School: Computer Science
Institution: University of Windsor
Term: Fall 2019
Course: Comp-3150 (03-60-315-1) : Database Management Systems
Instructor: Dr. C. I. Ezeife
Assignment #1 : Total: 50 marks
Handed Out:Thurs. Sept 12, 2019; Due Thurs Sept 26, 2019

Objective of Assignment: To test on knowledge of database concepts and its 3-level architecture necessary for designing databases and their applications as well as practice on use of entity-relationship (ER) model to design databases.
Scope: Assignment covers materials from Chapters 1, 2 and 3 of book discussed in class.
Electronic Assignment Submission: Done through http://blackboard.uwindsor.ca
Marking Scheme: The mark for each of the questions is indicated beside each question.
Academic Integrity Statement: Remember to submit only work that is yours and include the following confidentiality agreement and statement at the beginning of your assignment.

CONFIDENTIALITY AGREEMENT & STATEMENT OF HONESTY
I confirm that I will keep the content of this assignment/examination confidential.
I confirm that I have not received any unauthorized assistance in preparing for or doing this assignment/examination. I confirm knowing that a mark of 0 may be assigned for copied work.

Andrea Bonato
- ______________________ ___AndreaBonato________
Student Signature Student Name (please print)
__104760390________________________ __2019-09-21________________
Student I.D. Number Date

Marking Scheme: The mark for each question and sub question is shown with the question below. Place your solutions in tables provided for answers where possible.

For office Use only

Question	**Mark**
1	**/10**
2	**/10**
3	**/10**
4	**/20**
Total	**/50**

CHAPTER 1: DATABASES AND DATABASE USERS

1. Given the simple Employee-Workson-Project database schema that contains three files described as follows, answer the following questions with regards to this database.

(Total for que 1 is 10 marks)

Emp (eid : integer, ename : string, age : integer, salary: real)
Workson (eid : integer, pid : integer, hours : integer)
Project (pid : integer, pname : string, budget : real, managerid : integer)

Note : eid, ename, age and salary are the employee id, name, age and salary respectively. Also, hours is the number of hours worked by employee on a project. The rest of the attributes pid, pname, budget and managerid are the project id, name, budget and managerid respectively. A manager is an employee.

i) Create a valid instance of this database containing values for its records with at least four records in each file. A valid instance does not violate any database integrity contraints.
(3 marks)

ii) Provide 2 informal English queries from this database with their answers. Each query should involve at least 2 of the files in the database and your answer should indicate the files (e.g., Emp, Workson) needed to answer each query and specify what fields are being retrieved as the result (e.g., ename, age). Please, provide your solution in the 3 column table below.
(4 marks)

iii) Specify at least 3 relationships with other database files (one for each of the 3 database files) among the records of the database. For each file (e.g., Emp) list any relationships it has with other files through its fields (e.g., eid). Provide your solution using the table below.
(3 marks)

Solution : (3 marks for 1)

<table>
<tr><th>Query</th><th>Answer</th><th>Files involved</th></tr>
<tr>
<td>1. Create a valid instance of this database containing values for its records with at least four records in each file.

(3 marks)</td>
<td>An instance of the Employee-Workson-Project database is :
Emp
<table>
<tr><th>EID</th><th>EName</th><th>Age</th><th>Salary</th></tr>
<tr><td>1</td><td>Carla Jones</td><td>19</td><td>15000</td></tr>
<tr><td>2</td><td>Jessica Roberts</td><td>56</td><td>11600</td></tr>
<tr><td>3</td><td>Roberto Giovanni</td><td>23</td><td>14000</td></tr>
<tr><td>4</td><td>Costa Vinci</td><td>29</td><td>45.55</td></tr>
<tr><td>23</td><td>Marcus Manage</td><td>52</td><td>56.00</td></tr>
</table>
Workson
<table>
<tr><th>PID</th><th>EID</th><th>Hours</th></tr>
<tr><td>5</td><td>1</td><td>26</td></tr>
<tr><td>6</td><td>2</td><td>48</td></tr>
<tr><td>7</td><td>3</td><td>156</td></tr>
<tr><td>8</td><td>4</td><td>86</td></tr>
</table>
Project</td>
<td>-----
Emp, Project and Workson</td>
</tr>
</table>

