

S3871491

Channon Harper

Database Concepts

Assessment 3

Although this may not be my best work I have been overly restricted in time the last 4 weeks and have done the best I could with the time I had I know some will be wrong I will look over material after the course ends when I have time however does not help me with this assignment I figured better to submit something than nothing at all.

Task 1

Q1.1

Primary Key would be CID

Redundant would be EventName, EventDate, TeamCoach, TeamRating explained further in the coming examples these are redundancies as any changes would have to be made to multiple CID entries and therefore redundant.

Insert anomaly – if a new event or date were made for an event and with no competitors yet it would not be able to be created.

Deletion anomaly – if a coach were to retire this would delete all data about all the students the coach has with them deleting important data.

Update anomaly – if a date of an event were to be updated multiple lines would need to be changed thus increasing chance of errors and time wasting.

Q1.2

Competitor (CID,Cname,TeamName*)

Team (TeamName,TeamCoach)

Event (EventID,EventName,EventDate)

Competing (EventID*,CID*)

Rating (TeamName*,EventID*,TeamRating)

Task 2

Q2.1

Redundant would be staffNum,Staffname which is 1 as would be joined and vanNum as only 1 van can be with 1 staff member at a time and only 1 staff member would see 1 customer these are made redundant.

Insert anomaly – if a new customer were to be made and no vans are available as checked out then would not be able to be created.

Deletion anomaly – since more than 1 customer can be at 1 address if they were to delete address important data for other customers would also be deleted.

Update anomaly – if address were to be updated would change address on multiple customers that may not have changed address also.

Q2.2

jobNum -> custNum,custAddress
staffNum -> staffName
vanNum -> jobNum,staffNum,date

Candidate Keys would be jobNum(audit), StaffNum and vanNum

Q2.3

existing form Audit->custAdress, custNum, date, staffNum,staffName,vanNum

1NF multiple address could input into the same audit for the staff member doing that audit.

Audit -> custNum, date, staffNum, staffName, vanNum
custNum -> Address

2NF we can identify staffName with staffNum being the same alongside the customer.

Audit ->custNum,custAdress,date,vanNum
Staff-> staffNum,StaffName

3NF vanNum should be dependant on another primary of staff which is not.

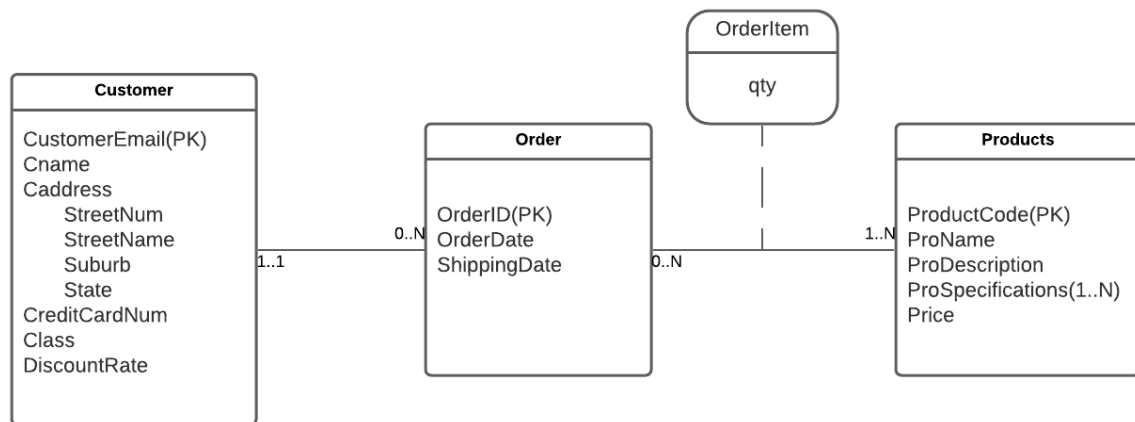
Audit ->custNum,custAdress
staff->staffNum,StaffName
van->auditNum, staffNum

Q2.4

customer(custNum,customerAddress)
staff(staffNum,staffName)
van(vanID,vanName)
audit(custnum*,staffnum*,vanid*,date)

Task 3

Q3.1



Assumptions are made that the shipping date will be updated in the system once order has been sent. The others are straight forward customer can have 0 or multiple orders each order has 1 to many products but must have at least 1, products do not need to be on an order but can belong to multiple orders. Each order can only belong to 1 customer.

Q3.2

Step 1 Entities

step 1.1 Strong Entities

Customer(CustomerEmail, Cname, Address, CreditCardNum, Class, DiscountRate)

Order(OrderID, OrderDate, ShippingDate)

Products(ProductCode, ProName, ProDescription, Price)

Step 1.2 Weak entities

none

Step 2 – Binary relationships

step 2.1 1:1 relationships

none

step 2.2 1:N relationships

Order(OrderID, OrderDate, ShippingDate, CustomerEmail*)

step 2.3 N:N relationships

OrderedItems(OrderID*, ProductCode*, Qty)

step 3 Multi-valued attributes

ProSpecs(Procode*, ProSpecifications)

Step 4 higher degree

none

Final Schema

Order (OrderID, OrderDate, ShippingDate, CustomerEmail*)

OrderedItems(OrderID*, ProductCode*, Qty)

ProSpecs(ProductCode*, ProSpecifications)

Customer(CustomerEmail, Cname, Caddress, CreditCardNum, Class, DiscountRate)

Products(ProductCode, ProName, ProDescription, Price)

Q3.3

OrderId-> OrderDate, ShippingDate, CustomerEmail

orderedItems-> orderID, productcode, qty

customer-> CustomerEmail, Cname, Caddress, CreditCardNum, Class, DiscountRate

Products-> ProductCode, ProName, ProDescription, Price, prospects

1NF- I believe this holds the form of 1NF as even though the user may hold multiple orders each order would be atomic therefore individual lines.

2NF- I believe that it takes the form of 2NF as the nonprime attributes are not reliant on a part of other candidate keys. I may however be wrong on this account.

3NF- Once again I believe this holds true to the third form factor as the attributes hold true on solely the primary key all non-prime attributes only depend on candidate keys and do not have a dependency on another key. This is my understanding once again may be wrong as although ordereditems relies on order ids, but the attributes do not rely on this however rely on a passed through id that would be the primary for that section.

Task 4

The following works on any new insert and I am assuming that is all the question is asking if it were to be updated or removed, I would assume more triggers would need to be wrote however with much the same coding. This kind of SQL using the DB and writing the queries is fun, form factors are not.

Q4.1

```
CREATE TRIGGER after_insert_works_on
AFTER INSERT
ON works_on
FOR EACH ROW
BEGIN
UPDATE works_on_total
SET totalProjects = (
        SELECT count(Pno)
        FROM works_on
        WHERE ESSN = NEW.essn),
totalHours = (
        SELECT sum(hours)
        FROM works_on
        WHERE essn = New.essn)
WHERE essn = new.essn;
END;
```