

28th Oct. 2023

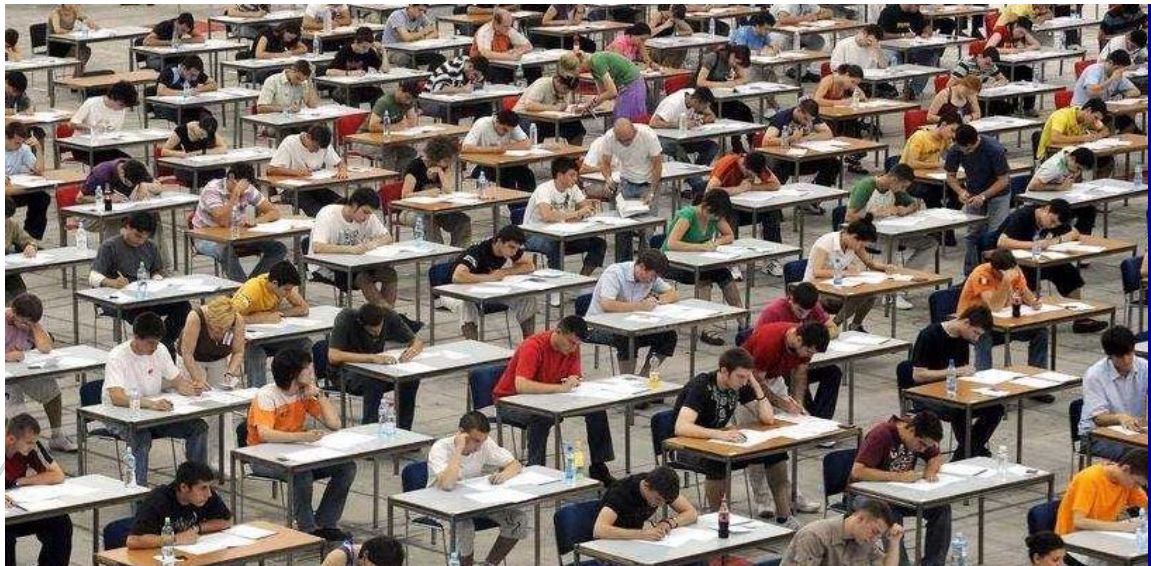


NYU

TANDON SCHOOL
OF ENGINEERING

Midterm Exam

CS-GY 6083 - B, Fall 2023 Principles of Database Systems



MIDTERM EXAM [100 points with 20% weight]

10/28/2023 11:00 AM to 1:00 PM EST: Time Limit: 2 Hours

Please read instructions carefully before writing exam

Write your name and net id below

- Last Name: Jie First Name: Cheng Net ID: jc12300

This is an online exam. Please login to zoom meeting using your net id (mandatory, do not login with your personal email account). Login to Brightspace course website to join the zoom meeting.

- Join to zoom meeting no later than 10:45am. Please keep video on and mute microphone during exam.
- Write your answers under each question in this word document and submit it in single PDF file before the deadline to Brightspace course site. The file should be in Netiid_FA23_Midterm.PDF format (e.g., asp13_FA23_Midterm.PDF). You may resubmit your assignment multiple times before the submission deadline. The latest submission will be considered for the grading.
- This exam has four sections A, B, C. All sections and questions have grading points. There are no negative points for any wrong answers.
- If you have any question during the exam, please send your question privately to professor/TA on zoom meeting chat window. Please do not speak in microphone.
- Plagiarism of any type will not be graded.

GOOD LUCK!

A) Answer following questions briefly [20 points]

- i) **Explain importance of referential integrity in OLTP database. Explain How to implement referential integrity with an example.**

Referential integrity ensures that relationships between tables are maintained, and that data remains consistent and accurate. It prevents situations where, for example, a foreign key in one table points to a non-existent record in another table. By upholding referential integrity, you can ensure data quality by averting the presence of orphaned records. Orphaned records refer to entries in child tables that lack matching parent records, resulting in incomplete and erroneous data. It also helps to prevent errors that might arise from inadvertent data manipulation, such as inserting or updating records with invalid references.

