

Group 7 - Database Project (Gradebook)

Task 2:

Professor Create Table & Insert Value:

```
CREATE TABLE Professor (  
    professor_id INT PRIMARY KEY,  
    first_name VARCHAR(50),  
    last_name VARCHAR(50),  
    contact_info VARCHAR(50),  
    office_location VARCHAR(50)  
);
```

```
INSERT INTO Professor  
VALUES  
(18571,'john', 'doe', "763-587-9999", "room 37"),  
(18572,'mary', 'jane', "202-777-9489", "room 22"),  
(18573,'monica', 'lewinsky', "404-697-3674", "room 1"),  
(18574,'lana', 'del ray', "625-222-7548", "room 6");
```

Courses Create Table & Insert Value:

```
CREATE TABLE Courses (  
    course_id INT PRIMARY KEY,  
    professor_id INT,  
    department VARCHAR(100),  
    course_section INT,  
    course_name VARCHAR(100),  
    semester VARCHAR(25),  
    year INT,  
    FOREIGN KEY (professor_id) REFERENCES Professor (professor_id)  
);
```

```
INSERT INTO  
Courses(course_id,professor_id,department,course_section,course_name,semester,year)  
VALUES  
(808, 18571,'math', 1, 'linear algebra', 'fall', 2023),  
(607, 18572,'english', 1, 'russian literature', 'fall', 2023),  
(505, 18573, 'science', 1, 'astronomy', 'spring', 2024),  
(304, 18574, 'math', 1, 'pre calculus', 'spring', 2024);
```

Category Create Table & Insert Value:

```
CREATE TABLE Category (  
    category_id INT PRIMARY KEY,  
    category_name VARCHAR(100),  
    weight_percent DECIMAL(10,2)  
);  
INSERT INTO Category  
VALUES  
(1, 'Participation', 0.10),  
(2, 'Homework', 0.20),  
(3, 'Tests', 0.50),  
(4, 'Project', 0.20);
```

Student Create Table & Insert Value:

```
CREATE TABLE Student (  
    student_id INT PRIMARY KEY,  
    first_name VARCHAR(50),  
    last_name VARCHAR(50)  
);  
  
INSERT INTO Student  
VALUES  
(259871, 'steve', 'jobs'),  
(259872, 'ivy', 'Qee'),  
(259873, 'sam', 'smith'),  
(259874, 'victoria', 'paris');
```

Assignment Create Table & Insert Value:

```
CREATE TABLE Assignment (  
    assignment_id INT PRIMARY KEY,  
    category_id INT,  
    course_id INT,  
    assignment_name VARCHAR(100),  
    max_score INT,  
    FOREIGN KEY (category_id) REFERENCES Category (category_id),  
    FOREIGN KEY (course_id) REFERENCES Courses (course_id)  
);  
  
INSERT INTO Assignment(assignment_id, category_id, course_id, assignment_name,  
max_score)
```

VALUES

```
(105, 1, 808, 'Exit ticket 1', 100),
(112, 1, 808, 'Exit ticket 2', 100),
(113, 1, 808, 'Exit ticket 3', 100),
(101, 2, 808, 'Homework 1', 100),
(106, 2, 808, 'Homework 1', 100),
(110, 2, 808, 'Homework 2', 100),
(111, 2, 808, 'Homework 3', 100),
(108, 3, 808, 'Linear Test 1', 100),
(109, 3, 808, 'Linear Test 2', 100),
(114, 4, 808, 'Project 1', 100),
(104, 3, 304, 'Theory Exam', 100),
(107, 2, 304, 'Homework 1', 100),
(102, 3, 607, 'Test 3', 100),
(103, 4, 505, 'Bison Project', 100);
```

Grade Create Table & Insert Value:

```
CREATE TABLE Grade (
grade_id INT PRIMARY KEY,
student_id INT,
assignment_id INT,
course_id INT,
student_score INT,
date DATE,
FOREIGN KEY (student_id) REFERENCES Student (student_id),
FOREIGN KEY (assignment_id) REFERENCES Assignment (assignment_id),
FOREIGN KEY (course_id) REFERENCES Courses (course_id)
);
```

```
INSERT INTO Grade (grade_id, student_id, assignment_id, course_id, student_score, date)
VALUES
```

```
(201, 259871, 101, 808, 65, '2024-04-15'),
(207, 259871, 107, 304, 86, '2024-04-15'),
(202, 259872, 102, 607, 75, '2024-04-15'),
(208, 259872, 110, 808, 100, '2024-04-15'),
(209, 259872, 111, 808, 70, '2024-04-15'),
(210, 259872, 108, 808, 89, '2024-04-15'),
(211, 259872, 109, 808, 79, '2024-04-15'),
(212, 259872, 114, 808, 100, '2024-04-15'),
(215, 259872, 112, 808, 45, '2024-04-15'),
(216, 259872, 113, 808, 32, '2024-04-15'),
(203, 259873, 103, 505, 85, '2024-04-15'),
```

(204, 259874, 104, 304, 65, '2024-04-15'),
(205, 259874, 106, 808, 40, '2024-04-15'),
(206, 259874, 105, 808, 67, '2024-04-15');

Task 3:

Professor Table:

	professor_id	first_name	last_name	contact_info	office_location
▶	18571	john	doe	763-587-9999	room 37
	18572	mary	jane	202-777-9489	room 22
	18573	monica	lewinsky	404-697-3674	room 1
	18574	lana	del ray	625-222-7548	room 6
•	NULL	NULL	NULL	NULL	NULL

