# CZ2007: INTRODUCTION TO DATABASES

# LAB-3 REPORT

**SS2 Group 4:**

***Lim Wi Teow*** *(U1921765G)*

***Taneja Parthasarthi*** *(U1722927B)*

***Chong Zhe Ming*** *(U1920757K)*

***Soh Qian Yi*** *(U1922306C)*

***Jacob Law Zhen*** *(U1922430D)*

***Zhu Weiji*** *(U1922876G)*

***Poh Kai Kiat*** *(U1922819C)*

**Submission Date:** 25/9/2020

1. Shops

---

**Shops** (ShopName)
**Key**: ShopName
**Primary Key**: ShopName
**Functional Dependencies**:
(1) ShopName → ShopName

**Prove: If relation is in 3NF**

- **Shops** only has 1 attribute, hence **Shops** is already in BCNF.
- Since BCNF is stricter than 3NF, **Shops** is in 3NF.

2. Users

---

**Users** (UID, Name)
**Key**: UID
**Primary Key**: UID
**Functional Dependencies**:
(1) UID → Name

**Prove: If relation is in 3NF**

- **Users** only has 2 attributes, hence **Users** is already in BCNF.
- Since BCNF is stricter than 3NF, **Users** is in 3NF.

3. Orders

---

**Orders** (OID, Shipping-Address, Total-Shipping-Cost, UID)
**Key**: OID
**Primary Key**: OID
**Functional Dependencies**:
(1) OID → Shipping-Address, Total-Shipping-Cost, UID

**Prove: If relation is in 3NF**

- FD (1) is non-trivial because the left side of FD (1) contains OID, which is a key to **Orders**, so **Orders** is in BCNF.
- Since BCNF is stricter than 3NF, so **Orders** is in 3NF.

## 4. Complaints

---

**Complaints** (<u>ComplaintID</u>, UID,  EID, Handled-Date-Time, Text, Status, Filled-Date-Time)
**Key**: ComplaintID
**Primary Key**: ComplaintID
**Functional Dependencies**:

(1) ComplaintID → UID,  EID, Handled-Date-Time, Text, Status, Filled-Date-Time
(2) EID + Handled-Date-Time + Filled-Date-Time → Status

**Assumptions:** Each employee can only handle 1 complaint at a time.
**Prove: If relation is in 3NF**

- The relation is not in 3NF because FD (2) violates 3NF . The LHS does not contain a key and its RHS is not contained in a key or in its LHS.
- FD (2) occurs when:
    a) When an employee (EID = 1) handles a complaint filed at 12.35pm (Filled-Date-Time) at 12.40pm (Handle-Date-Time), the status of the complaint can be determined.
    b) Suppose that no employee handles the complaint, there will only be Handle-Date-Time and no Filled-Date-Time, the status of the complaint can still be determined, which is 'Not-Handled'.

### 4.1 Decomposing **Complaints**

1) **A minimal basis for the FDs** : {ComplaintID → UID, ComplaintID → EID, ComplaintID → Handled-Date-Time , ComplaintID → Text , ComplaintID → Status , ComplaintID → Filled-Date-Time , EID + Handled-Date-Time + Filled-Date-Time → Status}
2) **Combine FDs with the same left hand side**:
   S = {(ComplaintID → UID,  EID, Handled-Date-Time, Text, Status, Filled-Date-Time) ,(EID + Handled-Date-Time + Filled-Date-Time → Status)}
3) **For each FDs in S create a table**:
    a) **Complaints_1** (<u>ComplaintID</u>, UID, EID, Handled-Date-Time, Text, Status, Filled-Date-Time)
       **Key** : ComplaintID
       **Primary Key**: ComplaintID
       **Functional Dependencies**:
       (1) ComplaintID → UID,  EID, Handled-Date-Time, Text, Status, Filled-Date-Time
       Therefore, the relation is in 3NF.

    b) **Complaints_2** (<u>EID</u>, <u>Handled-Date-Time</u>, <u>Filled-Date-Time</u>, Status)
       **Keys**: {(EID, Handled-Date-Time, Filled-Date-Time)}
       **Primary Key**: (EID, Handled-Date-Time, Filled-Date-Time)
       **Functional Dependencies**:

(1) EID + Handled-Date-Time + Filled-Date-Time → Status

Therefore, the relation is in 3NF.

