

## **Homework 3 Part2**

### **Group members:**

**Yiwei Zhao A20303653**

**Shengyuan Ye A20354182**

**Kunyang Zhao A20359397**

## Project Design Report

### Introduction

The team members of our project are Kunyang Zhao, Yiwei Zhao, Shengyuan Ye. There are 20 schemas in our project, every schema satisfies at least 3NF. After finalizing the schemas, we found out the functional dependence( FD). This report will discuss what attributes, FD and keys every schema has, and give the proof why this schema satisfy BCNF or 3NF in the discussion part.

### Discussion

Use the E/R-Style Conversion approach to deal with the subclass structures: User, Staff, Member.

1. f15\_user(id, name, phone, address, password)

Primary key: id

Candidate key: phone

In our ER diagram, this schema corresponds to the user entity. It's the core of the diagram, also the core of the whole program. It has two subclass: Staff and Member and it has relation with Ticket entity that it can buy ticket. The "id" is the primary key. The "phone" is the candidate key.

FD: id→name, phone, address, password

phone→id, name, address, password

This schema is BCNF, the FD satisfy BCNF, since "id" and "phone" is key, so "id" and "phone" is also superkey.

```
CREATE TABLE f15_user(  
id NUMBER(4) PRIMARY KEY,  
name VARCHAR2(30) NOT NULL,  
phone VARCHAR2(30) UNIQUE,  
address VARCHAR2(100) NOT NULL,  
password VARCHAR2(30) NOT NULL  
)
```

2.f15\_member(id,userid,email, creditCardNum, creditPoint)

Primary key: id

Candidate key: email

Foreign key: userid references id in f15\_user

Foreign key: creditCardNum references creditCardNum in f15\_creditCard

Foreign key: creditPoint references creditPoint in f15\_membershipStatus

This schema corresponds to the Member entity in our ER diagram. "id" is the primary key. The "email" is the candidate key. The "userid" is the foreign key referencing "id" in f15\_user. "creditCardNum" is the foreign key referencing "creditCardNum" in f15\_creditCard. "creditPoint" is the foreign key referencing "creditPoint" in f15\_membershipStatus. Besides, It is a subclass of User entity, and we use the E/R-Style Conversion approach to deal with the subclass structures, so Member entity also has some hidden attributes that are connected by foreign key "userid" with User entity——name, phone, address, password.

FD: id→userid, email, creditCardNum, creditPoint

This schema is BCNF, the FD satisfy BCNF, since “id” is key, so “id” is also superkey.

```
CREATE TABLE f15_member(  
id NUMBER(4) PRIMARY KEY,  
userid NUMBER(4) REFERENCES f15_user(userid),  
email VARCHAR2(100) UNIQUE,  
creditCardNum VARCHAR2(30) REFERENCES f15_creditCard(creditCardNum),  
creditPoint NUMBER(3) REFERENCES f15_membershipStatus(creditPoint)  
)
```

3.f15\_membershipStatus(creditPoint, status)

Primary key: creditPoint

This schema corresponds to the MembershipStatus entity in our ER diagram.

“creditPoint” is the primary key.

FD: creditPoint → status

This schema is BCNF, the FD satisfy BCNF, since “creditPoint” is key, so “creditPoint” is also superkey.

```
CREATE TABLE f15_membershipStatus(  
creditPoint NUMBER(3) PRIMARY KEY,  
status VARCHAR2(30) CHECK( status IN('SILVER','GOLD','PLATINUM'))  
)
```

4.f15\_creditCard(creditCardNum, creditType, expirationDate,balance)

Primary key: creditCardNum

This schema corresponds to the CreditCard entity in our ER diagram. “creditCardNum” is the primary key.

FD: creditCardNum → creditType, expirationDate, balance

This schema is BCNF, the FD satisfy BCNF, since “creditCardNum” is key, so “creditCardNum” is also superkey.

```
CREATE TABLE f15_creditCard(  
creditCardNum VARCHAR2(30) PRIMARY KEY,  
creditType VARCHAR2(30) NOT NULL,  
expirationDate DATE NOT NULL,  
balance NUMER(6,3)  
)
```

5. f15\_rewards(name, worthPoint)

Primary key: name

This schema corresponds to the Rewards entity in our ER diagram. “name” is the primary key.

