Name	CWID

Homework Assignment 3

November 9th, 2017 Due on November 23th, 11:59pm (midnight)

CS425 - Database Organization Results

Please leave this empty!

3.1	3.2	3.3	Sum

Instructions

- Try to answer all the questions using what you have learned in class
- Some questions are marked as bonus. You do not have to answer these questions to get full points for the assignment. However, you can get bonus points for these questions!

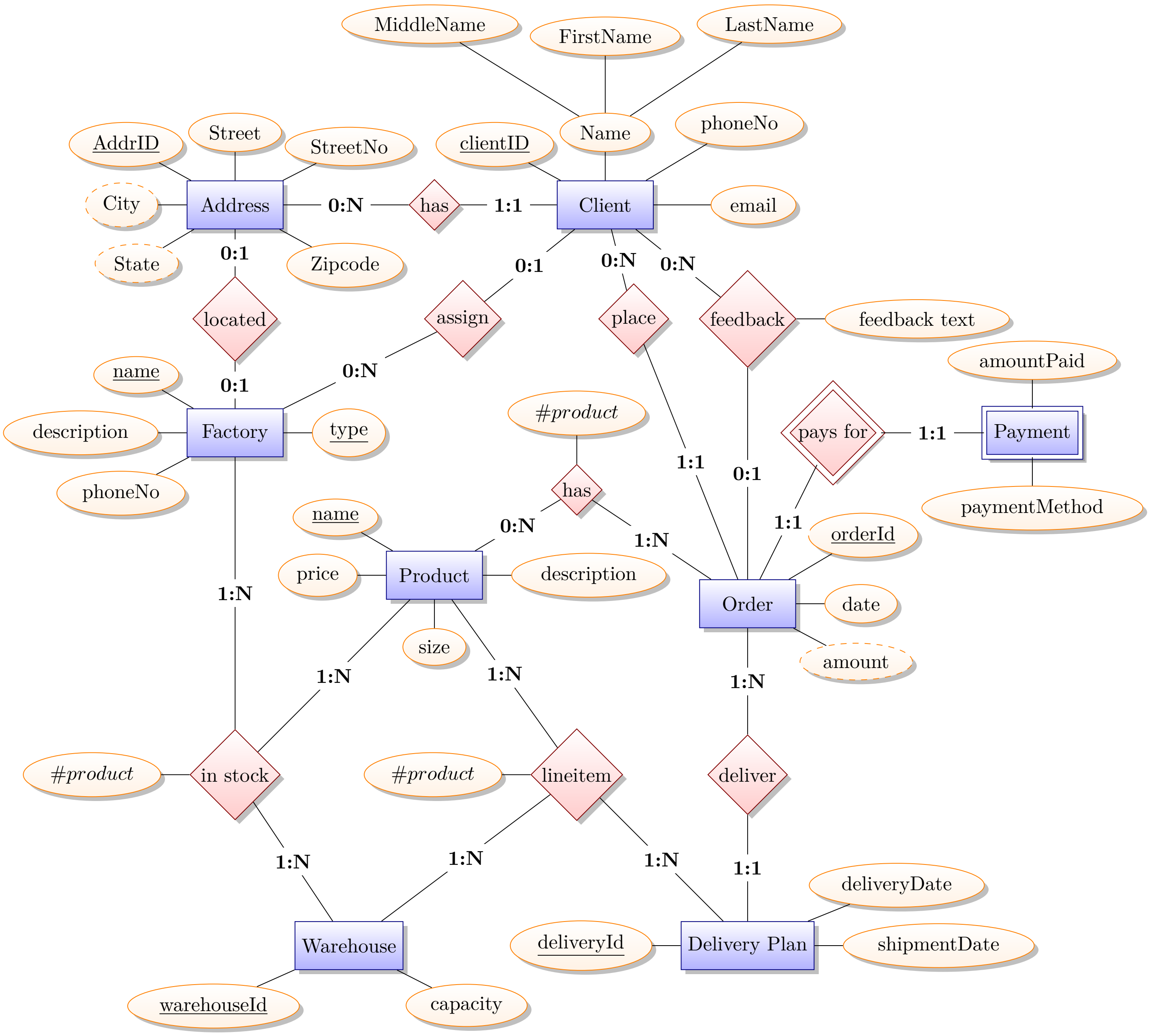
Part 3.1 Modelling (Total: 60 Points)

Question 3.1.1 (60 Points)

Build a conceptional model for a **Factory Management System**. The solution should be presented as an **ER-diagram**. Base your design on the following requirements.

