|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Part** | **1** | **2** | **3** | **4** | **Total** |
| *maximum* | **25** points | **25** points | **25** points | **25** points | **100**G101010 pointsG |
| ***Your Score*** |  |  |  |  |  |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Topic: Data Models**

Reading Assignment: Thoroughly read Chapter 2 in the course textbook(s).

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part 1 Glossary Terms - Compare and Contrast**

Review, in detail, each of these glossary terms from the realm of computer database systems and computer topics, in general. Then, in your own words, compare and contrast the terms.   
If applicable, use examples to support your definitions. Consult your notes or course textbook(s) as references.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(a) Attribute versus Relationship**

|  |
| --- |
| **Attribute:**  An Attribute is one of the traits or properties of a particular entity inside a database. While attributes define different aspects or characteristics of such things, entities are actual objects. Attributes are just the data fields that contain details about an entity. The entities in a database may be defined, identified, and categorized using attributes. Similar to fields in conventional file systems, attributes hold information relevant to the item they represent. When storing and retrieving data in a relational database, attributes are arranged into tables and columns based on their values.  **Example**:  Consider a company’s database. A company will have employees, so consider “Employee” as an entity. So, the attributes which can be included in this entity can be “Employee Name”, “Employee ID”, “Employee Date of birth”, “Employee salary”, etc.  **Relationship:**  In the context of computer database systems, a relationship refers to the link between multiple entities. It outlines the relationships and interactions between entities in the database. When modelling complicated real-world scenarios and maintaining data integrity, relationships are crucial for establishing sensible connections between various entities. Relationships are mainly of 3 types: One to one (1:1), One to many (1:M) and Many to Many (M:M or M:N). The following is an example of one of those.  **Example:**  Consider a Sports team’s database. Here we can have a relationship between two entities, which are “Players” and “Coach”. This relationship can be called “One to many relationship”. Multiple team members can be managed and trained by a single coach. The key player who offers direction, creates strategy, and oversees the team's overall performance is the coach. |

**(b) Business Rule versus Business Requirement**

**( Refer to:** [**https://www.ibm.com/cloud/learn/business-rules**](https://www.ibm.com/cloud/learn/business-rules) **)**

|  |
| --- |
| **Business rule:**  An organization's policies, practices, or limitations are defined by a statement or series of statements known as a business rule. These rules provide particular conditions or instructions to follow, which aid in decision-making and automating operations. Both recorded and undocumented information within the company may be used to create business rules. They are employed to make sure that various company processes are efficient, compliant, and consistent.  **Example:**  Implementing business rules can be used, for instance, to ensure that papers adhere to standards by obeying regulations such as those governing finance or healthcare. Financial assessments are utilized in banking and real estate to figure out who gets credits, for instance, rules might keep somebody from utilizing administrations on the off chance that they don't pay for them. Rules in retail can regulate things like return policy. Additionally, rules in marketing enable websites display different content to various users, such as popular items to new users and products you've already seen to returning visitors.  **Business requirement:**  A business requirement is an official statement of what a project or system must do to satisfy the organization's goals. Business requirements describe the functions, features, and results that should be achieved by a project or system. They act as the cornerstone for the conception, execution, and assessment of projects. Business requirements are essential for ensuring that stakeholders comprehend a project's scope and purpose and that it is in line with the objectives of the company.  Business requirements are high-level, comprehensive statements detailing the overall aims and functions of a project or system, whereas business rules are particular, conditional assertions that regulate decision-making and automation within current business processes. Business requirements serve as project foundations and are developed at the project's inception to align objectives and scope with the organization's goals. Business rules concentrate on operational advice and are implemented regularly. |

**(c) Chen Model versus Crow’s Foot Model**

|  |
| --- |
| **Chen model:**  A conceptual data modelling method used in database systems and information modelling is called the Chen Model. Based on Peter Chen's work, it was created in the 1970s and is sometimes referred to as the Entity-Relationship Diagram (ERD). This model depicts the relationships between the entities (i.e., the things or ideas about which data is kept) in a database.  **Crow’s Foot model:**  Another method for developing ERDs is known as the "Crow's Foot Model" because of the visual notation it employs, which looks like a crow's foot. In database design and documentation, it is a version of the Chen Model.  Database entity-relationship modelling strategies include the Chen Model and Crow's Foot Model. Chen Model used diamond shapes and labelled verbs to provide descriptive insights, whereas Crow's Foot Model uses crow's feet symbols and straight lines to indicate cardinality, which is thought to be more intuitive and is frequently used in practice. These visual representations of relationships are where they primarily diverge. Personal taste and project conventions determine which option to choose. |