<table>
<tr>
<td></td>
<td>
<table>
<tr><th>PID</th><th>PName</th><th>Budget</th><th>ManagerID</th></tr>
<tr><td>5</td><td>APKJ</td><td>26000.00</td><td>23</td></tr>
<tr><td>6</td><td>Wires</td><td>15000.00</td><td>23</td></tr>
<tr><td>7</td><td>Alpha</td><td>625.00</td><td>23</td></tr>
<tr><td>8</td><td>Proj5.6</td><td>16000.00</td><td>23</td></tr>
</table>
(Assuming a manager can lead multiple projects)
</td>
<td></td>
</tr>
<tr>
<td>2. Provide 2 informal English queries from this database with their answers. Each query should involve at least 2 of the files in the database and your answer should indicate the files (e.g., Emp, Workson) needed to answer each query and specify what fields are being retrieved as the result (e.g., ename, age).
(4 marks)

i. Managers check to see which employees are assigned to which projects

ii. Managers check to see the budget for a specific project and determines which employees are on it based on pay</td>
<td>Result of query
i. These are the employees that are assigned to the projects
<table>
<tr><th>Emp: EID</th><th>Project: PID</th><th>Project: Name</th><th>Emp: Name</th></tr>
<tr><td>1</td><td>5</td><td>APKJ</td><td>Carla Jones</td></tr>
<tr><td>2</td><td>6</td><td>Wires</td><td>Jessica Roberts</td></tr>
<tr><td>3</td><td>7</td><td>Alpha</td><td>Roberto Giovanni</td></tr>
<tr><td>4</td><td>8</td><td>Proj5.6</td><td>Costa Vinci</td></tr>
</table>
ii. These are the budgets in comparison to pay for specific employees
<table>
<tr><th>Project Budget</th><th>Project PID</th><th>Project Name</th><th>Emp EID</th><th>Emp: Salary</th></tr>
<tr><td>26000.00</td><td>5</td><td>APKJ</td><td>1</td><td>15.60</td></tr>
<tr><td>15000.00</td><td>6</td><td>Wires</td><td>2</td><td>11.60</td></tr>
<tr><td>625.00</td><td>7</td><td>Alpha</td><td>3</td><td>14.00</td></tr>
<tr><td>16000.00</td><td>8</td><td>Proj5.6</td><td>4</td><td>19.50</td></tr>
</table>
</td>
<td>i. Project and Emp

ii. Project and Emp</td>
</tr>
<tr>
<td>3. Specify at least 3 relationships with other database files (one for each of the 3 database files) among the records of the database. For each file (e.g., Emp) list any relationships it has with other files through its fields (e.g., eid). Provide your solution using the table below.

(3 marks)</td>
<td>1. Project has a relationship through PID which connects it to Workson
2. Emp has a relationship through EID which connects it to Workson
3. Workson has a relationship through EID and PID which connects it to both Project and Emp</td>
<td>Project, Emp and Workson</td>
</tr>
</table>

2. Recall that a database has many types of users, each of whom may require a different view of the database. For example, one user of the Employee-Workson-project database of question 1 may be accessing and printing the details and salaries of each employee frequently and thus a view for this user is created. Another view for this database is checking that project has available budget before expenditure such as paying salaries. (Total for que 2 is 10 marks)

i) Using this Employee-Workson-Project database, give 2 additional views that may be needed by other user groups for the database. (5 marks)

Solution : (5 marks for 2i)

One additional view may be a for user that needs to see a printout of the hours and budget associated with a project to help determine how the budget will be broken down, and what they can afford to pay out as possible salary. Generally, to help budget and manage the time expenses.
Another additional view may be a for user that needs to see a printout of the people that are assigned to specific projects. This would be to manage who is on a project, and who is available to be put on a project.
A third possible view may be for a user that needs to see a printout of all the projects that the company has ongoing and the hours worked associated to them. This may be used to determine whether a company can or can not accept additional projects and generally see if they are on schedule.