- The database should record information about Clients, Factories, Addresses, Products, Orders, WareHouses, Delivery Plans, and Payments.
- A **Client** has a name which consists of *firstName*, *middleName* and *lastName*. **Clients** are identified by a unique *clientID*. A **Client** has an *Address*, *phoneNo* and *email* address. A **Client** may be assigned to a *Factory*.
  - **Clients** can place any number of *Orders* (including none). **Client** may provide *Feedback* for each order they have placed (optional). For every order, a **Client** has to make a *Payment*.
- A **Factory** is identified by the combination of *name* and *type*. A **Factory** may have one *Address*. A **Factory** has a type, phone number and a short description.
- An **Address** consists of a unique *addrID*, *street*, *streetNumber*, *city*, *state* and *zipcode*. The attributes *city* and *state* can be derived from the attribute *zipcode*.
  - There may be some **Addresses** which are not be associated with any *Client* or *Factory*.
- A **Product** is identified by its *name*. A **Product** has a *price* and *size*, and a *description*.
  - **Products** are stored in warehouses. For each **Product** stored in a *Warehouse* we record the number of *Products* in stock produced by a certain *Factory* (e.g., 3 Macbooks in Warehouse 1 were produced by Factory 1).
- A **Warehouse** is identified by its *warehouseId*. A **Warehouse** has a *capacity*.
  - A **Warehouse** may service multiple *Factories*.
  - There may be one or more **Warehouses** for a *Factory*.
- An **Order** is uniquely identified by the *orderId*. An **Order** is created by a *Client*. For each order we store a *Total Amount* of ordered items and a *Date*.
  - An **Order** is associated with one or more *Products*. For each *Product* in an **Order**, we have to record how many items of this product are ordered (e.g., Order 1 contains 3 Spoons and 15 Forks). Note that the *Total Amount* can be computed as the sum of number of items for each ordered *Product*.
- A **Delivery Plan** is uniquely identified by the *deliveryId*. For a **Delivery Plan** we record a *Shipment Date* and *Delivery Date*.
  - Each *Order* is associated with one or more *Delivery Plans*. Each *Delivery Plan* is associated with exactly one *Order*.
  - A *Delivery Plan* consists of multiple lineitems. A lineitems encodes the shipment of a number of items of one *Product* which reside in one *Warehouse*. For example, consider an order of 10 Spoons. One possible way to service this order is to create 2 delivery plans: the first delivery plan ships 3 Spoons from Warehouse 1 and 2 Spoons from Warehouse 2 while the second delivery plan ships 5 Spoons from Warehouse 3.
- A **Payment** is identified by the *Order* for which the payment was made. It consists of the *amountPaid* and *paymentMethod* (Credit Card, E-Check, etc.)

