Normalization Assignment

1. For the big university database table, list FDs with the column *StdCity* as the determinant that are not true due to the sample data. For each FD, you should identify one falsification example or indicate that no falsification example exists in the sample rows. Remember that it takes two rows to falsify an FD in which the LHS is the same in both rows but the RHS is different.

Table 1: Sample Data for the Big University Database Table

StdNo	StdCity	StdClass	OfferNo	OffTerm	OffYear	EnGrade	CourseNo	CrsDesc
S1	SEATTLE	JUN	O1	FALL	2017	3.5	C1	DB
S1	SEATTLE	JUN	O2	FALL	2017	3.3	C2	VB
S2	BOTHELL	JUN	O3	SPRING	2018	3.1	C3	OO
S2	BOTHELL	JUN	O2	FALL	2017	3.4	C2	VB

Solution:

stdcity FDs and sample rows that falsify the FDs. The rows refer to the sample data above.

FD	Falsifications
StdCity ® OfferNo	(1,2), (3,4)
StdCity ® OffTerm	(3,4)
StdCity ® EnGrade	(1,2), (3,4)
StdCity ® CourseNo	(1,2), (3,4)
StdCity ® CrsDesc	(1,2), (3,4)
StdCity ® OffYear	(3,4)
StdCity ® StdNo	None
StdCity ® StdClass	None

2. You should add one or more sample rows to Table 1 for the FDs in problem 1 with no falsification example. For each non-falsified FD with *StdCity* as the determinant, you should add one or more sample rows and then identify a falsification example involving the new row(s) and rows from Table 1.

Solution:

FDs with *stdcity* as the determinant not falsified by Table 1
are listed below along with a reference to new rows (after row 4) that falsify the FDs.

stdcity ® stdno is falsified by two pairs of rows: <1,5> and <2,5>

stdcity ® stdclass is falsified by two pairs of rows: <1,5> and <2,5>

StdNo	StdCity	StdClass	OfferNo	OffTerm	OffYear	EnGrade	CourseNo	CrsDesc
S1	SEATTLE	JUN	O1	FALL	2013	3.5	C1	DB
S1	SEATTLE	JUN	O2	FALL	2013	3.3	C2	VB
S2	BOTHELL	JUN	O3	SPRING	2014	3.1	C3	OO
S2	BOTHELL	JUN	O2	FALL	2013	3.4	C2	VB
S3	SEATTLE	SR	O1	FALL	2014	3.3	C1	DB

3. Convert the ERD in Figure 1 into tables and perform further normalization as needed. Your conversion result should show the list of tables with primary keys, foreign keys, and not null constraints. After converting the ERD to tables, specify FDs for each table. Since the primary key of each table determines the other columns, you should only identify FDs in which the LHS is not the primary key. If a table is not in BCNF, explain why and split it into two or more tables that are in BCNF.
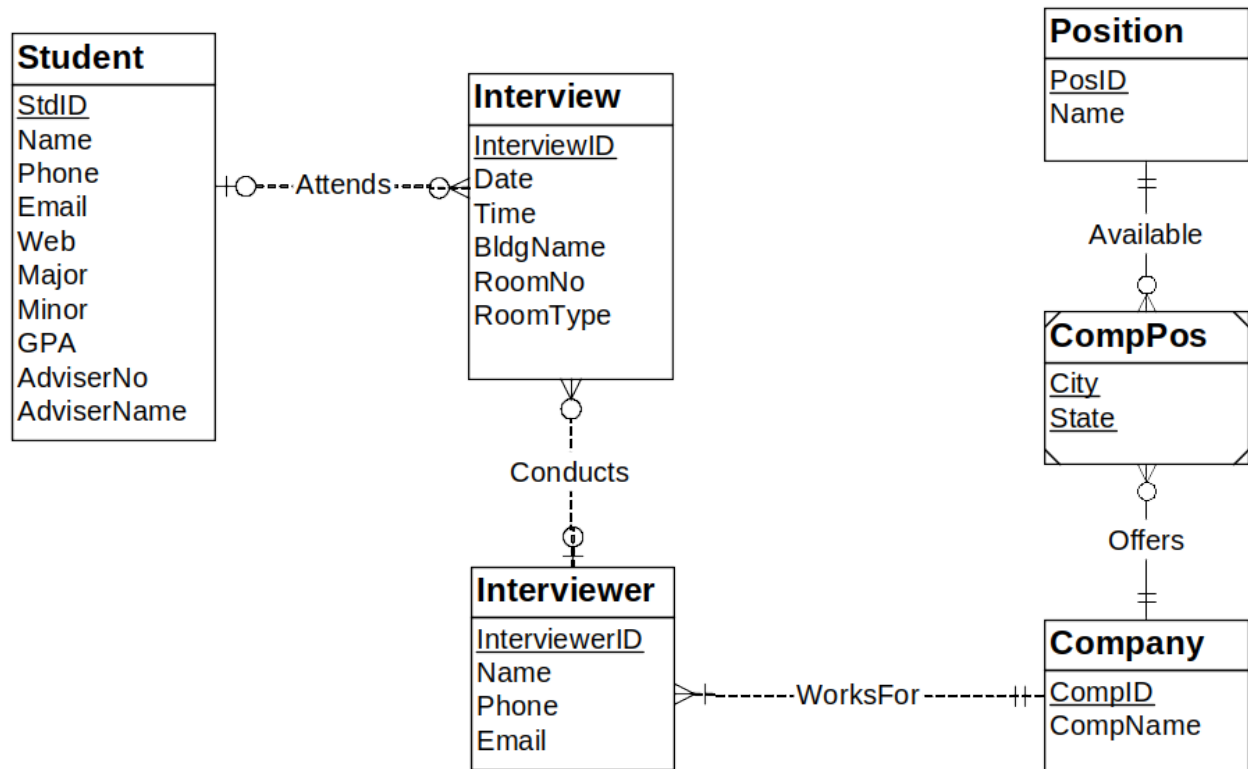
Figure 1: ERD for Problem 3