For example, we can create a parent table called Customers where CustomerID is defined as the primary key, ensuring that each customer has a unique identifier.

```
CREATE TABLE Customers (  
    CustomerID INT PRIMARY KEY,  
    FName VARCHAR(50),  
    LName VARCHAR(50),  
    Email VARCHAR(100)  
);
```

Then another child table called Orders can be created.

```
CREATE TABLE Orders (  
    OrderID INT PRIMARY KEY,  
    CustomerID INT,  
    OrderDate DATE,  
    TotalAmount DECIMAL(10, 2),  
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)  
);
```

In the Orders table, we define a foreign key constraint on the CustomerID column, referencing the Customers table. This ensures that any value in the CustomerID column in the Orders table must match an existing CustomerID in the Customers table.

ii) What is good quality data and its importance in business. Explain how to implement consistency in data with an example.

Good quality data refers to data that is accurate, complete, reliable, relevant, and consistent. It is free from errors, duplicates, and inconsistencies, making it an asset for businesses.

Importance in business:

1. Clean and accurate customer data allows businesses to better understand and engage with their customers, leading to improved customer satisfaction and loyalty.
2. Good data quality reduces errors and inefficiencies in business processes. This leads to improved operational efficiency.
3. Businesses with clean, accurate, and up-to-date data have a competitive advantage. They can respond more effectively to market changes and customer needs.
4. Many industries have strict regulations regarding data accuracy and privacy. Good data quality helps businesses comply with these regulations and mitigate risks associated with data breaches or non-compliance.

For example, in the relationship between product and category, we have a Product table and a Category table. The Product table has a column called CategoryID which refers to the product category in the Category table.

First, we ensure that the Category table follows a consistent data model. Each category should have a unique identifier (e.g., CategoryID), a name, and a description.

Then, in the Product table, enforce data consistency by making sure that the CategoryID column only contains values that exist in the Category table. We can do this by setting a foreign key constraint. With this foreign key constraint, we ensure that any value entered in the Product table's CategoryID column corresponds to a valid category in the Category table.

When adding or updating products, users are required to select a valid category from the predefined list in the Category table. This enforces data consistency and prevents the use of arbitrary or inconsistent category values.

B) Draw ERD (Logical & Relational model) [50 points]

Acme Inc. is a national retail chain, specializing in consumer electronics products. Historically, the company has grown through acquisitions. This had led to a high level of heterogeneity in the IT systems at Acme. The business executives find it difficult to get timely, relevant, accurate information for better business decision making. Also, the information does not have uniform formats and integrity. To

overcome this issue, the IT department has suggested creating a relational database that resolves the issues of data consistency and integrity.

The business team has provided a sample invoice to the database designer as a source to create a relational data model.

Business team of Acme has stated following requirements and rules:

- a) There could be multiple customers residing on same address.
- b) An order can have different address for billing and shipping.
- c) Data model needs to have employee name, email address, and phone number.
- d) Data model needs store address, and phone number.
- e) Unit price in invoice is the sales price and not the MSRP (Maximum Sale Retail Price).
- f) No separate entity is needed for promotion codes.

INVOICE

Acme Inc
345 Victoria Street
Kolkata 702 207

Bill To Address:

Mr. Vaghela, Shankar
Mint street, Kolkata

Ship To Address:

Mr. Vaghela, Shankar
Mint street, Kolkata

Invoice No I10005
Store No S303
Emp Id E777

Order No O123456
Order Date 06-Sep-14
Promo code None

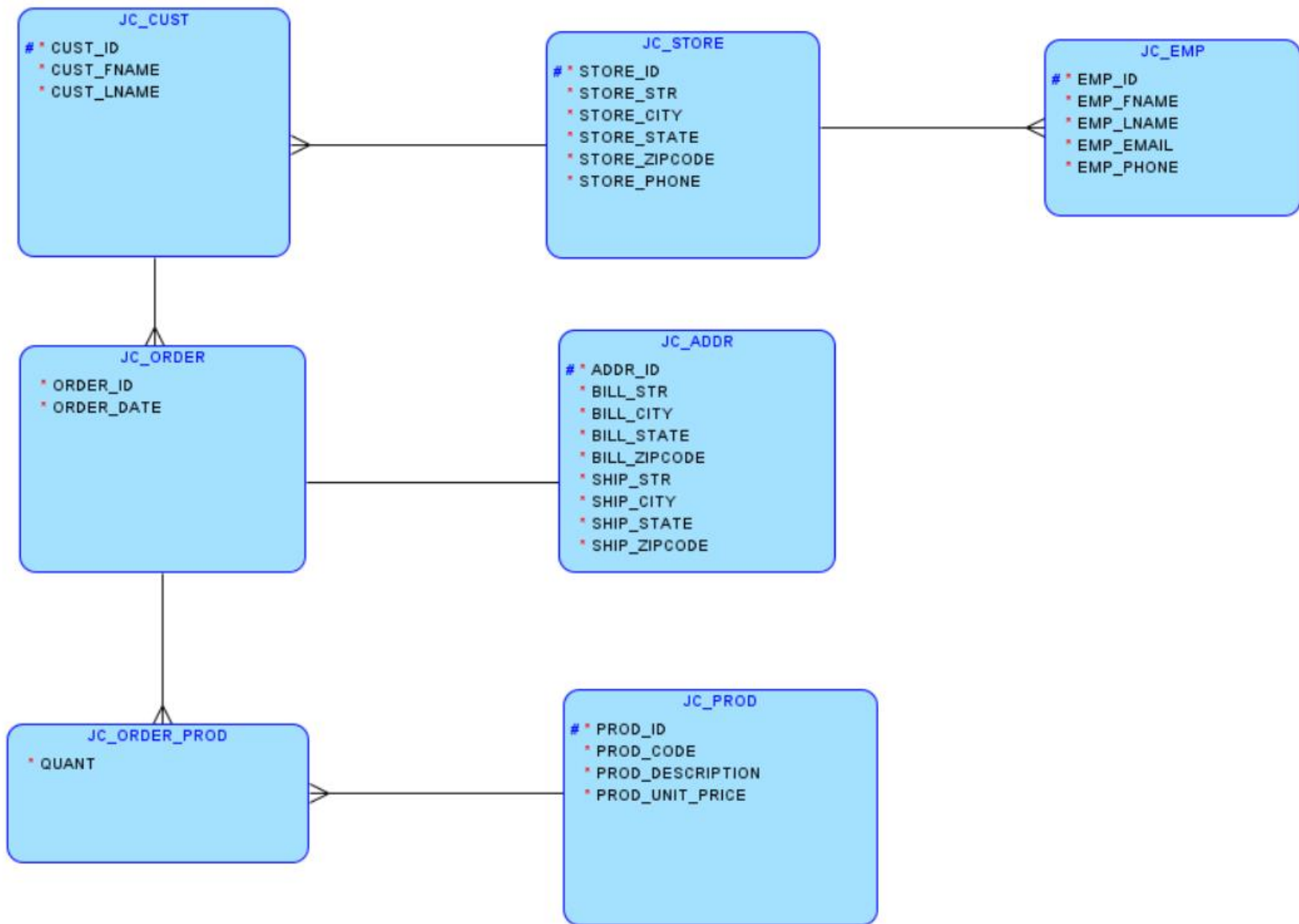
Serial no.	Product Code	Product Description	Qty	Unit Price	Amount
1	TV1001	24" LCD TV	1	35000	35000
2	W1002	Extended warranty	1	3000	3000
3					
4					
5					