Solution

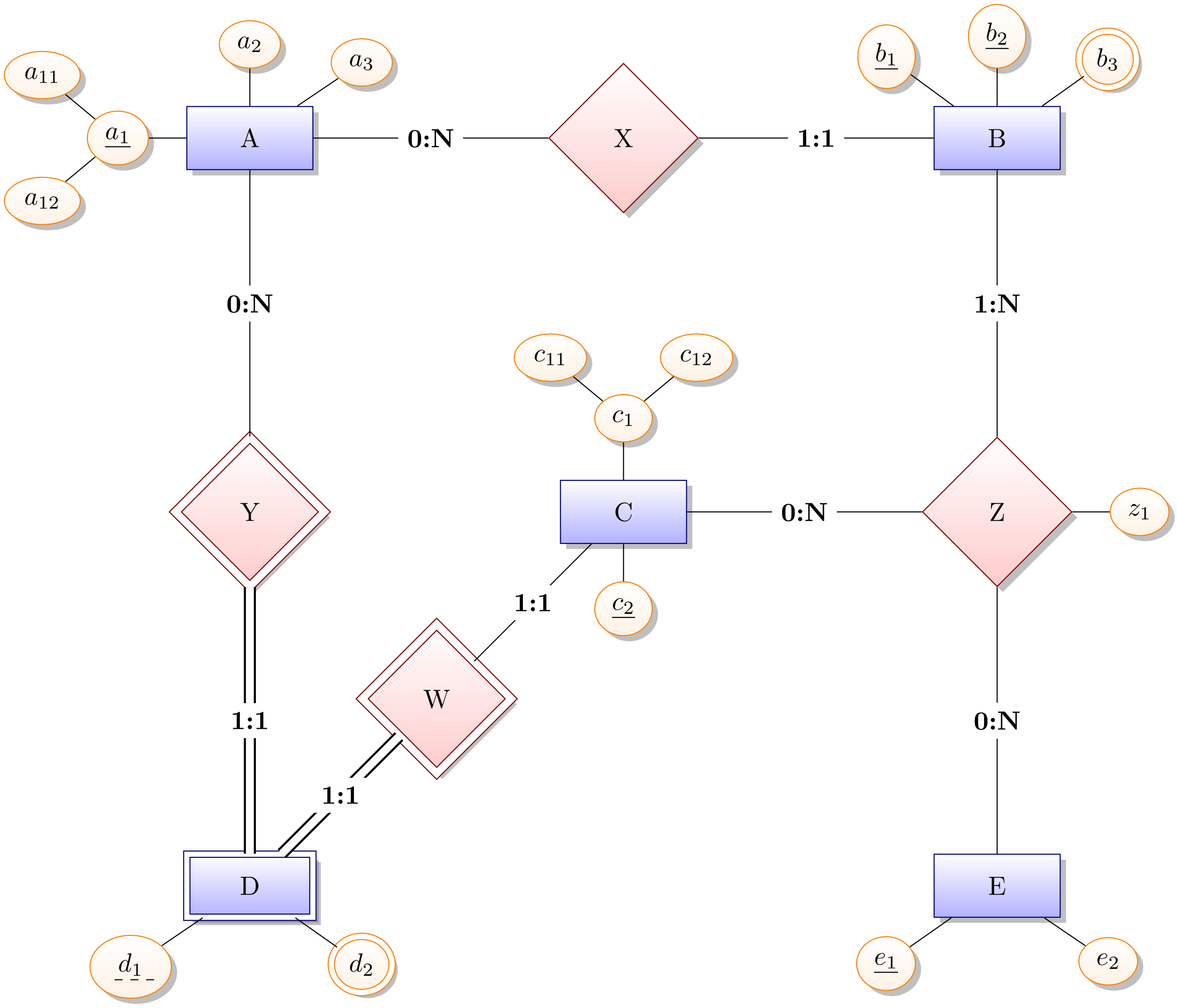

MiddleName
FirstName
LastName
Name
phoneNo
AddrID
Street
StreetNo
clientID
City
Address
0:N
has
1:1
Client
email
State
0:1
Zipcode
0:1
0:N
0:N
located
assign
place
feedback
feedback text
name
0:N
0:1
amountPaid
description
Factory
type
#product
pays for
1:1
Payment
phoneNo
has
1:1
0:1
name
0:N
1:1
paymentMethod
1:N
orderId
price
Product
description
Order
1:N
date
size
amount
1:N
1:N
1:N
#product
in stock
#product
lineitem
deliver
1:N
1:N
1:N
1:1
deliveryDate
Warehouse
deliveryId
Delivery Plan
shipmentDate
warehouseId
capacity

Part 3.2 Translation of ER into Relational Model (Total: 40 + 10 BONUS Points)

Question 3.2.1 (40 Points)

Take the following ER-model and translate it into a relational schema using the rules presented in class. Present the relational schema as an SQL script (assume that all attributes are of data type INT). Present the results of the following intermediate steps in this order:

1. Translate strong entities + unnest composite attributes
2. Translate weak entities
3. Translated multi-valued attributes
4. Translate relationships



Solution

1st Step (strong entities)

```
CREATE TABLE A (
   a11 INT,
   a12 INT,
   a2 INT,
   a3 INT,
   PRIMARY KEY(a11, a12)
);

CREATE TABLE B (
   b1 INT,
   b2 INT,
   PRIMARY KEY (b1,b2)
);

CREATE TABLE C (
   c2 INT PRIMARY KEY,
   c11 INT,
   c12 INT
);

CREATE TABLE E (
   e1 INT PRIMARY KEY,
   e2 INT
);
```

2nd Step (weak entities)

```
CREATE TABLE A (
   a11 INT,
   a12 INT,
   a2 INT,
   a3 INT,
   PRIMARY KEY(a11, a12)
);

CREATE TABLE B (
   b1 INT,
   b2 INT,
   PRIMARY KEY (b1, b2)
);

CREATE TABLE C (
   c2 INT PRIMARY KEY,
   c11 INT,
   c12 INT
);

CREATE TABLE D (
   a11 INT,
   a12 INT,
   c2 INT,
   d1 INT,
   PRIMARY KEY (a11, a12, c2, d1),
   FOREIGN KEY a11, a12 REFERENCES A,
   FOREIGN KEY c2 REFERENCES C
);

CREATE TABLE E (
   e1 INT PRIMARY KEY,
   e2 INT
);
```

