ISTE-230 Introduction to Database & Data Modeling

## Practice Exercise # 5 – Normalization through 2NF

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**All assignments will be graded with regard to the standards that were discussed in class, which can be found in the Standards Content area.**

For each problem below, given the original relation and functional dependencies, normalize the original and all resulting relations to 2NF. Be sure to use proper relational notation: RELATION(pkattr, attribute, *fkattr*). Include reference statements for foreign keys.

**Problem #1**



EMPLOYEE2(empID, name, deptName, salary, courseTitle, dateCompleted)

Functional Dependencies:

empID, courseTitle 🡺 name, deptName, salary, dateCompleted

empID 🡺 name, deptName, salary

**YOUR ANSWER (Final set of relations normalized to 2NF):**

Employee(empID, name, deptname, salary)

Course(courseTitle, dateCompleted, *empID*)

Course(empID) must exist in Employee(empID)

**Problem #2**

ENGINEER\_SERVICE(empID, firstName, lastName, email, serviceID, serviceName)

Functional Dependencies:

empID, serviceID 🡺 firstName, lastName, email, serviceName

empID 🡺 firstName, lastName, email

email 🡺 empID, firstName, lastName

serviceID 🡺 serviceName

**YOUR ANSWER (Final set of relations normalized to 2NF):**

Engineer(empID, firstName, lastName, email)

Service(serviceID, serviceName)

EngineerService(*serviceID*, *empID*)

EngineerService(serviceID) must exist in Service(serviceID)

EngineerService(empID) must exist in (Engineer(empID)

**Problem #3**



MOVIE(title, year, length, type, studio, star)

Functional Dependencies:

title, star 🡺 year, length, type, studio

title 🡺 year, length, type, studio

**YOUR ANSWER (Final set of relations normalized to 2NF):**

Movie(title, year, length, type, studio)

MovieStar(star, *title)*

MovieStar(title) must exist in Movie(title)

**Problem #4**

APPOINTMENT(clientID, providerID, apptDate, startTime, endTime, firstName, lastName, notes, street, city, state, zipcode, phone, fName, lName,cellNum, serviceID, serviceName, price, duration, description, email)

Functional Dependencies:

clientID, providerID, apptDate, serviceID, startTime 🡺 endTime, firstName, lastName, notes, street, city, state, zipcode, phone, fName, lName, cellNum, serviceName, price, duration, description, email

clientID 🡺 firstName, lastName, street, city, state, zipcode, phone, email

email 🡺 clientID, firstName, lastName, street, city, state, zipcode, phone

providerID, serviceID 🡺 price

providerID 🡺 fName, lName, cellNum

serviceID 🡺 serviceName, duration, description

**YOUR ANSWER (Final set of relations normalized to 2NF):**

Client(ClientID, firstName, lastName, street, city, state, zipcode, phone, email)

Provider(providerID, firstName, lastName, cellNum)

Service(serviceID, serviceName, duration, description)

Appointments(apptDate, startTime, endTime, notes, price, *clientID*, *providerID*, *serviceID*)

Appointments(clientID, providerID, serviceID) must exist in Client(ClientID), Provider(providerID), Service(serviceID)