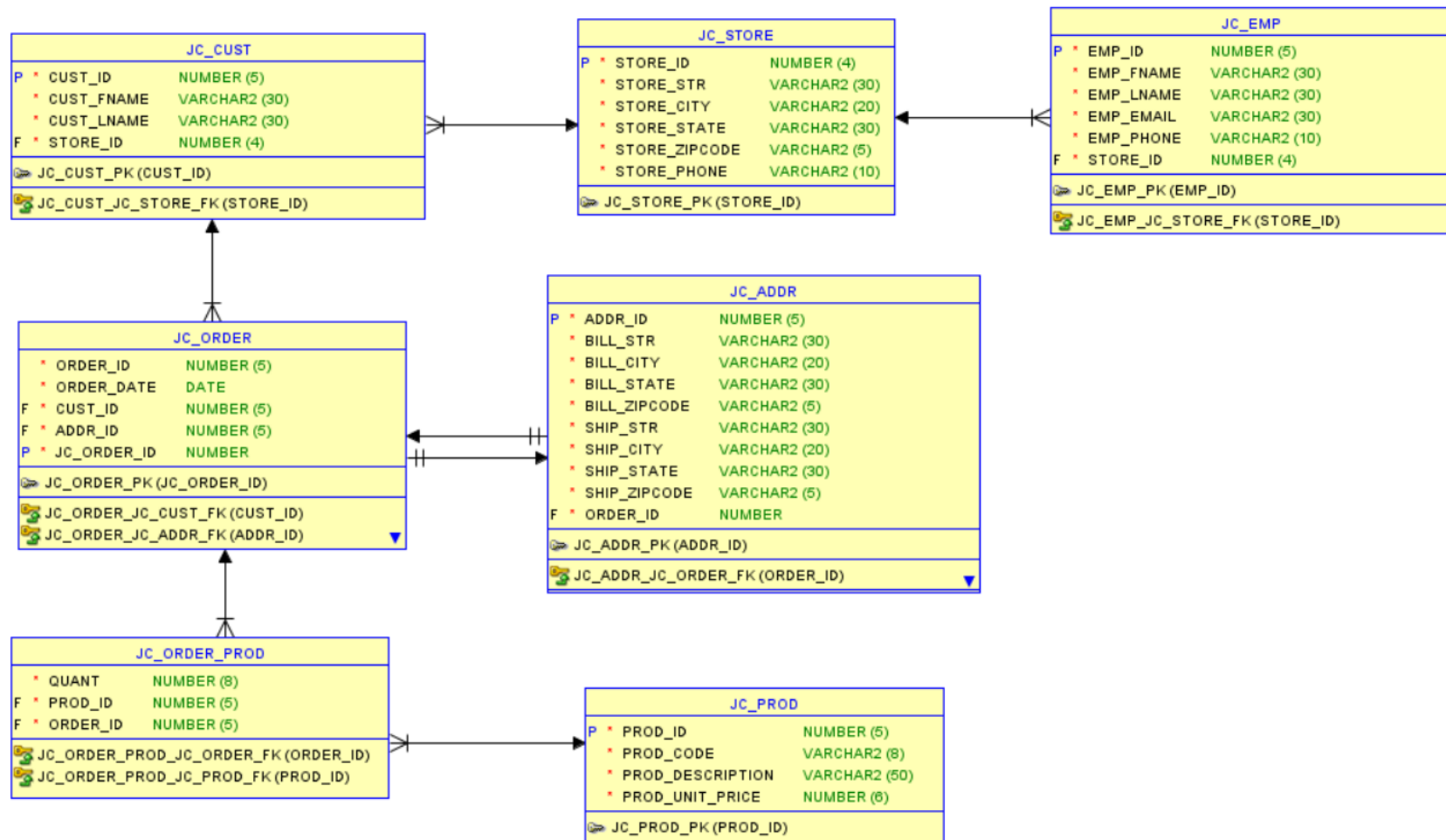
Sales Amount 38000
Sales Tax 4750
Freight 300
Discount Nil
Total Amount 43050

Create a normalized data model that represents all data items in the provided invoice. All table in your database model must have your initial as prefix, e.g., AP_EMP. **Ignore blue arrow lines in invoice picture.**