FD: name → worthPoint

This schema is BCNF, the FD satisfy BCNF, since “name” is key, so “name” is also superkey.

```
CREATE TABLE f15_rewards(  
name VARCHAR2(30) PRIMARY KEY,  
worthPoint NUMBER(3) NOT NULL  
)
```

6.f15\_movie(id, title, director, description)

Primary key: id

This schema corresponds to the Movie entity in our ER diagram. “id” is the primary key.

FD: id → title, director, description

This schema is BCNF, the FD satisfy BCNF, since “id” is key, so “id” is also superkey.

```
CREATE TABLE f15_movie(  
id NUMBER(4) PRIMARY KEY,  
title VARCHAR2(30) NOT NULL,  
director VARCHAR2(30) NOT NULL,  
description VARCHAR2(1000) NOT NULL  
)
```

7.f15\_movieStar(id,name)

Primary key: id

This schema corresponds to the MovieStar entity in our ER diagram. “id” is the primary key.

FD: id → name,

This schema is BCNF, the FD satisfy BCNF, since “id” is key, so “id” is also superkey.

```
CREATE TABLE f15_movieStar(  
id NUMBER(4) PRIMARY KEY,  
name VARCHAR2(30)  
)
```

8.f15\_moviewithStar(movieid,starid)

Primary key: movieid, starid

This schema corresponds to the MoviewithStar relationship in our ER diagram.

“movieid” and “starid” together are the primary key. The “movieid” is the foreign key referencing “id” in f15\_movie. The “starid” is the foreign key referencing “id” in f15\_moviestar.

FD: none,

This schema is BCNF, since there is no FD.

```
CREATE TABLE f15_moviewithStar(  
movieid NUMBER(4) REFERENCES f15_movie(id),  
staridNUMBERB(4) REFERENCES f15_moviestar(id),  
CONSTRAINT moviewithStar_PK PRIMARY KEY( movieid,starid)  
)
```

9.f15\_movieType(id,typename)

Primary key: id

This schema corresponds to the MovieType entity in our ER diagram. "id" is the primary key.

FD:  $id \rightarrow \text{typename}$ ,

This schema is BCNF, the FD satisfy BCNF, since "id" is key, so "id" is also superkey.

```
CREATE TABLE f15_movieType(  
id NUMBER(4) PRIMARY KEY,  
typename VARCHAR2(30)  
)
```

10.f15\_moviewithType(movieid,typeid)

Primary key: movieid,typeid

This schema corresponds to the MoviewithType relation in our ER diagram. "movieid" and "typeid" together are the primary key. The "movieid" is the foreign key referencing "id" in f15\_movie. The "typeid" is the foreign key referencing "id" in f15\_movieType.

FD: none,

This schema is BCNF, since there is no FD.

```
CREATE TABLE f15_moviewithType (  
movieid NUMBER(4) REFERENCES f15_movie(id),  
typeid NUMERB(4) REFERENCES f15_movietype(id),  
CONSTRAINT moviewithType_PK PRIMARY KEY( movieid,typeid)  
)
```

11.f15\_movieReview(id, movieid, rating,content,time,userid)

Primary key: id

Foreign key: movieId references id in f15\_movie  
                  userId references id in f15\_user

This schema corresponds to the MovieReviewrelationship in our ER diagram. "id" is the primary key. "userId" is the foreign key referencing "id" in f15\_user. "movieId" is the foreign key referencing "id" in f15\_movie.

