CSC460

Dr. McCann

TA Shuo Yang

Analysis of Normal Forms and Functional Dependencies

Relation: Office(OfficeID, city, mgrID)

FDs: OfficeID -> city, mgrID

1nf: No attributes are set-valued

2nf: city, mgrID are FD on OfficeID

3nf: OfficeID is a superkey

Relation: Employee(IdNo, firstName, middleName, lastName, Salary)

FDs: IdNo -> firstName, middleName, lastName, salary

1nf: No attributes are set-valued

2nf: firstName, middleName, lastName, and Salary, are FD on IdNo

3nf: IdNo is a superkey

Relation: Car(CarID, licensePlate, assignedEmp)

FDs: CarId -> LPlate, assignedEmp

The reason LPlate is not a candidate key is because if the car is from a different region, two cars potentially can have the same license plate value temporarily

1nf: No attributes are set-valued

2nf: LPlate, assignedEmp are FD on CarId

3nf: CarId is a superkey

Relation: Client(ClientId, firstName, middleName, lastName, dob, registered, provisionNum, requestedInstructor, assignedInstructor)

FDs: ClientId -> firstName, middleName, lastName, dob, registered, provisionNum, requestedInstructor, assignedInstructor

1nf: No attributes are set-valued

2nf: firstName, middleName, lastName, dob, registered, provisionNum, requestedInstructor, assignedInstructor are FD on ClientId

3nf: ClientId is a superkey

Relation: Interview(Interviewer, time, Client)

FDs: Interviewer, time -> Client

The reason time is part of the primary key is just in case a client needs to reschedule last minute and thus there could temporarily be two interviews scheduled for the same person (may need to bill client for both meetings because of the last minute change and the instructor not being able to meet with anyone at that first time)

1nf: No attributes are set-valued

2nf: Client is FFD on Interviewer, time because each interviewer can have multiple interviews and there can be multiple interviews at the same time

3nf: Interviewer, time is a superkey

Relation: Role(empId, role, officeNo)

FDs: No non-trivials

The reason role is part of the primary key is because as stated in the description, an employee can assume multiple roles. In each office, there can be multiples of the same role. Also, by having officeNo part of the primary key, employees can take roles in multiple offices (temporarily for a transfer, or if we need one manager to watch two offices since a manager at an office quit suddenly).

1nf: No attributes are set-valued

2nf: No non-prime attributes

3nf: No non-trivial FDs.

Relation: Need(Instructor, Client, Description)

FDs: No non-trivials

The reason Description is part of the primary key is because an instructor could record multiple needs for a client

1nf: No attributes are set-valued

2nf: No non-prime attributes

3nf: No non-trivial FDs

Relation: Inspection(InsId, instDate, Inspector, CarId)

FDs: InsId-> CarId, instDate, Inspector

CarId, instDate -> Inspector, InsId

Inspector, instDate -> CarId, InsId

The reason for adding an InsId in this case is so that every Fault stores less information

1nf: No attributes are set-valued

2nf: No non-prime attributes (CKs are: {InsId}, {CarId, instDate}, {Inspector, instDate}

3nf: For all FDs, the left-hand side is a superkey

Relation: Fault(InsId, Description)

FDs: No non-trivials

The reason Description is part of the primary key is because an Inspection can have multiple Faults

1nf: No attributes are set-valued

2nf: No non-prime attributes

3nf: No non-trivial FDs

Relation: Lesson(LessonId, Instructor, Client, time, isBlocked, milesDriven)

FDs: LessonID-> Instructor, Client, time, isBlocked, milesDriven

Instructor, time -> Client, isBlocked, milesDriven, LessonID

Client, time -> Instructor, isBlocked, milesDriven, LessonID

The reason for adding a LessonID in this case is so that every Note stores less information

1nf: No attributes are set-valued

2nf: isBlocked and milesDriven are FFD on every CK since they are on the right side of every FD and removing any attribute from any of the left-hand sides in the FDs listed will break the dependency (CKs are: {LessonID}, {Client, time}, {Instructor, time}

3nf: For all FDs, the left-hand side is a superkey

Relation: Note(LessonId, Note)

FDs: No non-trivials

The reason Note is part of the primary key is because a Lesson can have multiple Notes

1nf: No attributes are set-valued

2nf: No non-prime attributes

3nf: No non-trivial FDs

Relation: DrivingTest(TestId, Instructor, Client, time, testType, pass)

FDs: TestId-> InstructorId, Client, time, testType, pass

InstructorId, time -> Client, testType, pass, TestId

Client, time -> Instructor, testType, pass, TestId

The reason for adding a TestId in this case is so that every Failure stores less information

1nf: No attributes are set-valued

2nf: testType and pass iare FFD on every CK since they are on the right side of every FD and removing any attribute from any of the left-hand sides in the FDs listed will break the dependency (CKs are: {TestId}, {InstructorId, time}, {Client, time}

3nf: For all FDs, the left-hand side is a superkey

Relation: Failure(TestId, Description)

FDs: No non-trivials

The reason Description is part of the primary key is because a Test can have multiple Failures

1nf: No attributes are set-valued

2nf: No non-prime attributes

3nf: No non-trivial FDs