**(d) ERD versus ERDM**

|  |
| --- |
| **ERD:**  ERD (Entity Relationship Diagram): An ERD is a diagram that is used to model a database's structure. Entities (similar to tables in a database), attributes (properties of entities), and relationships (connections between entities) are represented by graphical symbols. For instance, in an ERD for a apple farm database, you would have entities like "apple," "farmer," and "consumer," and you would demonstrate how they link to one another, such as "apples are grown by farmers" or "consumer buys apples." ERDs are user-friendly and aid designers in quickly comprehending the database structure.  **ERDM:**  The Entity Relationship Model (ERM) in data set engineering is important for a bigger idea known as ERDM. It indicates how qualities ought to be developed, how elements ought to cooperate with each other, and how information ought to be organized. ERDM serves as the guiding framework for the construction of ERDs rather than provide the visual representation itself. Consider it the manual or design philosophy for efficiently constructing databases.  In the field of database design, the ERD (Entity Relationship Diagram) and ERDM (Entity Relationship Data Model) play complimentary roles. ERDM offers the theoretical foundation and guiding principles for database design, whereas ERD is a specialized visual tool that visually illustrates a database's structure. In an example, ERDM is like the game's rules, outlining how data and entities should relate to one another, while ERD is the actual game board, illustrating the database structure in accordance with ERDM's regulations. Together, they provide efficient database design, with ERD providing a visual depiction of the design concepts given out by ERDM. |

**(e) Volume ( Big Data ) versus Velocity ( Big Data )**

|  |
| --- |
| **Volume (Big Data):**  The sheer number or quantities of data being created and stored within a system are referred to as volume in the context of big data. Along with Velocity and Variety, it is a crucial aspect of big data. The expression stresses how much information is currently being assembled and held by organizations because of contemporary innovation, the spread of the web, online entertainment, and different information producing apparatuses.  **Example:**  Any social media platform like Facebook, Instagram, Twitter, and so forth gathers a lot of information on normal premise which addresses gigantic volume of the data/information.  **Velocity (Big data):**  While examining huge information, the expression "velocity" refers to the speed or speed at which new information is delivered, enters a framework, and should be handled. It considers both the steady contribution of information and the direness with which it should be handled to be significant. Similar to Volume, Velocity is a core attribute of Big Data and poses difficulties for real-time data processing and analysis.  **Example:**  For instance, the online e-commerce platforms track each and every step of the customers such as searches, products added to card, comparisons, etc including the purchase. This continuous inflow of data requires quick processing and analysis in order to provide individualized suggestions and enhance the user experience.  While Velocity refers to the pace at which data is created and must be processed in real-time, Volume in Big Data refers to the sheer quantity of data, stressing its immensity. While Velocity deals with processing quickly flowing data streams from sources like IoT devices or social media, Volume works with building up big databases over time, such historical records. The essential characteristics of Big Data are both volume and velocity, with volume emphasizing the size of the data and velocity emphasizing the necessity of real-time data processing for useful insights. |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part 2 Completion: Fill the Blanks Exercises**

For each of these exercises, complete each sentence or phrase.

**(1)** A(n) schema is the conceptual organization of an entire database as viewed by a database administrator.

**(2)** A(n) Business rule is a brief, precise and unambiguous description of a policy, procedure, or principle within a specific organization.

**(3)** A(n) data model is a relatively simple representation of more complex   
 real - world data structures.

**(4)** Each row in the relational table is known as a(n) entity instance or entity occurrence.

**(5)** Unified Modeling Language (UML) is a language based on OO concepts that describes a set of diagrams and symbols used to graphically model a system.

**(6)** The term logical design is used to refer to the task of creating a conceptual data model that could be implemented in any DBMS.

**(7)** The external model is the end users’ view of the data environment.

**(8)** An internal schema refers to a specific representation of an internal model, using the database constructs supported by the chosen database.

**(9)** From a database point of view, the collection of data becomes meaningful only when it reflects properly defined business rules.

**(10)** The movement to find new and better ways to manage large amounts of

Web - and sensor - generated data and derive business insight from it, while simultaneously providing high performance and scalability at a reasonable cost is referred to as " Big Data".

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Part 3 Short Answer Exercises**

Consider the data schema that follows and then answer the questions pertaining to these tables of information. .

**[ Orders\_Details** **table ]**

A table with numbers and text

Description automatically generated

**(1)** Write an SQL query which will display the arithmetic mean value of all the entries in the **UnitPrice** column.

* Select avg(UnitPrice) From Orders\_Details

**(2)** Construct an SQL query which will determine all records, from the above table, which have a order quantity which is strictly between 20 and 30 .