## 5. Complaints-On-Shops

---

**Complaints-On-Shops** (ER Approach) (ComplaintID , ShopName)

**Key**: ComplaintID

**Primary Key**: ComplaintID

**Functional Dependencies**:

(1) ComplaintID → ShopName

**Assumptions**: We have used the ER Approach for this subclass.

**Prove: If relation is in 3NF**

- **Complaints-On-Shops** only has 2 attributes, hence **Complaints-On-Shops** is already in BCNF.
- Since BCNF is stricter than 3NF, so **Complaints-On-Shops** is in 3NF.

## 6. Complaints-On-orders

---

**Complaints-On-Orders** (ER Approach) (ComplaintID, OID)

**Key**: ComplaintID

**Primary Key**: ComplaintID

**Functional Dependencies**:

(1) ComplaintID → OID

**Assumptions**: We have used the ER Approach for this subclass.

**Prove: If relation is in 3NF**

- **Complaints-On-Orders** only has 2 attributes, hence **Complaints-On-Orders** is already in BCNF.
- Since BCNF is stricter than 3NF, so **Complaints-On-Orders** is in 3NF.

## 7. Employees

---

**Employees** (EID, Name, Salary)
**Key**: EID
**Primary Key:** EID
**Functional Dependencies**:
(1) EID → Name, Salary

**Prove: If relation is in 3NF**

- FD (1) is non-trivial because the left side of FD (1) contains EID, which is a key to **Employees**, so **Employees** is in BCNF.
- Since BCNF is stricter than 3NF, so **Complaints-On-Orders** is in 3NF.

## 8. Products

---

**Products** (PID, ProductName, Maker, Category)
**Key:** PID
**Primary Key**: PID
**Functional Dependencies**:
(1) ProductName → Maker, Category
(2) PID → ProductName, Maker, Category

**Assumptions:**
Every single item will have a unique **PID** regardless of the shop, maker or category. For example, ShopA which sells iPhone X, will have a unique **PID** for all iPhone X's they sell. All iPhone X will have the same **PID**. However, Shop A's iPhone X's **PID** will be different from Shop B's.

**Prove: If relation is in 3NF**

- The left hand side of FD (1) does not contain a key and no attributes on the right hand side of the FD is not contained in the key.
- Therefore, **FD(1)** is not in 3NF.
- FD (2) is in 3NF because the left hand side of FD (2) contains a key.

### 8.1 Decomposing **Products**:

1) **Minimal Basis for FDs**: {(ProductName → Maker), (ProductName → Category), (PID → ProductName), (PID → Maker), (PID → Category)}
2) **Combine FDs with same LHS**: {(ProductName → Maker, Category), (PID → ProductName)}
3) **Create a Table for each FD remained**:
    a) **Products_1** (ProductName, Maker, Category)

**Key**: ProductName
**Primary Key**: ProductName
**Functional Dependencies**:
(1) ProductName → Maker, Category
Therefore the relation is in 3NF.

b) **Products_2** (PID, ProductName)
**Key**: PID
**Primary Key**: PID
**Functional Dependencies**:
(1) PID -> ProductName
Therefore, the relation is in 3NF.

## 9. Products-In-Orders

---

**Products-In-Orders** (PID, OID, ShopName, Status, Delivery-date, Price, Quantity)
**Keys**: {(PID, OID)}
**Primary Key**: (PID, OID)
**Functional Dependencies**:
(1) (PID, OID) → ShopName, Status, Delivery-date, Price, Quantity

**Assumptions**: We have used the ER Approach for this subclass.
**Prove: If relation is in 3NF**
- FD (1) is non-trivial, both (PID, OID) are keys to **Products-In-Orders**, so **Products-In-Orders** is in BCNF.
- Since BCNF is stricter than 3NF, **Products-In-Orders** is in 3NF.