Solution:

Student(<u>StdId</u>, Name, Email, Phone, Web, Major, Minor, GPA, AdviserNo, AdviserName)

Interview(<u>Interviewid</u>, BldgName, RoomNo, RoomType, Date, Time, *StdId*, *InterviewerId*)

FOREIGN KEY(InterviewerId) REFERENCES Interviewer

FOREIGN KEY(StdId) REFERENCES Student

Interviewer(<u>InterviewerId</u>, Name, Phone, Email, *CompId*)

FOREIGN KEY(CompId) REFERENCES Company

CompId NOT NULL

Company(CompId, CompName)

Position(PosId, Name)

CompPos(CompId, PosId, City, State)

FOREIGN KEY(CompId) REFERENCES Company

FOREIGN KEY(PosId) REFERENCES Position

4. Apply the simple BCNF procedure to the following list of FDs. Show the result of each step. In the final list of tables, you should show the primary keys, foreign keys, and unique constraints. You do not need to write CREATE TABLE statements.

AuthNo AuthName

AuthEmail AuthNo

PaperNo Primary-AuthNo

AuthNo AuthAddress

AuthNo AuthEmail

PaperNo PapTitle

PaperNo PapAbstract

PaperNo PapStatus

RevNo RevName

RevNo RevEmail

RevEmail RevNo

RevNo, PaperNo Auth-Comm

RevNo, PaperNo Prog-Comm

RevNo, PaperNo RevDate

RevNo, PaperNo Rating

RevNo RevAddress

Step 1: Arrange the remaining FDs into groups by determinant

AuthNo AuthName, AuthEmail, AuthAddress

AuthEmail AuthNo

PaperNo Primary-AuthNo, Title, Abstract, Status

RevNo RevName, RevEmail, RevAddress

RevEmail RevNo

RevNo, PaperNo Auth-Comm, Prog-Comm, Date, Rating1, Rating2, Rating3,

Step 2: For each FD group, make a table with the determinant as the primary key. In the table list, the primary keys are underlined.

Author(<u>AuthNo</u>, AuthName, AuthEmail, AuthAddress)

FOREIGN KEY (AuthEmail) REFERENCES Author

AuthEmail(AuthEmail, AuthNo)

FOREIGN KEY (AuthNo) REFERENCES Author

Paper(<u>PaperNo</u>, Primary-AuthNo, Title, Abstract, Status)

FOREIGN KEY (Primary-AuthNo) REFERENCES Author

Reviewer(<u>RevNo</u>, RevName, RevEmail, RevAddress)

FOREIGN KEY (RevEmail) REFERENCES ReviewerEmail

ReviewerEmail(<u>RevEmail</u>, <u>RevNo</u>)

FOREIGN KEY (RevNo) REFERENCES Reviewer

Review(<u>PaperNo, RevNo</u>, Auth-Comm, Prog-Comm, Date, Rating1, Rating2, Rating3,

Rating4, Rating5)

FOREIGN KEY (PaperNo) REFERENCES Paper

FOREIGN KEY (RevNo) REFERENCES Reviewer

Step 3: Merge tables with the same columns. The Author and AuthEmail tables are merged. The Reviewer and ReviewerEmail tables are merged. UNIQUE constraints are added for AuthEmail and RevEmail.

Solution:

Author(<u>AuthNo</u>, AuthName, AuthEmail, AuthAddress)

UNIQUE (AuthEmail)

Paper(<u>PaperNo</u>, Primary-AuthNo, Title, Abstract, Status)

FOREIGN KEY (Primary-AuthNo) REFERENCES Author

Reviewer(<u>RevNo</u>, RevName, RevEmail, RevAddress)

UNIQUE (RevEmail)

Review(<u>PaperNo, RevNo</u>, Auth-Comm, Prog-Comm, Date, Rating1, Rating2, Rating3,

Rating4, Rating5)

FOREIGN KEY (PaperNo) REFERENCES Paper

FOREIGN KEY (RevNo) REFERENCES Reviewer