* Select \* from Orders\_Details Where Quantity > 20 AND Quantity < 30

**(3)** Construct an SQL query whereby the count of all the entries from **OrderID**

number 10250 will be displayed.

* Select count(\*) From Orders\_Details Where OrderID = 10250;

**(4)** Write an SQL query of your choice whereby zero records will be displayed.

* Select \* From Orders\_Details Where ID = 29

**(5)** Write an SQL query which will display the total dollar value of the discounts that were taken.

* Select sum(Discount) as TotalDiscountedValue From Orders\_Details

**Part 4 Essays Exercises**

Write a brief but complete answer to each of these questions / exercises.

**(1)** Business rules describe the main and distinguishing characteristics of the data as viewed by the company. Business rules define the operational policies, procedures and constraints of an organization.

Consider an antique shop such as the one named below.

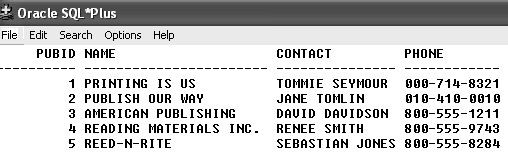
**Ashland Avenue Antiques**

List five business rules of your choice that you think could be implemented by such an enterprise.

|  |
| --- |
| The rules which can be implemented are as follows:  **Cost rule:** Set pricing on elements such market demand, historical relevance, rarity, and condition.  **Maintenance and repair rule**: Establish standards for item maintenance and repair to protect their historical value while guaranteeing they are in marketable condition.  **Security rule:** Implement security measures, like as alarm systems and camera systems, to guard expensive or rare antiques from theft and damage.  **Engagement rule:** Develop techniques for interacting with customers, such as presenting events, giving historical context on antiques, and providing  suggestions.  **Return and Refund rule:** Define the shop's return and refund policy, taking into mind the distinctive character of antique objects and any applicable regulatory restrictions. |

**(2)** Consider the **Publishers** table, shown below.

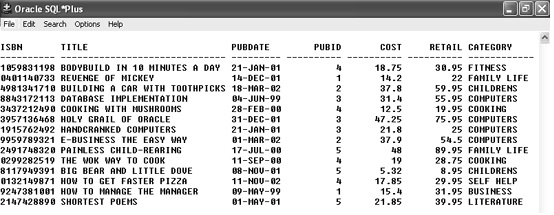
Using a **WHERE** clause in each statement, construct four separate SQL statements that will return specific results from the table that you think would be pertinent for information purposes.



|  |
| --- |
| 1. Select \* From Publishers Where PUBID > 1 and PUBID < 4 2. Select \* From Publishers Where Name = “AMERICAN PUBLISHING” 3. UPDATE Publishers Set Publishers.PUBID = “7” Where (Publishers>PUBID = 1) 4. Delete \* From Publishers Where CONTACT IN (Select Publishers.CONTACT From Publishers Where Publishers.CONTACT = “RENEE SMITH”) |

**(3) (a)** Consider the **Books** table, shown below.

Write an SQL statement that will display the profit generated by each book currently stored in the **Books** table?



**(3) (b)** Consider the **Orders** table, shown below.

List five other fields that you believe could / should be included within this **Orders** table to supplement it with additional attributes.

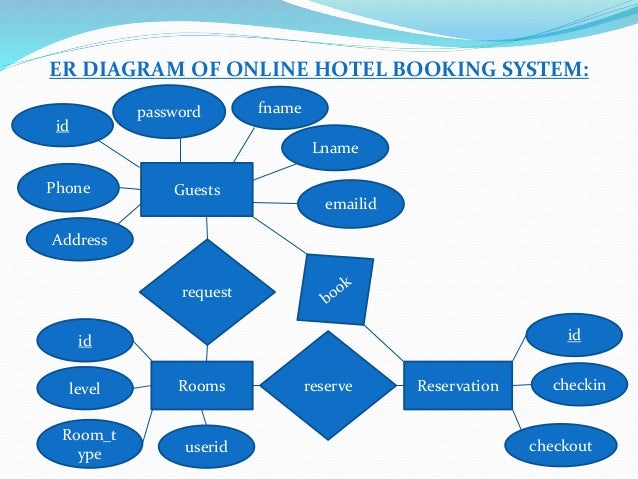


|  |
| --- |
| 1. Select ISBN,TITLE,(RETAIL-COST) AS PROFIT From Books 2. PAYMENT\_METHOD, ORDER\_VALUE, DISCOUNT, ORDER\_STATUS, EXPECTED\_DELIVERY |

**(4) ( Hotel Registration Data Model )**

This figure below illustrates a representation of the data storage for an application using a relational database model.