## 10. Products-In-Shops

---

**Products-In-Shops** (ShopName, PID, Price, Quantity)
**Keys**: {(ShopName, PID)}
**Primary Key**: (ShopName, PID)
**Functional Dependencies**:
(1) (ShopName, PID) → Price, Quantity

**Assumptions**: We have used the ER Approach for this subclass.
**Prove: If relation is in 3NF**
- FD (1) is non-trivial because the left side of the FD contains (ShopName, PID), which is a key to **Products-In-Shops**, so **Products-In-Shops** is in BCNF.
- Since BCNF is stricter than 3NF, so **Products-In-Shops** is in 3NF.

## 11. Price History

**Price History** (PID, Start-Date, End-Date, Price)
**Keys**: {(PID, Start-Date), (PID, End-Date)}
**Primary Key**: (PID, Start-Date)
**Functional Dependencies**:

(1) (PID, Start-Date) → End-Date, Price
(2) (PID, End-Date) → Start-Date, Price

**Assumptions**: For a given PID, there can be multiple instances of prices at different dates. However, if **(PID, Start-Date)** is a key, there can only be one listing of the Price at that time frame. This is because the price history is recorded in chronological order and hence every Start-Date for a particular PID will be unique. Hence from these assumptions, we have decided that both **(PID, Start-Date)** and **(PID, End-Date)** are keys for **Price History**. We have decided for **(PID, Start-Date)** to be the primary key as for **(PID, End-Date),** the current price listing will not have an End-Date as the price has not changed yet.

**Prove: If relation is in 3NF**

- Both FD (1) and FD (2) are non-trivial because the left side of both FDs contain (PID, Start-Date) and (PID, End-Date), which is a key to **Price History**.
- Since both FDs are non-trivial, **Complaints-On-Orders** is hence in 3NF.

## 12. Feedback

**Feedback** (UID, PID, Date-Time, Rating, Comment)
**Keys**: {(UID, PID, Date-Time)}
**Primary Key**: (UID, PID, Date-Time)
**Functional Dependencies**:

(1) (UID, PID, Date-Time) → Rating, Comment

**Assumptions:** User can rate once after every purchase of the same item, this means that given **UID** and **PID** will result in multiple entries of the same user giving the same product different comments and ratings at different timings.

**Prove: If relation is in 3NF**

- FD (1) is non-trivial because the left side of the FD contains (UID, PID, Date-Time), which is a key to **Feedback**, so **Feedback** is in BCNF.
- Since BCNF is stricter than 3NF, so **Feedback** is in 3NF.

13. IN(Shops_Products-In-Shops)

---

**IN(Shops_Products-In-Shops)** (ShopName, PID)
**Key:** {(ShopName, PID)}
**Primary Key**: (ShopName, PID)
**Functional Dependencies**:

(1) (ShopName, PID) → ShopName, PID

**Prove: If relation is in 3NF**

- FD (1) is non-trivial because the left side of the FD contains (ShopName, PID), which is a key to **IN(Shops_Products-In-Shops)**, so **IN(Shops_Products-In-Shops)** is in BCNF.
- Since BCNF is stricter than 3NF, so **IN(Shops_Products-In-Shops)** is in 3NF.

14. IN(Shops_Products-In-Orders)

---

**IN(Orders_Products-In-Orders)** (PID, OID, Date-Time)
**Key:** {(PID, OID)}
**Primary Key**: (PID, OID)
**Functional Dependencies**:

(1) (PID, OID) → Date-Time

**Prove: If relation is in 3NF**

- FD (1) is non-trivial because the left side of the FD contains (ShopName, PID), which is a key to **IN(Orders_Products-In-Orders)**, so **IN(Orders_Products-In-Orders)** is in BCNF.
- Since BCNF is stricter than 3NF, so **IN(Orders_Products-In-Orders)** is in 3NF.

**Individual Components**
**Report:**
Lim Wi Teow (U1921765G)
Taneja Parthasarthi (U1722927B)
Chong Zhe Ming (U1920757K)
Soh Qian Yi (U1922306C)
Jacob Law Zhen (U1922430D)
Zhu Weiji (U1922876G)
Poh Kai Kiat (U1922819C)