FD:  $id \rightarrow \text{movieid, rating,content,time,userid}$

This schema is BCNF, the FD satisfy BCNF, since "id" is key, so "id" is also superkey.

```
CREATE TABLE f15_movieReview(  
id NUMBER(4) PRIMARY KEY,  
movieId NUMBER(4) REFERENCES f15_movie(id),  
userId NUMBER(4) REFERENCES f15_user(id),  
rating NUMBER(4) NOT NULL,  
content VANCHAR2(1000) NOT NULL,  
time DATE  
)
```

12 f15\_moviecomment(id,reviewid,userid,content,time)

Primary key: id

Foreign key: reviewid references id in f15\_moviereview

Userid references id in f15\_user

This schema corresponds to the moviecomment entity in our ER diagram. "id" is the primary key. "userId" is the foreign key referencing "id" in f15\_user. "reviewId" is the foreign key referencing "id" in f15\_moviereview.

FD: id → reviewid,userid,content,time

This schema is BCNF, the FD satisfy BCNF, since "id" is key, so "id" is also superkey.

```
CREATE TABLE f15_moviecomment (  
id NUMBER(4) PRIMARY KEY,  
reviewid NUMBER(4) REFERENCES f15_moviereview(id),  
userid NUMBER(4) REFERENCES f15_user(id),  
content VARCHAR2(1000) NOT NULL,  
time DATE  
)
```

13.f15\_Theater(id, address,ticketcount)

Primary key: id

Candidate key: address

In our ER diagram, this schema corresponds to the Theatre entity. The "id" is the primary key. "address" is the candidate key.

FD: id → address, ticketcount

address → id, ticketcount

This schema is BCNF, the FD satisfy BCNF, since "id" and "address" is key, so "id" and "address" is also superkey.

```
CREATE TABLE f15_theatre(  
id NUMBER(4) PRIMARY KEY,  
address VARCHAR2(100) UNIQUE,  
ticketcount NUMBER(4)  
)
```

14.f15\_theaterReview(id, theaterId,content, memberId, rating,time)

Primary key: id

Foreign key: theaterId references id in f15\_theater

memberId references id in f15\_member

This schema corresponds to the TheatreReviewrelationship in our ER diagram. "id" is the primary key. "memberId" is the foreign key referencing "id" in f15\_member. "theatreId" is the foreign key referencing "id" in f15\_theatre.

FD: id → theaterId, content, memberId, rating,time

This schema is BCNF, the FD satisfy BCNF, since "id" is key, so "id" is also superkey.

```
CREATE TABLE f15_theaterReview (  
id NUMBER(4) PRIMARY KEY,  
theatreId NUMBER(4) REFERENCES Theatre(id),
```

```
memberId NUMBER(4) REFERENCES Member(id),
rating NUMBER(4) NOT NULL,
content VARCHAR2(1000) NOT NULL
time DATE
)
```

15.f15\_screensRoom(id,theatreId,roomNum, capacity)

Primary key: id

Candidate key:theatreId,roomNum

This schema corresponds to the ScreensRoom entity in our ER diagram. “id” is the primary key. “theatreId” is the foreign key referencing “id” in Theatre.

FD: id→theatreId,roomNum, capacity

This schema is BCNF, the FD satisfy BCNF, since “id” is the super key.

```
CREATE TABLE f15_screensRoom (
Id NUMBER(4) primary key,
theatreId NUMBER(4) REFERENCES f15_theatre(id),
roomNum NUMBER(2) NOT NULL,
capacity NUMBER(4) NOT NULL,
)
```

16.f15\_movieSchedule(id, screeningRoomNo, movieId, time,price)

Primary key: id

Foreign key: screeningRoomNo references roomNum in f15\_screensRoom  
movieId references id in f15\_movie

This schema corresponds to the MovieSchedulerelationship in our ER diagram. “id” is the primary key. “screenRoomNo” is the foreign key referencing “id” in f15\_screensRoom, “movieId” is the foreign key referencing “id” in f15\_movie.