List ten observations that you consider important with respect to the typical elements that comprise such an ER diagram.



|  |
| --- |
| 1) 3 major entities room, reserve, guests  2) Aggregating the entities, when necessary, for instance, while representing a booking show.  3) Reservation and room entities.  4) Missing foreign key to link two entities such as customer id and reservation  5) Id attribute has an underline, which indicates that it is a key.  6) Repeated attribute name can cause issues.  7) Relationship is not mentioned through entities correctly.  8) Cardinality is not mentioned.  9) Constraints should be added, like we cannot book a room that is already booked.  10) Adding more/essential attributes to the Guest entity would be beneficial. |

**(5) ( Database Table Design )**

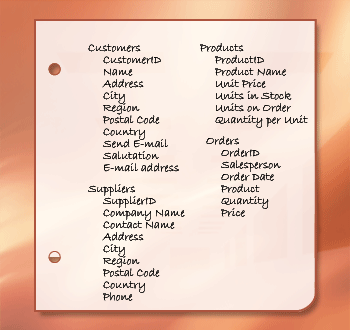
Review the Database Design Basics as discussed at the Web link below.

[**https://support.microsoft.com/en-us/office/database-design-basics-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5**](https://support.microsoft.com/en-us/office/database-design-basics-eb2159cf-1e30-401a-8084-bd4f9c9ca1f5)

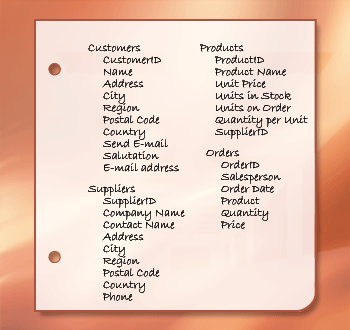
Then, examine the development of their **Product Sales Database Schema**, which is shown below, in three waves. Expound on your findings for the mentality of their development.

**[ Product Sales Database Schema ]**

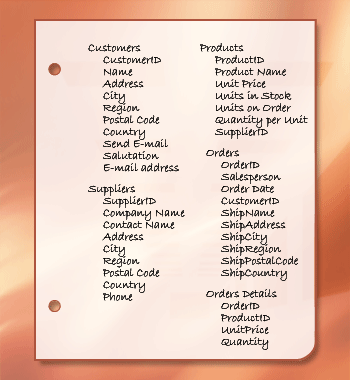
**[ First Wave ]**



**[ Second Wave ]**



**[ Third Wave ]**



|  |
| --- |
| **First wave:**  In the first phases of creating our database, we painstakingly collect and arrange data, removing duplicates to guarantee data quality. We then decide to concentrate our database architecture on four main tables: "Customers," "Products," "Suppliers," and "Orders." We follow the rule of just recording data once in order to minimize errors and conserve disk space, providing the foundation for an effective, scalable structure. The acquired data is then organized into columns inside the corresponding tables, for example, first and last names are separated in the "Customers" table, and the "Address" column is expanded to include the city, state, country, and ZIP code. We continue to be adaptable to different address formats for clients from other countries.  **Second wave:**  Finding Unique and Stable Columns: When building our core tables, it is essential to select columns that are both unique and stable to act as primary keys. These codes guarantee accurate data retrieval and data integrity. A primary key must be unique and immutable to avoid misunderstanding and data mistakes. It must be immutable in order to remain consistent across time for use as a reference in other tables. We can use the "autonumber" data type when a natural unique key, such as an employee or product ID, is not present. It automatically increases with every new record, ensuring uniqueness. Creating Stable Relationships: The establishment of these primary keys serves as the cornerstone for stable table relationships that guarantee data consistency. This improves database design, increasing the effectiveness of data retrieval and querying.  **Third wave:**  After building tables and specifying primary keys, the crucial following step entails merging data by forming relationships. For effective data storage and retrieval, this is essential. Table connections may be made simpler by using a relational database management system like Microsoft Access to do this. For easier management of their connections, we may, for instance, use Supplier ID as a foreign key to establish a one-to-many link between the Supplier and Product tables. A junction table is required in situations needing a many-to-many relationship, such as between Orders and Products. As a bridge, this intermediary table helps to retain data accuracy while facilitating complicated interactions. Extensive data validation checks are essential following the definition of tables, fields, and connections. Effective database management requires keeping accurate data in all columns and removing duplicates in order to ensure data consistency and dependability. |