT INTO Coursera\_Certificates (enrollment\_id) VALUES (10);

-- Create procedures

CREATE OR REPLACE PROCEDURE Get\_Enrollment\_Status (

p\_user\_id IN VARCHAR2,

p\_course\_id IN VARCHAR2,

p\_status OUT VARCHAR2

)

AS

BEGIN

SELECT completion\_status INTO p\_status

FROM Coursera\_Enrollments

WHERE user\_id = p\_user\_id AND course\_id = p\_course\_id;

EXCEPTION

WHEN NO\_DATA\_FOUND THEN

p\_status := 'Not enrolled';

END;

/

-- Procedure 2: Get Student Courses

CREATE OR REPLACE PROCEDURE Get\_Student\_Courses (

p\_user\_id IN VARCHAR2,

p\_result OUT SYS\_REFCURSOR

)

AS

BEGIN

OPEN p\_result FOR

SELECT c.title, e.enrollment\_date, e.completion\_status

FROM Coursera\_Courses c

JOIN Coursera\_Enrollments e ON c.course\_id = e.course\_id

WHERE e.user\_id = p\_user\_id;

END;

/

COMMIT;