# Assignment4 Normalization

Zien zz5106

## Exercise 1

(a). Assumptions
1: Employee name identifies the employee who processed the order and depends on the order.
2: PartNumber identifies a part uniquely
3: An order is uniquely identified by CustomerNumber, Date, Time.
4: CustomerNumber identifies customer uniquely
5: Unitprice, PartName, PartType depend only on PartNumber
6: PartType identifies the category of a part, and CageCode depends only on PartType, not directly on PartNumber.
7. CustomerName and CustomerType depend only on CustomerNumber.
8. QuantityOrdered depends on CustomerNumber, PartNumber, date and time.

(b). Normalization Process

UnnormalizedForm:

| Customer Name | Customer Number | Customer Type | Date | Time | Employee | Part Number | Part Name | Part Type | Cage Code | Quantity Ordered | Unit Price |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Jeff Peterson | HG54587 | Consumer | 7/1/2024 | 10:30am | D.Harrison | 10654 | Float Control | Plumbing | G413 | 4 | 12 |
| | | | | | | 10456 | Modulator | Electrical | H433 | 3 | 7 |
| | | | | | | 10776 | Hose Assembly | Plumbing | G413 | 7 | 9 |
| | | | | | | 10657 | Float Assembly | Plumbing | G413 | 5 | 10 |

Step1-1NF
Table: OrderForm

| Customer Name | **customer Number** | customerType | **date** | **time** | employee | **partNumber** | partName | partType | cageCode | quantityOrdered | unitPrice |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Jeff Peterson | HG54587 | Consumer | 7/1/2024 | 10:30am | D.Harrison | 10654 | Float Control | Plumbing | G413 | 4 | 12 |
| Jeff Peterson | HG54587 | Consumer | 7/1/2024 | 10:30am | D.Harrison | 10456 | Modulator | Electrical | H433 | 3 | 7 |
| Jeff Peterson | HG54587 | Consumer | 7/1/2024 | 10:30am | D.Harrison | 10776 | Hose Assembly | Plumbing | G413 | 7 | 9 |
| Jeff Peterson | HG54587 | Consumer | 7/1/2024 | 10:30am | D.Harrison | 10657 | Float Assembly | Plumbing | G413 | 5 | 10 |

Why:
The unnormalized form had a hierarchical structure with repeated groups of parts under one order header.
To achieve 1NF, we need to create a flat table where each row represents one order line item. So there are no repeating groups and every cell has an atomic value.

The primary keys are **customerNumber, partNumber, date, time**, they uniquely identify each order line.

Step2-2NF
Table: Customer

| **customerNumber** | customerName | customerType |
|---|---|---|
| HG54587 | Jeff Peterson | Consumer |

Table: Order

| **customerNumber** | **date** | **time** | employee |
|---|---|---|---|
| HG54587 | 7/1/2024 | 10:30am | D.Harrison |

Table: Part

| **partNumber** | partName | partType | cageCode | unitPrice |
|---|---|---|---|---|
| 10654 | Float Control | Plumbing | G413 | 12 |
| 10456 | Modulator | Electrical | H433 | 7 |
| 10776 | Hose Assembly | Plumbing | G413 | 9 |
| 10657 | Float Assembly | Plumbing | G413 | 10 |

Table: OrderLine

| **customerNumber** | **date** | **time** | **partNumber** | quantityOrdered |
|---|---|---|---|---|
| HG54587 | 7/1/2024 | 10:30am | 10654 | 4 |
| HG54587 | 7/1/2024 | 10:30am | 10456 | 3 |
| HG54587 | 7/1/2024 | 10:30am | 10776 | 7 |
| HG54587 | 7/1/2024 | 10:30am | 10657 | 5 |

Why:
1NF contained partial dependencies that
CustomerName and CustomerType depend only on CustomerNumber,
Employee depends on CustomerNumber, Date and Time,
PartName, PartType, CageCode, and UnitPrice depend only on PartNumber
Only QuantityOrdered depends on the full primary key.
To reach 2NF, we need to decompose the 1NF table into four tables based on the functional dependencies and the primary key(s) each table needs to have. So that these data only need to be stored once.