Course Table:

	course_id	professor_id	department	course_section	course_name	semester	year
▶	304	18574	math	1	pre calculus	spring	2024
	505	18573	science	1	astronomy	spring	2024
	607	18572	english	1	russian literature	fall	2023
	808	18571	math	1	linear algebra	fall	2023
•	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Category Table:

	category_id	category_name	weight_percent
▶	1	Participation	0.10
	2	Homework	0.20
	3	Tests	0.50
	4	Project	0.20
•	NULL	NULL	NULL

Assignment Table:

	assignment_id	category_id	course_id	assignment_name	max_score
▶	101	2	808	Homework 1	100
	102	3	607	Test 3	100
	103	4	505	Bison Project	100
	104	3	304	Theory Exam	100
	105	1	808	Exit ticket 1	100
	106	2	808	Homework 1	100
	107	2	304	Homework 1	100
	108	3	808	Linear Test 1	100
	109	3	808	Linear Test 2	100
	110	2	808	Homework 2	100
•	NULL	NULL	NULL	NULL	NULL

Student Table:

	student_id	first_name	last_name
▶	259871	steve	jobs
	259872	ivy	Qee
	259873	sam	smith
	259874	victoria	paris
•	NULL	NULL	NULL

Grade Table:

	grade_id	student_id	assignment_id	course_id	student_score	date
▶	201	259871	101	808	65	2024-04-15
	202	259872	102	607	75	2024-04-15
	203	259873	103	505	85	2024-04-15
	204	259874	104	304	65	2024-04-15
	205	259874	106	808	40	2024-04-15
	206	259874	105	808	67	2024-04-15
	207	259871	107	304	86	2024-04-15
	208	259872	110	808	100	2024-04-15
	209	259872	111	808	70	2024-04-15
	210	259872	108	808	89	2024-04-15
•	NULL	NULL	NULL	NULL	NULL	NULL

Task 4:

```

SELECT AVG(G.student_score) AS average_score
FROM Grade G
JOIN Assignment A ON G.assignment_id = A.assignment_id
WHERE A.assignment_name = 'Homework 1' -- Change 'Homework 1' to the desired
assignment name
AND A.course_id = 808; -- Change 808 to the desired course_id

```

```

SELECT MAX(student_score) as max_score
FROM Grade as GR JOIN Assignment as A ON GR.assignment_id = A.assignment_id
WHERE assignment_name = 'Homework 1';

```

```

SELECT MIN(student_score) as min_score
FROM Grade as GR JOIN Assignment as A ON GR.assignment_id = A.assignment_id
WHERE assignment_name = 'Homework 1';

```

Task 5:

```

SELECT DISTINCT
    C.course_name,
    GR.course_id,

```

```
GR.student_id,  
ST.first_name,  
ST.last_name  
FROM Grade as GR JOIN Student as ST ON GR.student_id = ST.student_id  
      JOIN Courses as C ON C.course_id = GR.course_id  
WHERE GR.course_id = 808;
```

Task 6:

```
SELECT  
      ST.student_id,  
      ST.first_name,  
      ST.last_name,  
      A.assignment_name,  
      GR.student_score  
FROM  
      Grade AS GR  
JOIN  
      Student as ST ON GR.student_id = ST.student_id  
JOIN  
      Assignment AS A ON GR.assignment_id = A.assignment_id  
JOIN  
      Courses AS C ON GR.course_id = C.course_id  
WHERE  
      C.course_id = 808;
```

Task 7:

```
INSERT INTO Assignment(assignment_id, category_id, course_id, assignment_name,  
max_score)  
VALUES  
(115, 4, 607, 'Russian Essay', 100);  
SELECT *  
FROM Assignment;
```

Task 8:

```
UPDATE Category
SET weight_percent = 0.15
WHERE category_id= 1 ;
```

Task 9:

```
UPDATE Grade
SET student_score = student_score + 2
WHERE grade_id = 201;
```

Task 10:

```
UPDATE Grade as GR
JOIN Student as ST ON GR.student_id = ST.student_id JOIN Courses AS C ON C.course_id =
GR.course_id
SET GR.student_score = GR.student_score + 2
WHERE ST.last_name ='Qee' AND GR.course_id = 808;
```

Task 11:

```
SELECT
    ST.student_id,
    ST.first_name,
    ST.last_name,
    C.course_id,
    calc_category_score(ST.student_id, C.course_id, 1) AS part_score,
    calc_category_score(ST.student_id, C.course_id, 2) AS homework_score,
    calc_category_score(ST.student_id, C.course_id, 3) AS test_score,
    calc_category_score(ST.student_id, C.course_id, 4) AS project_score,
    ((calc_category_score(ST.student_id, C.course_id, 1) +
    calc_category_score(ST.student_id, C.course_id, 2) +
    calc_category_score(ST.student_id, C.course_id, 3) +
    calc_category_score(ST.student_id, C.course_id, 4)) * 100) AS final_grade
FROM
    Student AS ST,
    Courses AS C
ORDER BY ST.student_id;
```


Task 12:

```
SELECT
    ST.student_id,
    ST.first_name,
    ST.last_name,
    C.course_id,
    calc_category_score_with_drop(ST.student_id, C.course_id) AS final_grade
FROM
    Student AS ST,
    Courses AS C
ORDER BY
    ST.student_id;
```