3rd Step (multivalued attributes)

```
CREATE TABLE A (
   a11 INT,
   a12 INT,
   a2 INT,
   a3 INT,
   PRIMARY KEY(a11, a12)
);

CREATE TABLE B (
   b1 INT,
   b2 INT,
   PRIMARY KEY (b1, b2)
);

CREATE TABLE B3 (
   b1 INT,
   b2 INT,
   b3 INT,
   FOREIGN KEY (b1, b2) REFERENCES B,
   PRIMARY KEY (b1, b2, b3)
);

CREATE TABLE C (
   c2 INT PRIMARY KEY,
   c11 INT,
   c12 INT
);

CREATE TABLE D (
   a11 INT,
   a12 INT,
   c2 INT,
   d1 INT,
   PRIMARY KEY (a11, a12, c2, d1),
   FOREIGN KEY a11, a12 REFERENCES A,
   FOREIGN KEY c2 REFERENCES C
);

CREATE TABLE D2 (
   a11 INT,
   a12 INT,
   c2 INT,
   d1 INT,
   d2 INT,
   PRIMARY KEY (a11, a12, c2, d1, d2),
   FOREIGN KEY a11, a12, c2, d1 REFERENCES D
);

CREATE TABLE E (
   e1 INT PRIMARY KEY,
   e2 INT
);
```

4th Step (relationships)

```
CREATE TABLE A (
   a11 INT,
   a12 INT,
   a2 INT,
   a3 INT,
   PRIMARY KEY(a11, a12)
);

CREATE TABLE B (
   b1 INT,
   b2 INT,
   a11 INT,
   a12 INT,
   PRIMARY KEY (b1, b2),
   FOREIGN KEY (a11, a12) REFERENCES A,
);

CREATE TABLE B3 (
   b1 INT,
   b2 INT,
   b3 INT,
   FOREIGN KEY (b1, b2) REFERENCES B,
   PRIMARY KEY (b1, b2, b3)
);

CREATE TABLE C (
   c2 INT PRIMARY KEY,
   c11 INT,
   c12 INT
);

CREATE TABLE D (
   a11 INT,
   a12 INT,
   c2 INT,
   d1 INT,
   PRIMARY KEY (a11, a12, c2, d1),
   FOREIGN KEY a11, a12 REFERENCES A,
   FOREIGN KEY c2 REFERENCES C
);

CREATE TABLE D2 (
   a11 INT,
   a12 INT,
   c2 INT,
   d1 INT,
   d2 INT,
   PRIMARY KEY (a11, a12, c2, d1, d2),
   FOREIGN KEY a11, a12, c2, d1 REFERENCES D
);


CREATE TABLE E (
   e1 INT PRIMARY KEY,
   e2 INT
);

CREATE TABLE Z (
   b1 INT,
   b2 INT,
   c2 INT,
   e1 INT,
   z1 INT,
   PRIMARY KEY (b1, b2, c2, e1),
   FOREIGN KEY (b1, b2) REFERENCES B,
   FOREIGN KEY (c2) REFERENCES C,
   FOREIGN KEY (e1) REFERENCES E
);
```

Question 3.2.2 (10 (BONUS) Points)

Consider the following relations and for each determine in which normal form the relation is (note that a relation can be in multiple normal forms). Please consider the following normal forms: 1NF, 2NF, 3NF and BCNF.

1. $R(A, B, C, D)$ and the Functional Dependencies are $C \rightarrow B, B \rightarrow D, A \rightarrow BD$
2. $R(A, B, C, D, E)$ and the Functional Dependencies are $A \rightarrow B, B \rightarrow C, AC \rightarrow D$
3. $R(A, B, C, D, E)$ and the Functional Dependencies are $A \rightarrow BC, D \rightarrow E, AB \rightarrow CD$
4. $R(A, B, C, D, E)$ and the Functional Dependencies are $A \rightarrow C, B \rightarrow C, AB \rightarrow DE, BC \rightarrow AD$

Solution

1. The relation is in 1NF.
2. The relation is in 1NF.
3. The relation is in 1NF and 2NF.
4. The relation is in 1NF and 2NF.