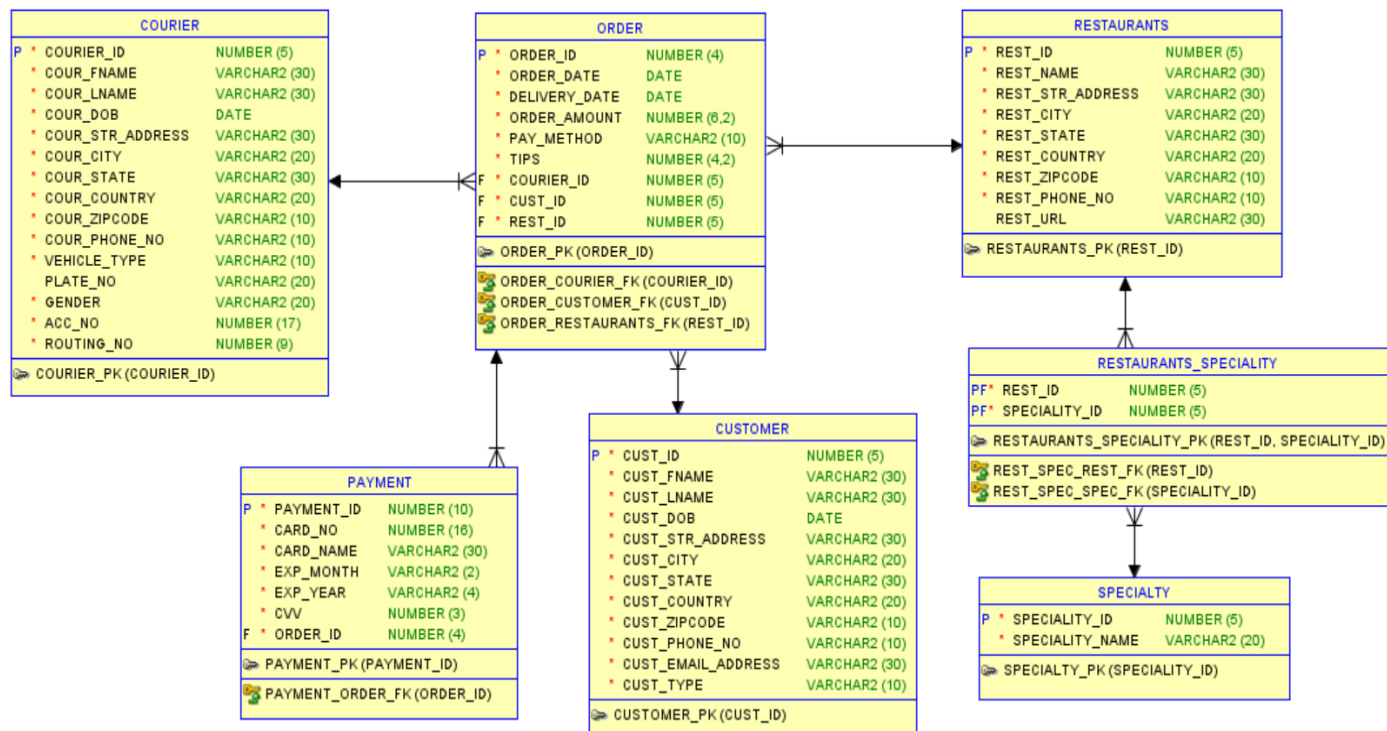
Submit: Screenshots of logical model and relational model, and any valid assumptions made other than stated.

NO NEED TO submit DDL code.





C) SQL DDL [30 points]



Answer the following questions in terms of writing SQL DDL code.

a) Apply a business rule that CVV number in Payment table has minimum 3 digits.

ALTER TABLE Payment

ADD CONSTRAINT CVV_Length

CHECK (LEN(CVV) >= 3);

b) Apply business rule that ORDER_DATE in ORDER table is not the past or future date.

```
ALTER TABLE Orders
```

```
ADD CONSTRAINT Order_Date
```

```
CHECK (ORDER_DATE >= CURRENT_DATE AND ORDER_DATE <= CURRENT_DATE);
```

c) Apply business rule that DOB of Courier is over 18 years of age.

```
ALTER TABLE Courier
```

```
ADD CONSTRAINT Courier_Age
```

```
CHECK (DATE_DIFF(CURRENT_DATE, DOB, YEAR) >= 18);
```

d) Identify surrogate key in this relational model and explain reason to have it as surrogate key.

```
CUSTOMER: CUST_ID
```

```
PAYMENT: PAYMENT_ID
```

```
COURIER: COURIER_ID
```

```
ORDER: ORDER_ID
```

```
RESTAURANTS: REST_ID
```

```
SPECIALTY: SPECIALTY_ID
```

The reason to have a surrogate key is to provide a stable, unique identifier for each record, even if the natural key (e.g., a customer's name or email) changes. Surrogate keys simplify database operations and improve performance in certain scenarios.

e) Explain how you can simplify this design. If any new entity needs to be added, what is it, and state relationships of that entity to other entities.

We can add a new entity called Delivery.

It has:

DELIVERY_ID (Primary Key)

ORDER_ID (Foreign Key)

Delivery_Date

It has one-to-one relationship with Order since each delivery is associated with one order.

Submit: Answer to each of the questions. Please do not remove any questions, write your answers underneath each question.

-----End of Exam Paper -----