ii) Give 5 examples of integrity constraints that you think can apply to the Employee-Workson-Project database of question 1. (5 marks)

Solution : (5 marks for 2ii)

Some possible integrity constraints that we can apply to the Employee-Workson-Project database in question 1 is:

- Each emp must have a unique employee ID (EID). This is a uniqueness constraint.
- Each project must have a unique project ID (PID). This is also a uniqueness constraint.
- Every project must have an emp assigned to work on it. Thus, each PID must have an EID attached to it through Workson. This is a referential integrity.
- The salary attribute associated with Emp must be a float, thus must not include symbols and punctuation. This is an integrity constraint.
- The manager must be an employee, and must have a unique id, or managerid, associated with them. This is a uniqueness constraint.
- A manager can oversee multiple projects.

CHAPTER 2: DATABASE SYSTEM CONCEPTS AND ARCHITECTURE

3.a. Design a simple database schema with not more than 4 files for a University database system indicating all applicable integrity constraints and information. Also, show a sample database state for the database. (5 marks for a)

b. Using your database, describe the difference between logical and physical data independence. (5 marks for b)

(Total for que 3 is 10 marks)

Question	Answers
a. Design a simple database schema with not more than 3 files for a University database system indicating all applicable constraints and information. Also, show a sample database state for	Simple University database schema is: Student (Name, Student_ID, Course_ID, Major) Course (Course_name, Course_Number, Credit_hours, Department) Grade (Student_ID, Course_Number, Grade) Some integrity constraints are:

the database.
(5 marks)

- In order to be in a course, you need to be a student with a student ID.
- In order to get a grade for a specific course, the course must have a registered course number, and the person must be a registered student with a student ID.

A state of this database is:

A state of this database is the content of the database at the current moment in time.

So, the state would look like:

Student:

Name	**Student_ID**	**Course_ID**	**Major**
Andrea Bonato	104760390	COMP354	Computer Science
Carla Jones	104760391	COMP312	Computer Science
Jacob Difazio	104760392	COMP312	Computer Science

Course:

Course_name	**Course_Number**	**Credit_Hours**	**Department**
Databases	COMP354	72	Comp Sci
Obj Oriented	COMP312	89	Comp Sci

Grade

Student_ID	**Course_Number**	**Grade**
104760390	COMP354	95
104760391	COMP312	86
104760392	COMP3128	92

b. Using your database, describe the difference between logical and physical data independence

