Project 4 CS60 Fall 2017

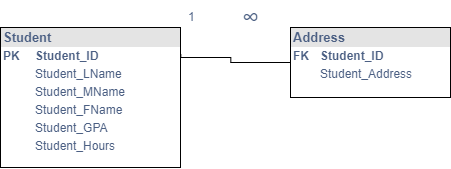
Designing normalized tables—Part 1 Redesigning Tables with First Normal Form Faults

Due Tuesday October 23 at 10:30pm

**Design 1:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE NAME** | COLUMN NAME Table and column names must satisfy naming rules listed on the first page, including no spaces or hyphens | **KEY** | For a foreign key, Tablename.Columnname it references. |
| ? | STUDENT FULLNAME  (This column stores 3 names in one column, a 1NF composite fault. What are you going to use for a primary key?) |  |  |
|  | STUDENT GPA  (For convenience, the student’s overall GPA is included in this table.) When would you update this column so the GPA is accurate? |  |  |
|  | STUDENT HOURS  (For convenience, this column stores the total number of credit hours completed by the student.) |  |  |
|  | STUDENT ADDRESS 1  (This column stores Street number, Street name, City, State, ZIP in one column; 1NF composite fault.) |  |  |
|  | STUDENT ADDRESS 2 (now the composite fault also becomes another 1NF fault: repeating groups of similar information.) |  |  |

**Corrections for the 1NF faults in Design 1 above:** Use unique names for your tables. In each table, use unique names for your column names. You can copy and paste the above Microsoft table and edit it to correct the faults: Delete the comments within parentheses here and in later designs.

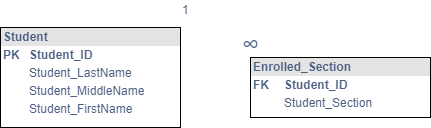
****

**FK Address.Student\_ID refers to Student.Student\_ID**

**Design 2:** I carried the Fullname column fault from Design 1 into this table and the later ones as well. In five separate columns, this table stores up to five sections that the student is enrolled in at the present time. Delete the comments within parentheses. The group that repeats has one column. In Design 6, the group that repeats has six columns.

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE NAME** | COLUMN NAME | **KEY** | Tablename.Columnname referenced by any foreign key |
|  | STUDENT FULLNAME  (1NF composite fault because it stores 3 names of a student in one column. You can copy and paste your solution from Design 1 and set up a good primary key.) |  |  |
|  | SECTION 1 |  |  |
|  | SECTION 2 |  |  |
|  | SECTION 3 |  |  |
|  | SECTION 4 |  |  |
|  | SECTION 5 |  |  |

**Corrections for the 1NF faults in Design 2 above:**

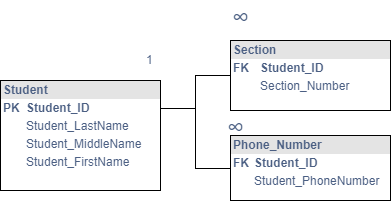
****

**FK Enrolled\_Section.Student\_ID refers to PK Student.Student\_ID**

**Design 3: Repeating groups of similar data arising from multi-valued attributes.**

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE NAME** | COLUMN NAME | **KEY** | Tablename.Columnname referenced by any foreign key |
|  | STUDENT FULLNAME  (This column stores 3 names in one column, a 1NF composite fault.) |  |  |
|  | SECTION  (This one column stores the Section numbers for up to 5 courses that the student is enrolled in. This is a 1NF fault with a multi-valued attribute.) |  |  |
|  | STUDENT PHONE  (This column stores 3 phone numbers for a student, another 1NF multi-valued fault.) |  |  |

**Corrections for the 1NF faults in Design 3 above:**

****

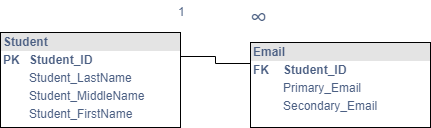
**FK Section.Student\_ID refers to PK Student.Student\_ID**

**FK Phone\_Number.Student\_ID refers to PK Student.Student\_ID**

**Design 4:**

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE NAME** | COLUMN NAME | **KEY** | Tablename.Columnname referenced by any foreign key |
|  | STUDENT FULLNAME  (This column stores 3 names in one column; 1NF composite fault.) |  |  |
|  | STUDENT E-MAIL  (This is the preferred e-mail.  This violates the principle that the order of columns is arbitrary.) |  |  |
|  | STUDENT E-MAIL  (This is an alternate e-mail to be used only after trying the preferred e-mail above. This violates the 1NF rule that the column order be arbitrary. Because you have two e-mails, you also have a repeating group of similar information. This column name also violates the 1NF rule that the column names in each table be unique.) |  |  |

**Corrections for the 1NF faults in Design 4 above:**

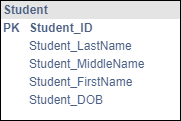
****

**FK Email.Student\_ID referes to PK Student.Student\_ID**

**Design 5: Violation of the rule that the order of rows and columns be arbitrary.**

|  |  |  |  |
| --- | --- | --- | --- |
| **TABLE NAME** | COLUMN NAME | **KEY** | Tablename.Columnname referenced by any foreign key |
|  | STUDENT FULLNAME  (This column stores the 3 names in one column, and stores the rows of students in an order from youngest age to oldest. This is a violation of the rule that the order of rows and columns stored in a table be arbitrary, or that the row order or column order convey no additional information.) |  |  |

**Corrections for the 1NF faults in Design 5 above:**

****

**Design 6. This table stores data about one or two cars owned by an employee.** Eliminate these violations of the first normal form. In this case, the group of columns that repeats has six columns.

**Employee**

|  |
| --- |
| **Employee\_ID** |
| Employee\_Lastname |
| Employee\_Firstname |
| Employee\_Middlename |
| VIN1 |
| LicensePlateNumber1 |
| Manufacturer1 |
| Model1  Two cars owned by an employee (An auto’s VIN is the Vehicle Identification Number) |
| Year1 |
| Color1 |
| VIN2 |
| LicensePlateNumber2 |
| Manufacturer2 |
| Model2 |
| Year2 |
| Color2 |

**Corrections for the 1NF faults in Design 6 above:**

****

FK Primary\_Automobile.Employee\_ID refers to PK Employee.Employee\_ID

FK Secondary\_Automobile.Employee\_ID refers to PK Employee.Employee\_ID