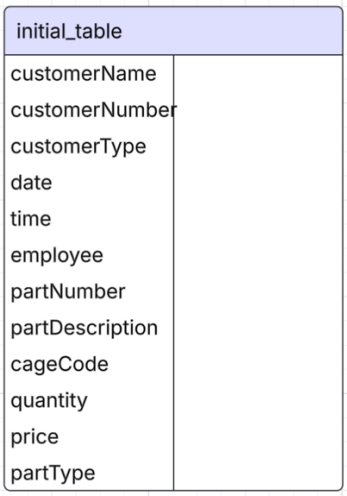
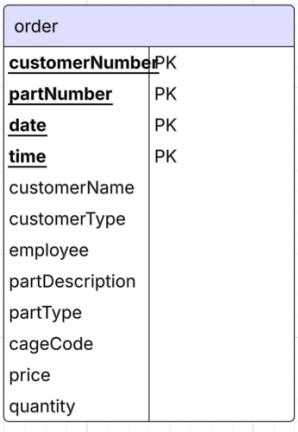
Question 1

a. Assumptions:
   Order uniqueness: each order is identified by (cutomerNumber, Date, Time) since there is no orderNumer in the table.
   Employee uniqueness: employeeName indentifies unqiue employee. We assume no employees have the same name.
   Customer uniqueness: customerNumber identifies unique customer.
   Part uniqueness: partNumber identifies unique part.
   Price is fixed for a part.

b. Normalizations
   1. The initial table looks like this:

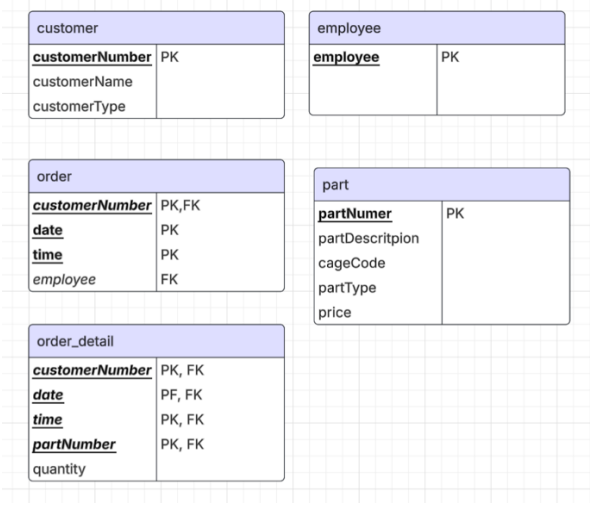



   2. It is at 1 NF because there's no repetitive group. We can define a primary key to make sure each row is identified by (customerNumber, date, time, partNumber).




   3. It is not at 2 NF because there is partial dependency. To eliminate them by making sure all the non-primary attributes are only dependent on the primary key. partType, partDescription, cageCode, and price are partial dependent on partNumber, so there should be another table for them. And customerName and customerType are partial dependent on customerNumber, so they should be segregated too. An individual table should be created for employee, so we don't lose information of them when

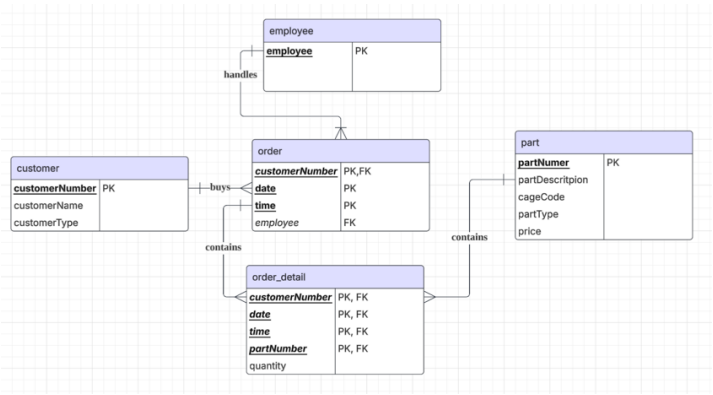
we delete an order. Also, an order might have multiple parts with different quantity. We need an order_detail table to reduce redundancy. So, we have:




4. To normalize to 3 NF, we need to eliminate transitive dependency. Here, there's no transitive dependency found. So, it's in 3 NF now.

c. The primary keys and foreign keys are noted in the graph.

d.