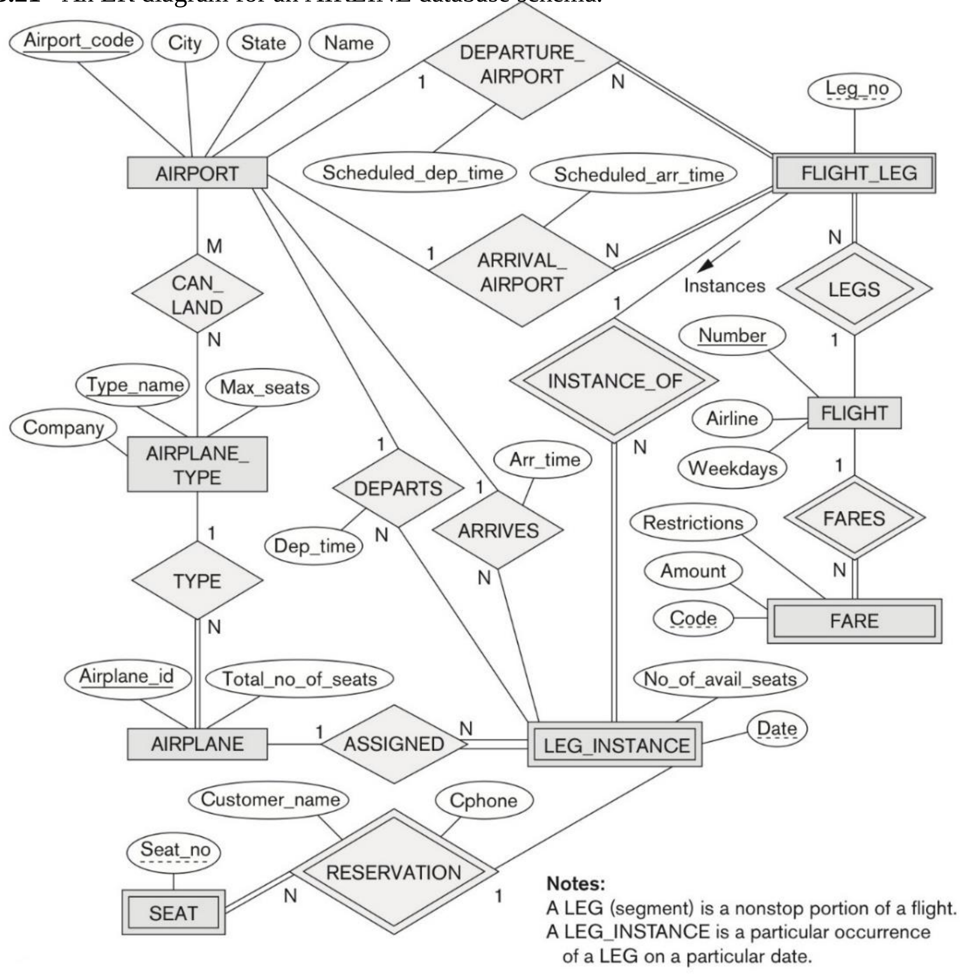
Logical data independence is the ability for the schema of the database to change on the conceptual and external level. For example, the external schema should not be affected by changing the Grade file that is shown above. If we manage and change only the view definition, then the database supports logical data independence.
Physical data independence is the ability for the schema of the database to change on the internal level, without changing the conceptual schema. For example, if we change the path of the file Student or Course to increase efficiency, then we are making a change to the actual physical database on the internal level. These changes to the internal schema, without having to change the conceptual schema, will support physical data independence.

CHAPTER 3: DATA MODELING USING THE ENTITY-RELATIONSHIP (ER) MODEL

4- Consider the ER diagram of Figure 3.21 given below, which shows a simplified schema for an airline reservations system. Using this ER diagram, provide the requirements description of this database by (a) clearly identifying all the entities, b) the relationships connecting the entities, c) the constraints in this schema (cardinality, participation) and d) the sentences specifying the requirements of the database whose design is the ER diagram. Try to be as precise as possible in your requirements and constraints specification. An example interpretation of the constraint label of 1 to N for DEPARTURE_AIRPORT is 'Each Airport has many flight legs departing from it'. An example part of the requirement specification sentences is : `The database represents each AIRPORT, keeping its unique AirportCode, the AIRPORT Name, and the City and State in which the AIRPORT is located.'
Place your answers in the table provided below. (Total for que 4 is 20 marks)

(Note : 10 marks for correct entity and relationship identifications with their attributes (5 for entities and 5 for relationships), 5 marks for correct constraints interpretations on the edge labellings, 5 marks for correct verbal interpretations of the database being represented by the ER digram.

Figure 3.21 An ER diagram for an AIRLINE database schema.



Solution :

(Total for que 4 is 20 marks)

Specific Requrieement/Constraint Type	**Requirements and Constraints from the ER diagram**