Step3-3NF
Table: Customer

| **customerNumber** | customerName | customerType |
|---|---|---|
| HG54587 | Jeff Peterson | Consumer |

Table: Order

| ***customerNumber*** | **date** | **time** | employee |
|---|---|---|---|
| HG54587 | 7/1/2024 | 10:30am | D.Harrison |

Table: Part

| **partNumber** | partName | *partType* | unitPrice |
|---|---|---|---|
| 10654 | Float Control | Plumbing | 12 |
| 10456 | Modulator | Electrical | 7 |
| 10776 | Hose Assembly | Plumbing | 9 |
| 10657 | Float Assembly | Plumbing | 10 |

Table: PartTypeStorage

| **partType** | cageCode |
|---|---|
| Plumbing | G413 |
| Electrical | H433 |
| Plumbing | G413 |
| Plumbing | G413 |

Table: OrderLine

| ***customerNumber*** | ***date*** | ***time*** | ***partNumber*** | quantityOrdered |
|---|---|---|---|---|
| HG54587 | 7/1/2024 | 10:30am | 10654 | 4 |
| HG54587 | 7/1/2024 | 10:30am | 10456 | 3 |
| HG54587 | 7/1/2024 | 10:30am | 10776 | 7 |
| HG54587 | 7/1/2024 | 10:30am | 10657 | 5 |

Why:

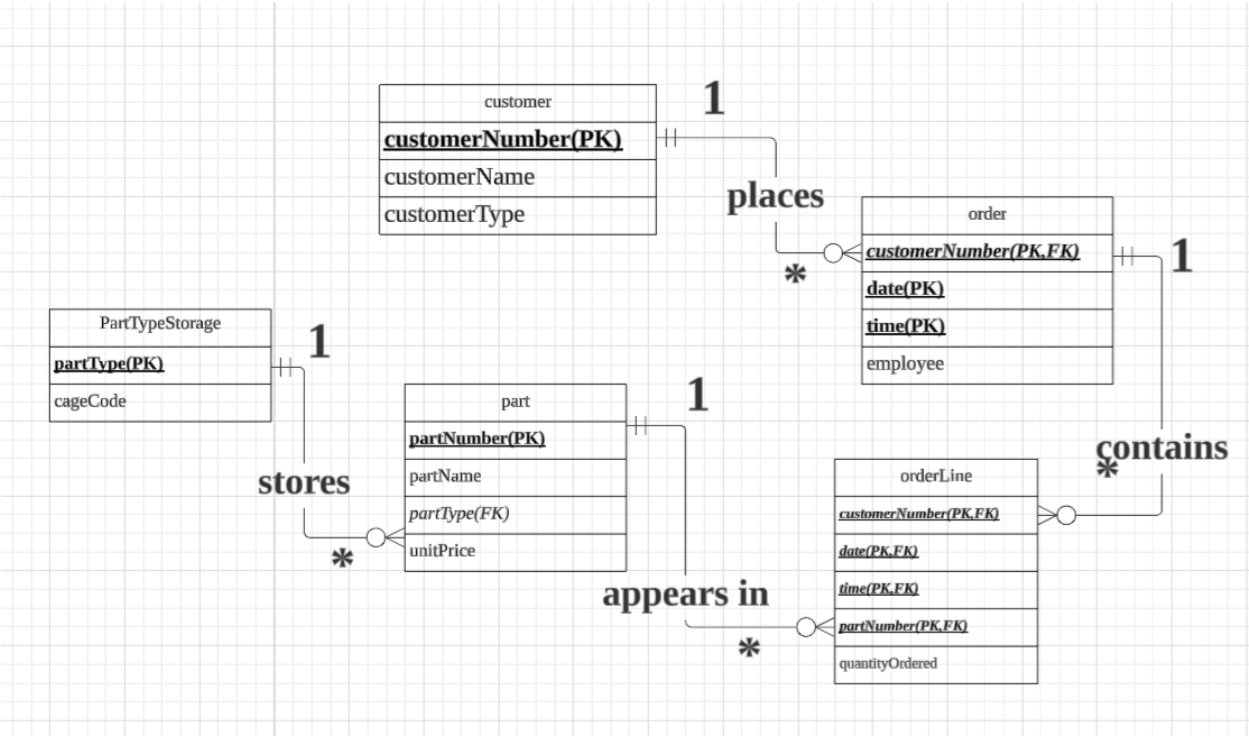
A transitive dependency existed because CageCode depends on PartType, and PartType depends on PartNumber.
To eliminate this transitive dependency and achieve 3NF, CageCode was moved to a separate table keyed by PartType.