FD: id→screeningRoomNo, movieId,time,price

screeningRoomNo, movieId→id, time, price

This schema is BCNF, both of the two FD satisfy BCNF, since “id” is key, so “id” is also superkey, and “screeningRoomNo, movieId” is the superkey, because together these three attributes determine all the other attributes.

```
CREATE TABLE f15_movieSchedule(
id NUMBER(4) PRIMARY KEY,
memberId NUMBER(4) REFERENCES Member(id),
screeningRoomNo NUMBER(2)REFERENCES ScreensRoom(roomNum),
timeDATE NOT NULL,
price NUMBER(4)
)
```

17.F15\_Ticket(id, price, userid, scheduleID)

Primary key: id

Foreign key: scheduleID references id in f15\_movieSchedule

userid references id in f15\_user

This schema corresponds to the Ticket entity in our ER diagram. “id” is the primary key. “userid” is the foreign key referencing “id” in f15\_user. “scheduleId” is the foreign key referencing “id” in f15\_movieSchedule.

FD: id → userid, scheduleId, price  
userid, scheduleId → id, price

This schema is BCNF, both of the two FD satisfy BCNF, since “id” is key, so “id” is also superkey, and “userid, scheduleId,” is the superkey, because together these two attributes determine all the other attributes.

```
CREATE TABLE f15_ticket(  
id NUMBER(4) PRIMARY KEY,  
userid NUMBER(4) REFERENCES f15_user(id),  
scheduleId NUMBER(4) REFERENCES f15_movieSchedule(id),  
price NUMBER(6,2) NOT NULL  
)
```

18.f15\_staff(ssn, userid)

Primary key: ssn

Foreign key: userid references id in f15\_user

This schema corresponds to the Staff entity in our ER diagram. “ssn” is the primary key. The “userid” is the foreign key referencing “id” in f15\_user. It is a subclass of User entity, and we use the E/R-Style Conversion approach to deal with the subclass structures, so Staff entity also has some hidden attributes that are connected by foreign key “userid” with User entity——name, phone, address, password.

FD: ssn → userid

This schema is BCNF, the FD satisfy BCNF, since “ssn” is key, so “ssn” is also superkey.

```
CREATE TABLE f15_staff(  
ssn NUMBER(9) PRIMARY KEY,  
userid NUMBER(4) REFERENCES User(id)  
)
```

19.f15\_jobType(job, description)

Primary key: job

This schema corresponds to the jobType entity in our ER diagram. “job” is the primary key.

FD: job → description

This schema is BCNF, the FD satisfy BCNF, since “job” is key, so “job” is also the superkey.

```
CREATE TABLE f15_jobType(  
job VARCHAR2(30) PRIMARY KEY,  
description VARCHAR2(1000) NOT NULL  
)
```



20.f15\_enrollment(employee, theatreid,job,workingtimestart,workingtimeend)

Primary key: employee

Foreign key: job references job in f15\_jobType

employee references ssn in f15\_staff

theaterId references id in f15\_theater

This schema corresponds to the Enrollment entity in our ER diagram. “employee,” is the primary key. “employee” is the foreign key referencing “ssn” in f15\_staff. “job” is the foreign key referencing “job” in f15\_jobType. “theatreId” is the foreign key referencing “id” in f15\_theatre.

FD: employee → theatreid, job, workingtimestart, workingtimeend

This schema is BCNF, the FD satisfy BCNF, since “employee” is the key, so “employee” is the superkey.

```
CREATE TABLE f15_enrollment(  
employee NUMBER(9) primary key,  
theatreId NUMBER(4) REFERENCES f15_theatre(id),  
job VARCHAR2(30) REFERENCES f15_jobType(job),  
workingtimestart DATE NOT NULL,  
workingtimeend DATE NOT NULL,  
CONSTRAINT time_CK check(workingtimestart<=workingtimeend)  
)
```

## Function and Ownership

Yiwei Zhao:

1. JDBC connect
2. create web pages and JSP
3. functions and requirements analysis
4. business logic analysis

Kunyang Zhao:

1. Database manage
2. sql statement control
3. tables design
4. handle SQL exception from application

Shengyuan Ye:

1. documents manage
2. create E/R diagrams
3. analysis FD and BCNF, 3NF
4. application test

## **Conclusion**

After discussion the detail of schemas and relationship between them, it is clarified that the relevant keys and functional dependencies of each schema and those schemas can support business logic and fully fulfill functions this project should have. The required queries are added as functionalities. Project is fully tested, detail test information is given in the demo test document.