Entities and attributes (5 marks)	The entities are AIRPORT, FLIGHT_LEG, FLIGHT, FARE, LEG_INSTANCE, AIRPLANE_TYPE, AIRPLANE and SEAT. The attributes of each entity are as followed : - AIRPORT: Airport_code, City, State and Name - FLIGHT_LEG: Leg_no - FLIGHT: Number, Airline and Weekdays - FARE : Restrictions, Amount and Code - LEG_INSTANCE: Date and No_of_avail_seats - AIRPLANE_TYPE: Company, Type_name and Max_seats - AIRPLANE: Airplane_id and Total_no_of_seats - SEAT: Seat_no
Relationships and attributes (5 marks)	The relationships are DEPARTURE_AIRPORT, ARIVAL_AIRPORT, INSTANCE_OF, LEGS, FARES, ARRIVES, DEPARTS, ASSIGNED, TYPE, RESERVATION and CANLAND The attributes of each relationship are as followed: - DEPARTURE_AIRPORT: Scheduled_dep_time - ARIVAL_AIRPORT: Scheduled_arr_time - INSTANCE_OF: none - LEGS: none - FARES: none - ARRIVES: Arr_time - DEPARTS: Dep_time - ASSIGNED: - TYPE: none - RESERVATION: Customer_name and Cphone - CANLAND: none
Interpretation of each of the constraints represented on the edge labels (5 marks)	The interpretation of each of the constraints represented on the edge labels are: - FLIGHT_LEG:FLIGHT is a constraint with a N:1 cardinality ratio. The constraint is each FLIGHT can be related to any number of FLIGHT_LEG through the relationship LEGS. The interpretation is "Each flight can have many nonstop portions of the flight". - FLIGHT:FARE is a constraint with a 1:N cardinality ration. The constraint is each FLIGHT can be related to any number of FARE through the relationship FARES. The interpretation is "Each flight can have many different prices". - SEAT:LEG_INSTANCE is a constraint with a N:1 cardinality ratio. The constraint is each LEG_INSTANCE can be related to any number of SEAT through the relationship RESERVATION. The interpretation is "Any number of seats on the plane can have a nonstop portion of the flight reserved" - AIRPLANE:LEG_INSTANCE is a constraint with a 1:N cardinality ratio. The constraint is each AIRPLANE can be related to any number of AIRPLANE through the relationship ASSIGNED. The interpretation is "An airplane can have many assigned nonstop portions of a flight".

	- AIRPLANE_TYPE:AIRPLANE is a constraint with a 1:N cardinality ratio. The constraint is each AIRPLANE_TYPE can relate to any number of AIRPLANE through the relationship TYPE. The interpretation is “Each airplane has an airplane type and there are many airplanes with the same type”. - AIRPORT:AIRPLANE_TYPE is a constraint with a M:N cardinality ratio. The constraint is that an AIRPORT can have many AIRPLANE and an AIRPLANE can have many AIRPORT. This relationship is through CAN_LAND. The interpretation is “Many airplanes can land at many different airports”. - AIRPORT:FLIGHT_LEG is a constraint with a 1:N cardinality ration. The constraint is that AIRPORT can be related to any number of FLIGHT_LEG through both the relationships DEPARTURE_AIRPORT and ARRIVAL_ARRIVAL. The interpretation is “Each airport has multiple departing and arriving nonstop flights on a particular day”.
Verbal requirements description in sentences of the database (5 marks)	***Answer:*** The verbal requirements description in sentences of the database is: - The database represents each AIRPORT, keeping its unique AirportCode, the AIRPORT Name, and the City and State in which the AIRPORT is located. - The database represents each AIRPLANE_TYPE, keeping its unique Type_name, the maximum number of seats that this AIRPLANE_TYPE can hold, and the Company that produced and owns the plane. - The database represents each AIRPLANE, keeping its unique Airplane ID and the total number of seats assigned on this plane. - The database represents each SEAT, keeping a derived number of seats that are reserved. - The database represents LEG_INSTANCE, keeping a partial unique date at which the nonstop portion of a flight with occur as well as the number of available seats for that LEG_INSTANCE. - The database represents FARE, keeping a partial unique code as well as the amount that the FARE costs, and the restrictions to the FARE - The database represents FLIGHT, keeping a unique number, or ID of the flight, as well as the Airline that the FLIGHT is a part of and the weekdays that the FLIGHT takes place. - The database represents FLIGHT_LEG, keeping only a partial unique number of nonstop portions of the flight.

	.