(c) Primary and Foreign Keys
1NF:
Table: orderForm
PK: (customerNumber, partNumber, date, time)

2NF:
Table: Customer
PK: customerNumber
Table: Order
PK: (customerNumber, date,time)
Table: Part
PK: partNumber
Table: OrderLine
PK: (customerNumber, partNumber, date, time)

3NF:
Table: Customer
PK: customerNumber
Table: Order
PK: (customerNumber, date,time)
FK: customerNumber
Table: PartTypeStorage:
PK: PartType
Table: Part
PK: partNumber
FK: partType
Table: OrderLine
PK: (customerNumber, partNumber, date, time)
FK: customerNumber, partNumber, date, time

(d) names of 3NF relations:
Customer, Order, Part, PartTypeStorage, OrderLine




**Exercise 2**

(a). Assumptions

1: PatNo uniquely identifies each patient.

2: TherapistName depends only on StaffNo.

3: StaffNo uniquely identifies each therapist.

4: PatName depends only on PatNo.

5: A patient may have multiple appointments with different therapists.

6: A therapist can only work at one branch on any given day.

(b). Normalization Process

Unnormalized Form

| **staffNo** | therapistNam e | **patNo** | patName | **appointment date time** | branchNo |
|---|---|---|---|---|---|
| S1011 | Fred Smith | P100 | Lily White | 9/12/2022 | M15 |

| | | | | | |
|---|---|---|---|---|---|
| | | | | 10:00 | |
| 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 |

Step1-1NF:
Table: AppointmentAll

| **staffNo** | therapistName | **patNo** | patName | **appointmentDate** | **appointmentTime** | branchNo |
|---|---|---|---|---|---|---|
| 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 |

Why:
The unnormalized form violated 1NF because the appointment date time column contains non-atomic data, so we need to split it into appointmentDate and appointmentTime to ensure all attributes contain atomic values.

Step2-2NF:
Table: Therapist

| **staffNo** | therapistName |
|---|---|
| S1011 | Fred Smith |
| S1024 | Heidi Pierce |
| S1032 | Richard Levin |

Table: Patient

| **patNo** | patName |
|---|---|
| P100 | Lily White |
| P105 | Jill Baker |
| P108 | Andy McKee |
| P110 | Jimmy Winter |

Table: Appointment

| **staffNo** | **patNo** | **appointmentDate** | **appointmentTime** | branchNo |
|---|---|---|---|---|
| S1011 | P100 | 9/12/2022 | 10:00 | M15 |
| S1011 | P105 | 9/12/2022 | 12:00 | M15 |
| S1024 | P108 | 9/12/2022 | 10:00 | Q10 |
| S1024 | P108 | 9/14/2022 | 14:00 | Q10 |
| S1032 | P105 | 9/14/2022 | 16:30 | M15 |
| S1032 | P110 | 9/15/2022 | 18:00 | B13 |

Why:
1NF table contained partial dependencies, which will cause redundancy that therapists and patient names are repeated for each appointment. By creating separate tables, we can eliminate redundancy and store these data only once.

