**Homework 4: ERD and Normalization (Total Points: 100)**

Due: Saturday March 12, 11:59PM ET

**Note: Functional Dependencies and Normalization will be covered in Lecture 6**

**Problem 1: Design (Points: 30)**

A school tracks students in its database as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| StudentId | FirstName | LastName | Last 4 of SSN | Email | Degree | Major |
| 123456 | Mary | Smith | 1111 | smithm@school.edu | BS | Business |
| 123457 | Jamie | Jameson | 2222 | jameson@school.edu | BA | Criminal Justice |
| 123458 | John | Jones | 3333 | jones@school.edu | BS | Business |

a. Some students wish to pursue a minor. What changes would the school need to make to its database to keep track of students who pursue a major and a minor? You can enter a descriptive explanation with **conceptual** diagrams to support your proposal (15 points):

|  |
| --- |
| A student can pursue a minor and a major. And a major can be pursued by many students.  A minor can be pursued by many students. So use major entity to represent the majors of students, and use minor entity to represent the minors of students. The relationship between major and student is one to many, and the relationship between minor to student is one to many. |

b. Some students wish to pursue double majors. What changes would the school need to make to the database to keep track of students who pursue double majors? You can enter a descriptive explanation with **conceptual** diagrams to support your proposal (15 points):

|  |
| --- |
| A student can pursue double or many majors and a major can also be pursued by many students. So use major entity to represent the majors of students. The relationship between student and major is many to many. |

**Problem 2: Entity Relationship Diagram (Points: 30)**

Karen has been making fashion jewelry for a few years now. She’s been so successful that she wants to take her passion to the Web. With this in mind, she founded KewlJewels.

KewlJewels sells different types of jewelry pieces (e.g., bracelets, pendants). Each jewelry piece has a specific description (e.g., white pearl earrings). She wants to be able to produce sales reports that indicate how many pieces per type the company has sold. Potential customers may browse the online catalog without registering on the site.

While browsing the catalog, they may add jewelry pieces to a virtual shopping cart. Once they select to go to the checkout, they will need to provide personal information : name, phone number, address, email. Only credit card payments are accepted. The site collects the credit card type and number, and expiration date.

Customers may choose a shipping address different from the billing address, and the database needs to keep track of both addresses. Only complete orders are shipped, no partials are allowed. The database does not track inventory control, only sales information. You are asked to design the database to support the Web operations

a. Create some sample data which meets the requirements identified in the description above. Use a table format similar to slide 46 of lecture 5. (10 points)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | OrderId | JewelryId | JewelryType | JewelryDesc | OrderDate | SoldPiece | CustID | Name | Phone | Address | Email | ShippingAddress | CreditType | CreditNumber | CreditExpiration | | O001 | J1 | bracelet | white pearl bracelet | 2022-03-01 | 2 | C01 | Aillen Gu | 11203654 | New York, XXX street 01 | [ailen@gmail.com](mailto:ailen@gmail.com) | New York, XXX street 07 | T01 | 123456789 | 2023-03-01 | | O001 | J2 | pendant | gold pearl pendant | 2022-03-01 | 1 | C01 | Aillen Gu | 11203654 | New York, XXX street 01 | [ailen@gmail.com](mailto:ailen@gmail.com) | New York, XXX street 07 | T01 | 123456789 | 2023-03-01 | | O002 | J3 | earring | white pearl earrings | 2022-03-02 | 1 | C03 | Susan Smith | 11023652 | New York, XXX street 02 | [susan@gmail.com](mailto:susan@gmail.com) | New York, XXX street 02 | T01 | 2345678912 | 2024-03-02 | | O003 | J1 | bracelet | white pearl bracelet | 2022-03-04 | 2 | C04 | Lucy Ross | 11452362 | New York, XXX street 03 | [lucy@gmail.com](mailto:lucy@gmail.com) | New York, XXX street 03 | T02 | 345678912 | 2026-03-04 | | O004 | J2 | pendant | gold pearl pendant | 2022-03-05 | 1 | C05 | Linsay Joe | 10123556 | New York, XXX street 04 | [linsay@gmail.com](mailto:linsay@gmail.com) | New York, XXX street 04 | T03 | 456789123 | 2023-03-05 | | O005 | J3 | earring | white pearl earrings | 2022-03-08 | 2 | C06 | Alfie Wang | 10254103 | New York, XXX street 05 | [alfie@gmail.com](mailto:alfie@gmail.com) | New York, XXX street 05 | T02 | 567891234 | 2025-03-08 | | O006 | J4 | ring | gold ring | 2022-03-08 | 1 | C07 | Luis Lee | 10241253 | New York, XXX street 06 | [luis@gmail.com](mailto:luis@gmail.com) | New York, XXX street 06 | T03 | 456123789 | 2023-06-29 | |

b) Create a **physical** ERD (identify entities, relationships, attributes, primary and foreign keys) using Crow’s foot notation. (20 points)

|  |
| --- |
|  |

**Problem 3: Database Keys (Points: 20)**

During the normalization process, you end up with the following candidate keys:

Key 1: InvoiceNo, InvoiceDate  
Key 2: InvoiceNo, SalespersonID  
Key 3: InvoiceNo, SalespersonEmail  
Key 4: InvoiceNo, SalespersonSSN

Assuming each invoice can be handled by more than one salesperson, which pair of the fields would you choose as the Primary Key? Explain the reasons for discarding the options as appropriate:

|  |
| --- |
| Key 1: I will not choose Key 1 as the primary key.  Key 2: I will choose Key 2 as the primary key.  Key 3: I will not choose Key 3 as the primary key.  Key 4: I will not choose Key 3 as the primary key.  Reason: each salesperson has an unique ID, Email and SSN. An email and ssn may need varying characters to represent but ID can use only number to represent. And invoice date need use “date” type to represent. Use numbers can reduce errors and can easily maintain consistency. |

**Problem 4: Functional Dependencies (Points: 20)**

Given the following four functional dependencies:

**FD-1:** **GuestID** --> FirstName, LastName, Email

**FD-2:** **GuestID, PhoneID** --> FirstName, LastName, Email, PhoneNo, PhoneType

**FD-3:** **EventID** --> EventDate, EventDescription

**FD-4:** **RSVPNo** --> GuestID, FirstName, LastName, Email, RSVPDate, NoOfGuests, EventID, EventDate, EventDescription

For each FD, indicate whether it is a FD or not. If any of the above are not, what changes would you need to make to turn them into full functional dependencies?

|  |
| --- |
| FD-1, Fd-3 and FD-4 are FDs.  FD-2 need to change to PhoneID-> GuestID, FirstName, LastName, Email, PhoneNo, PhoneType . |

**BONUS: BASIC NORMALIZATION PROCESS (10 points)**

The following is an inventory report from your new customer, a small plastic container reseller:

Pharmacy Inventory Report

Product Code: 01 Product Type: Medicine Manager: Darlene Snell

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item ID** | **Item Name** | **Category** | **Quantity on Hand** | **Last Physical Count** | **Re-order when inventory falls below** |
| 101  102  103 | Aspirin  Claritin  Ibuprofen | pain  allergy  pain | 120  400  300 | 1/30/2022  1/30/2022  2/15/2022 | 30  30  50 |

Product Code: 02 Product Type: Hygiene Manager: Wendy Byrde

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Item ID** | **Item Name** | **Category** | **Quantity on Hand** | **Last Physical Count** | **Re-order when inventory falls below** |
| 101  102  200 | Hand Soap  Shampoo  Facial Tissue | Bath  Bath  Body Care | 100  100  500 | 1/30/2022  1/30/2022  2/20/2022 | 20  20  100 |

1. Normalize the data in First Normal Form. Identify Functional Dependencies, Candidate Keys and Primary Keys.
2. Normalize the data into Second Normal Form.
3. Normalize the data into Third Normal Form.
4. Normalize the data in First Normal Form. Identify Functional Dependencies, Candidate Keys and Primary Keys.

Product Code, Item ID🡪 Product Type, Manager, Item Name, Category, Quantity on Hand, Last Physical Count, Re-order when inventory falls below

PharmacyInventory(Product Code, Product Type, Manager, Item ID, Item Name, Category, Quantity on Hand, Last Physical Count, Re-order when inventory falls below)

(2) Normalize the data into Second Normal Form.

Product Code 🡪 Product Type, Manager

Product Code, Item ID🡪 Item Name, Category, Quantity on Hand, Last Physical Count, Re-order when inventory falls below

Removing partial dependency:

Product (Product Code , Product Type, Manager)

PharmacyInventory(Product Code, Item ID, Item Name, Category, Quantity on Hand, Last Physical Count, Re-order when inventory falls below)

1. Normalize the data into Third Normal Form.

There’s transitive dependency:

Product Code, Item ID🡪 Item Name

Item Name 🡪 Category

So remove the transitive dependency:

Product (Product Code , Product Type, Manager)

Category (Item Name, Category)

PharmacyInventory(Product Code, Item ID, Item Name, Quantity on Hand, Last Physical Count, Re-order when inventory falls below)