



Xi'an Jiaotong-Liverpool University

西交利物浦大学

## XJTLU Entrepreneur College (Taicang) Cover Sheet

Module code and Title	Database Development and Design (DTS207TC)
School Title	School of AI and Advanced Computing
Assignment Title	001: Assessment Task 1 (CW)

Final Word Count	NA
If you agree to let the university use your work anonymously for teaching and learning purposes, please type "yes" here.	<b>Yes</b>

I certify that I have read and understood the University's Policy for dealing with Plagiarism, Collusion and the Fabrication of Data (available on Learning Mall Online). With reference to this policy I certify that:

- My work does not contain any instances of plagiarism and/or collusion.
- My work does not contain any fabricated data.
- My work does not contain any text or code from the Internet, Generative AI, or published textbooks

**By uploading my assignment onto Learning Mall Online, I formally declare that all of the above information is true to the best of my knowledge and belief.**

Scoring – For Tutor Use						
Student ID						
Stage of Marking		Marker Code	Learning Outcomes Achieved (F/P/M/D) (please modify as appropriate)			Final Score
			B	C	D	
1 <sup>st</sup> Marker – red pen						
Moderation – green pen		IM Initials	The original mark has been accepted by the moderator (please circle as appropriate):			Y / N
			Data entry and score calculation have been checked by another tutor (please circle):			
2 <sup>nd</sup> Marker if needed – green pen						
For Academic Office Use			Possible Academic Infringement (please tick as appropriate)			
Date Received	Days late	Late Penalty	<input type="checkbox"/> Category A		Total Academic Infringement Penalty (A,B, C, D, E, Please modify where necessary) _____	
			<input type="checkbox"/> Category B			
			<input type="checkbox"/> Category C			



			<input type="checkbox"/> Category D	
			<input type="checkbox"/> Category E	

**Database Development and Design (DTS207TC)****Assessment 001: Individual Coursework****Weight:** 60%**Maximum Marks:** 100**Overview & Outcomes**

This course work will be assessed for the following learning outcomes:

- B. Demonstrate an understanding of advanced SQL topics.
- C. Illustrate the issues related to Web technologies and DBMS and XML as a semi-structured data representation formalism.
- D. Identify the principles underlying object-relational models.

**Submission**

You must submit the following files to LMO:

- 1)A report named as Your\_Student\_ID.pdf.
- 2)A directory containing all your source code, named as Your\_Student\_ID\_code.

NOTE: The report shall be in A4 size, size 11 font, and shall not exceed **15** pages in length. You can include only key code snippets in your reports. The complete source code can be placed in the attachment.

**Question 1: Intermediate SQL (20 marks)**

You have been provided with three CSV files, **which content is given in next page within this document**, from the Academic Affairs Office of a university, containing information about students, courses, and course enrollments. Your task is to perform the following operations in a PostgreSQL database:

1. Table Creation (10 marks):

- Create tables with suitable data types for *students*, *courses*, and course *enrollments* based on the data in the CSV files.

2. View Creation (5 marks):

- Create a view named *student\_course\_view* that includes the student's name, course name, grade, and GPA. The results should be sorted by the student's name in ascending order.



3. Query Execution (5 marks):

- Execute a query to retrieve data from the `student_course_view` and provide a screenshot of the query results.

**Students.csv**

student_id	name	gender	major	gpa
1	Alice	F	Computer Science	3.5
2	Bob	M	Mathematics	3.8
3	Charlie	M	Physics	3.2
4	Diana	F	Computer Science	3.9
5	Ethan	M	Mathematics	3.6
6	Fiona	F	Physics	3.7
7	George	M	Computer Science	3.4
8	Hannah	F	Mathematics	3.3
9	Ian	M	Physics	3.1
10	Julia	F	Computer Science	3.0

**Courses.csv**

course_id	course_name	credits	department
1	Database Systems	4	Computer Science
2	Calculus	3	Mathematics
3	Quantum Mechanics	5	Physics
4	Operating Systems	4	Computer Science
5	Linear Algebra	3	Mathematics
6	Classical Mechanics	5	Physics
7	Data Structures	4	Computer Science
8	Differential Equations	3	Mathematics
9	Electromagnetism	5	Physics

--

**Enrollments.csv**

enrollment_id,student_id,course_id,grade
--

1,1,1,85

2,2,2,90

3,3,3,78

4,4,1,92

5,5,2,88

6,6,3,85

7,7,4,80

8,8,5,82

9,9,6,76

10,10,7,88

11,1,8,90

12,2,9,85

13,3,1,80

14,4,2,88

15,5,3,92

16,6,4,85

17,7,5,80

18,8,6,82

19,9,7,76

20,10,8,88

试用水印

**Question 2: Advanced SQL (20 marks)**

You are maintaining a student information management system that uses a PostgreSQL database. The database contains tables for students, classes, and grades with some existing data, as follows:



## SQL

-- Create a class table

```
CREATE TABLE classes (
    class_id SERIAL PRIMARY KEY,
    class_name VARCHAR(100) NOT NULL,
    average_score FLOAT
);
```

-- Create a student table

```
CREATE TABLE students (
    student_id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    age INT,
    class_id INT,
    FOREIGN KEY (class_id) REFERENCES classes(class_id)
);
```

-- Create a grade sheet

```
CREATE TABLE grades (
    grade_id SERIAL PRIMARY KEY,
    student_id INT,
    course_id INT,
    score FLOAT,
    FOREIGN KEY (student_id) REFERENCES students(student_id)
);
```

-- Insert class data

```
INSERT INTO classes (class_name) VALUES
```



```
('Class A'),
```

```
('Class B'),
```

```
('Class C');
```

```
-- Insert student data
```

```
INSERT INTO students (name, age, class_id) VALUES
```

```
('Alice', 20, 1),
```

```
('Bob', 22, 2),
```

```
('Charlie', 21, 1),
```

```
('David', 23, 3),
```

```
('Eve', 24, 2);
```

```
-- Insert grade data
```

```
INSERT INTO grades (student_id, course_id, score) VALUES
```

```
(1, 1001, 85.5),
```

```
(2, 1002, 90.0),
```

```
(3, 1001, 78.0),
```

```
(4, 1002, 88.5),
```

```
(5, 1001, 92.0);
```

### Task

1. Create a stored procedure called `insert_or_update_student` to insert or update a new student into the student table. The stored procedure requires the student's name, age, and class ID as parameters. (We assume that there will be no students with the same name.) (8 marks)
2. Create a trigger to automatically calculate and update the class average when a grade is inserted or updated in the grades table. The class average is stored in a new field called `average_score` in the class table. (8 marks)
3. Test your program using the following statements and provide a screenshot of the results.(4 marks)

### SQL

```
-- Insert a new student
```

```
CALL insert_or_update_student('Frank', 25, 1);
```

-- Insert a new grade

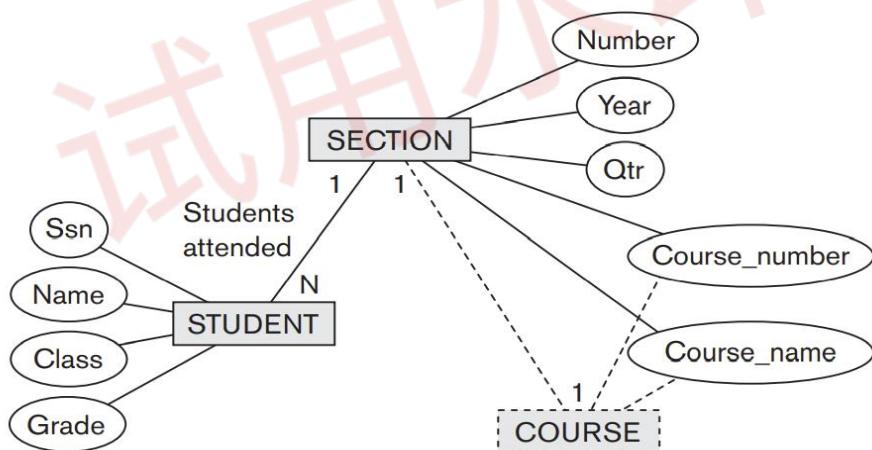
```
INSERT INTO grades (student_id, course_id, score) VALUES (1, 1003, 95.0);
```

-- Update an existing grade

```
UPDATE grades SET score = 98.0 WHERE student_id = 1 AND course_id = 1003;
```

### **Question 3: Complex Data Types (20 marks)**

Suppose that an application needs to extract XML documents for student, course, and grade information from a UNIVERSITY database. The data needed for this information is stored in the database attributes of the entity COURSE, SECTION, and STUDENT with One-To-One and One-To-Many relationship of SECTION with COURSE and STUDENT. Consider the following Hierarchical (tree) view for a subset of a University database and answer the following questions:



**(a)** Create XML DTD that corresponds to the given ER Diagram (the dotted box represents a weak entity). The DTD should be validated using the XML validation tool. (EditiX) (10 marks)

**(b)** Create an XML instance document corresponding to the DTD you have created in the first step. The XML document must have at least one record for every element specified in the DTD. Validate the instance document against the DTD using the XML validation tool. (EditiX) (10 marks)

### **Question 4: ORM (20 marks)**

Consider the following set of requirements for a UNIVERSITY database that is used to keep track of students' transcripts.