Question 2

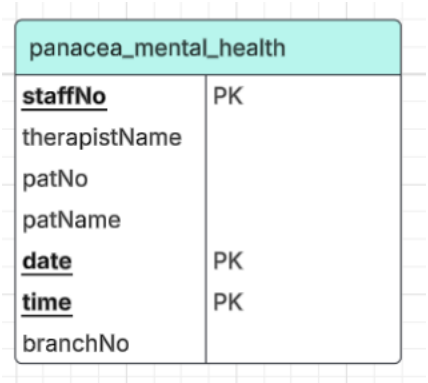
a. Assumptions are:
   1. A customer can have multiple appointments at different branches on different time.
   2. One therapist only works at one branch in a day.
   3. staffNo identifies unique therapists.
   4. patNo identifies unique customers.
   5. branchNo identifies unique branches.

b. The initial table is not in any NF because there's composite attribute. The appointment date column is a mixture of date and time.

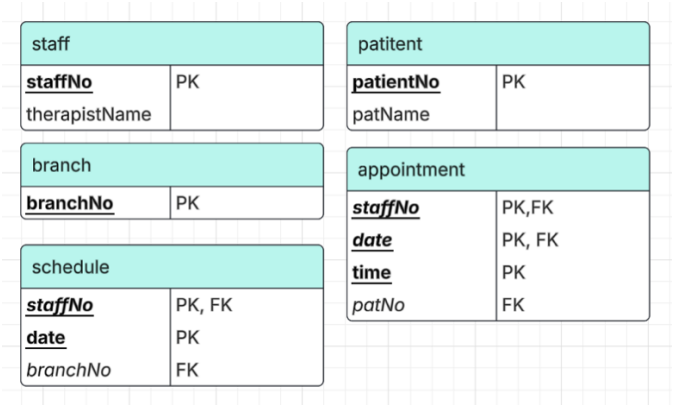
| staffNo | therapistName | patNo | patName | appointment | | branchNo |
|---|---|---|---|---|---|---|
| | | | | date | time | |
| S1011 | Fred Smith | P100 | Lily White | 9/12/2022 | 10:00 | M15 |
| S1011 | Fred Smith | P105 | Jill Baker | 9/12/2022 | 12:00 | M15 |
| S1024 | Heidi Pierce | P108 | Andy McKee | 9/12/2022 | 10:00 | Q10 |
| S1024 | Heidi Pierce | P108 | Andy McKee | 9/14/2022 | 14:00 | Q10 |
| S1032 | Richard Levin | P105 | Jill Baker | 9/14/2022 | 16:30 | M15 |
| S1032 | Richard Levin | P110 | Jimmy Winter | 9/15/2022 | 18:00 | B13 |

So, to normalize to 1 NF, I will separate appointment date and time into two columns.

And a composite key of (staffNo, date, time) can be a primary key of the table.