Step3-3NF:
Table: Therapist

| **staffNo** | therapistName |
|---|---|
| S1011 | Fred Smith |
| S1024 | Heidi Pierce |
| S1032 | Richard Levin |

Table: Patient

| **patNo** | patName |
|---|---|
| P100 | Lily White |
| P105 | Jill Baker |
| P108 | Andy McKee |
| P110 | Jimmy Winter |

Table: TherapistAssignment

| ***staffNo*** | **appointmentDate** | branchNo |
|---|---|---|
| S1011 | 9/12/2022 | M15 |
| S1024 | 9/12/2022 | Q10 |
| S1024 | 9/14/2022 | Q10 |
| S1032 | 9/14/2022 | M15 |
| S1032 | 9/15/2022 | B13 |

Table: Appointment

| ***staffNo*** | ***patNo*** | ***appointmentDate*** | **appointmentTime** |
|---|---|---|---|
| S1011 | P100 | 9/12/2022 | 10:00 |
| S1011 | P105 | 9/12/2022 | 12:00 |
| S1024 | P108 | 9/12/2022 | 10:00 |
| S1024 | P108 | 9/14/2022 | 14:00 |
| S1032 | P105 | 9/14/2022 | 16:30 |
| S1032 | P110 | 9/15/2022 | 18:00 |

Why:

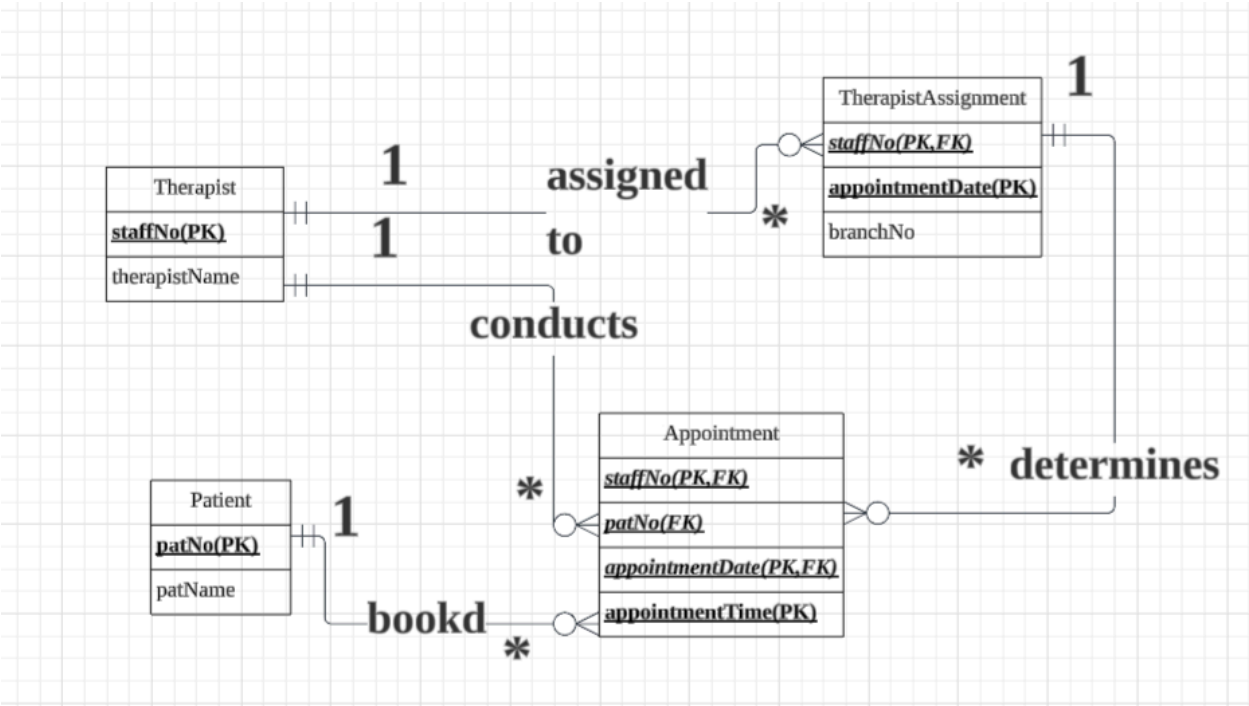
BranchNo depends on the composite key (StaffNo, AppointmentDate) rather than on the full primary key of Appointment, creating a transitive dependency.
Separating TherapistAssignment eliminates this dependency and achieves 3NF.