- The university keeps track of each student's name, student number, Social Security number, current address and phone number, permanent address and phone number, birth date, sex, class (freshman, sophomore, ..., graduate), major department, minor department (if any), and degree program (B.A., B.S., ..., Ph.D.). Some user applications need to refer to the city, state, and ZIP Code of the student's permanent address and to the student's last name. Both Social Security number and student number have unique values for each student.
  - Each department is described by a name, department code, office number, office phone number, and college. Both name and code have unique values for each department.
  - Each course has a course name, description, course number, number of semester hours, level, and offering department. The value of the course number is unique for each course.
  - Each section has semester year, course, and section number. The section number distinguishes sections of the same course taught during the same semester/year; its values are 1, 2, 3, ..., up to the number of sections taught during each semester.
- (a) Design a UML class diagram for this application. Specify key attributes of each entity type and structural constraints on each relationship type. Note any unspecified requirements, and make appropriate assumptions to make the specification complete. (10 marks)
- (b) Translate the UML class diagram created in the above question (a) into a relational schema. In the relational schema, only use relation name and attribute, e.g., student (name, number, SSN,...etc.) and specify foreign key as fk. (10 marks)

### **Question 5: Data Warehousing (20 marks)**

---

Suppose a company would like to design a data warehouse to facilitate vehicle analysis in an online analytical processing manner. The company registers huge amounts of vehicles' data in the format of (Vehicle ID, Driver ID, location, speed, and time). Each Vehicle ID represents a vehicle associated with information, such as vehicle type, plate number, etc., each Driver ID represents information such as ID, name, age etc., and each location may be associated with a street in a city. Assume that a street map is available for the city. Speed represents miles/hour and time represents day, month, quarter and year. The company is interested to measure the total number of vehicles at a specific location and time as well as average mileage.

- (a) State which schema (star, snowflake etc.) is the most appropriate to model the above data warehouse. Give your opinion of which might be more empirically useful and state the reasons behind your answer. (6 marks)
- (b) Draw a schema diagram for the above data warehouse using one of the most appropriate schema diagrams. Dimension and fact tables must have all primary keys by introducing the attribute that identifies each entity. Furthermore, a complete schema diagram must capture associations among all the entities to perform OLAP operations. (6 marks)



- (c) A driver may like to perform online analytical processing to determine how to move fastest from one location to another location at a particular time. Discuss how this can be done efficiently by using data stored in the warehouse. (8 marks)

试用水印



## Marking Criteria

The tasks in this assessment can be divided into 3 categories:

- ✓ Charts Presentation & Analysis;
- ✓ Essay;
- ✓ Programs.

Criteria(%)	Exemplary (100)	Good (75)	Satisfactory (50)	Limited (25)	Very Limited (0)
Design	Provides a detailed, accurate description of the methods. Provide comprehensive comparison between the methods, including pros and cons, performance analysis.	The analysis provided demonstrates that the student's understanding of the various methods is correct and that they have the ability to solve problems independently. Although there are certain flaws, or incomplete.	Provides adequate description of the methods. Comparison is provided with some level of details, however, with some obvious mistakes.	There are obvious deviations in the understanding of the main methods, and it fails to reflect the ability to independently design algorithms. The description of the problem is vague, or the thought is incomplete.	Limited or no description of methods. Limited comparison provided.
Programs	Demonstrated correctly implemented code that produces correct output. Excellent coding quality follows best practices.	The program runs correctly and gives the expected results. However, special cases are not fully considered, or the program performs redundant calculations.	Program basically works correctly for major functionality, however, with some conceptional problems.	The program implements some minor functionality, or incorrectly implements major functionality. There is a certain degree of misunderstanding about the requirements of the questions.	Program works incorrectly with limited attempt or irrelevant to the task.
Charts Presentation & Analysis	Excellent quality of report with clear structure, clear logic,	Most of the results in the chart are correct, but there is a	Moderate quality of report with basic structure,	Only some of the results in the chart are correct, or some of them are not filled in.	Limited or no attempt of report.



Xi'an Jiaotong-Liverpool University

西交利物浦大学

	concise writing, pleasing visual aids.	certain degree of sloppy or wordy in the overview and analysis.	where writing and visual aids can be improved.	The analysis of the results was obviously biased.	
--	--	---	--	---	--

The mark allocations for the above tasks are:

Question	Design	Programs	Charts Presentation & Analysis
1	10	10	
2	10	10	
3	10	10	
4	10		10
5	10		10

试用水印