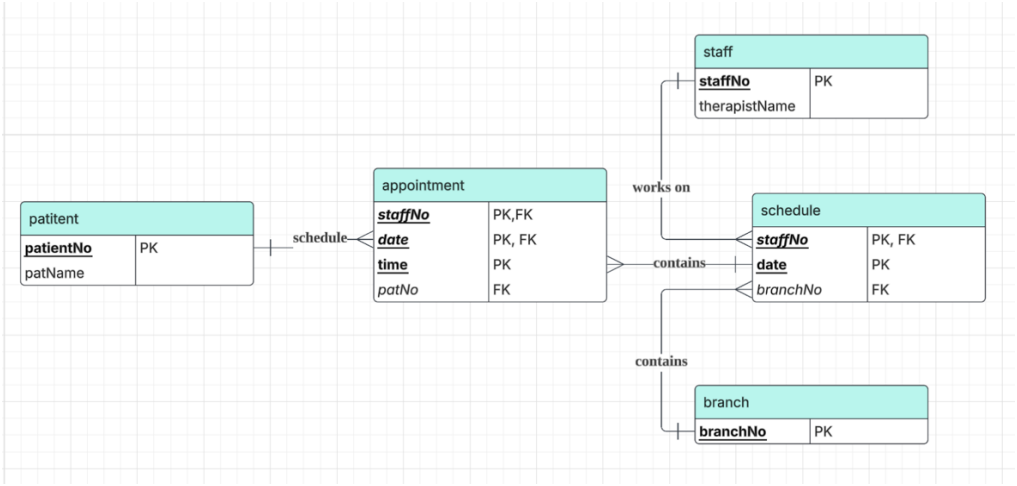



This table is not in 2 NF because there is partial dependency. The therapistName is only dependent on staffNo, patName is only dependent on patNo (non-key attribute). So, we should split this information into different tables. Also, to prevent branch information from being intervened by deleting appointment information, the branchNo should be stored in another table. Additionally, we have the assumption that a therapist only works at one branch a day. So, the branchNo does not depend on the whole primary key in the appointment table but the staffNo and date (time does not affect which branch is the therapist working). So a table of schedule should be created to avoid transitive




This table is in 3 NF as no transitive dependency exist.

The relationship is shown below:




c. The primary keys and foreign keys are shown in the diagram.
d. The relations are shown in the diagram.

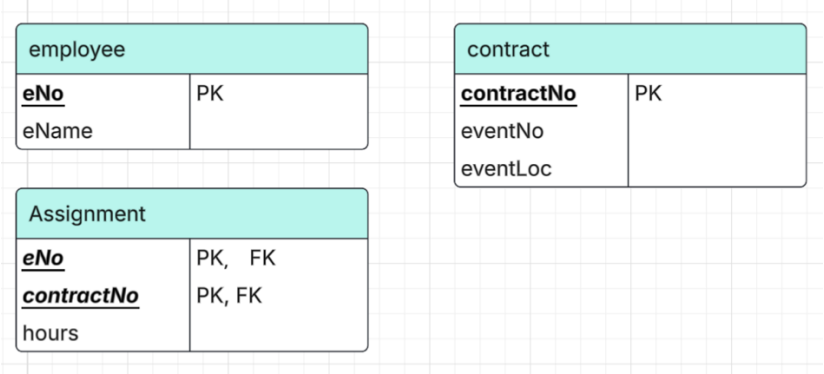
Question 3

a. Assumptions are:
   1. Each contract only applies to one event. There may be different contracts for an event depending upon different service needs.
   2. evenNo identifies unique eventLoc.
   3. eNo identifies unique employee.
   4. An employee can only work on
b. Initial table looks like:

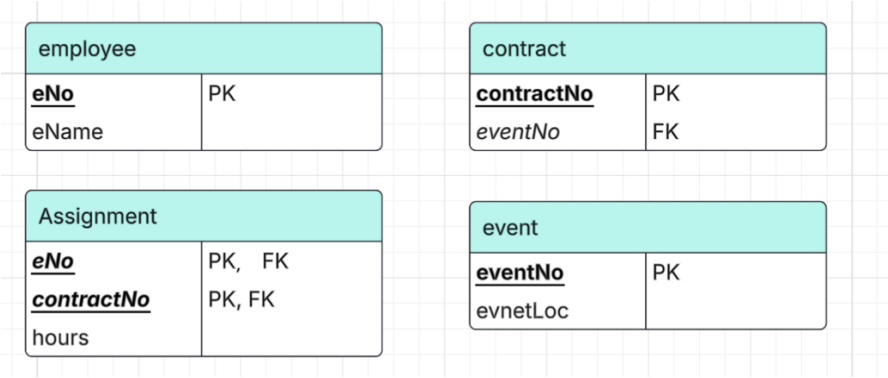
| eNo | contractNo | hours | eName | eventNo | eventLoc |
|---|---|---|---|---|---|
| 1135 | C1024 | 16 | Smith J | H25 | Queens |
| 1057 | C1024 | 24 | Hocine D | H25 | Queens |
| 1068 | C1025 | 28 | White T | H4 | Yonkers |
| 1135 | C1025 | 15 | Smith J | H4 | Yonkers |
| 1135 | C1026 | 10 | Smith J | H25 | Queens |

The table has no repetitive group, so it's in 1 NF.

To normalize to 2 NF, we need to erase partial dependency on the primary key of the table. A candidate key for the table is (eNo , contraNo).
Now we find that eName is only dependent on eNo, eventNo only depends on contractNo. So, we need to segregate them into different tables. Then we have:




Now it's in 2 NF, but not in 3 NF. Because evnetLoc and evntNo might have transitive dependency. We need to create another table for events.




Now the table is in 3 NF.

c. The keys are shown in the diagram.

d. The relations are shown in the diagram.