(c) Primary and Foreign Keys
1NF:
Table: AppointmentAll
PK: (staffNo, patNo, appointmentDate, appointmentTime)

2NF:
Table: Therapist
PK: staffNo
Table: Patient
PK: patNo
Table: Appointment
PK: (staffNo, patNo, appointmentDate, appointmentTime)

3NF:
Table: Therapist
PK: staffNo
Table: Patient
PK: patNo
Table: TherapistAssignment
PK: (staffNo, appointmentDate)
FK: staffNo
Table: Appointment)
PK: (staffNo, patNo, appointmentDate, appointmentTime)
FK: (staffNo, appointmentDate), patNo, StaffNo

(d) names of 3NF relations:
Therapist, Patient, TherapistAssignment, Appointment




**Exercise 3**

(a). Assumptions

1: eNo uniquely identifies each employee.

2: Hours worked depends on both eNo and contractNo.

3: Employee name depends only on eNo.

4: eventNo uniquely identifies each event.

5. contractNo uniquely identifies each contract.

6. Event location depends only on eventNo.

(b). Normalization Process

Unnormalized Form

| **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 |

Step1-1NF:

Table: Appointment

The Unnormalized Form already satisfies 1NF because all attributes contain atomic values and there are no repeating groups.

Step2-2NF:
Table: Employee

| **eNo** | eName |
|---|---|
| 1135 | Smith J |
| 1057 | Hocine D |
| 1068 | White T |

Table: Contract

| **contractNo** | eventNo | eventLoc |
|---|---|---|
| C1024 | H25 | Queens |
| C1025 | H4 | Yonkers |
| C1026 | H25 | Queens |

Table: Assignment:

| **eNo** | **contractNo** | hours |
|---|---|---|
| 1135 | C1024 | 16 |
| 1057 | C1024 | 24 |
| 1068 | C1025 | 28 |
| 1135 | C1025 | 15 |
| 1135 | C1026 | 10 |

Why:
The 1NF table contained partial dependencies causing redundancy that the same employee's name might be repeatedly mentioned for each of his work, the location queens and yonkers are also repeated. We can separate Employee and Contract tables to eliminate this redundancy.

Step3-3NF:
Table: Employee

| **eNo** | eName |
|---|---|
| 1135 | Smith J |
| 1057 | Hocine D |

| | |
|---|---|
| 1068 | White T |

Table: Event

| **eventNo** | eventLoc |
|---|---|
| H25 | Queens |
| H4 | Yonkers |

Table: Contract

| **contractNo** | *eventNo* |
|---|---|
| C1024 | H25 |
| C1025 | H4 |
| C1026 | H25 |

Table: Assignment:

| ***eNo*** | ***contractNo*** | hours |
|---|---|---|
| 1135 | C1024 | 16 |
| 1057 | C1024 | 24 |
| 1068 | C1025 | 28 |
| 1135 | C1025 | 15 |
| 1135 | C1026 | 10 |

Why:
The 2NF Contract table contained a transitive dependency that eventLoc depends on eventNo, but not directly on contractNo.
This means if multiple contracts exist for the same event, the event location will be repeated several times in different records. So we need to create an event table to store each event's location once.

(c) Primary and Foreign Keys
1NF:
Table: Unnormalized Form
PK: eNO, contractNo

2NF:

Table: Employee
PK: eNo
Table: Contract
PK: contractNo
Table: Assignment
PK: (eNO, contractNo)

3NF:
Table: Employee
PK: eNo
Table: Event
PK: eventNo
Table: Contract
PK: contractNo
FK: eventNo
Table: Assignment
PK: (eNO, contractNo)
FK: eNO, contractNo

(d) names of 3NF relations:
Employee, Event, Contract, Assignment