CS60/CS61 Project 0 Fall 2012 Primary and foreign keys

Nothing needs to be submitted or will be graded; we’ll discuss the keys in class.

**Rules for Primary Keys:**

1a. Primary keys have unique and not null values.

1b. If the primary key has more than one column (referred to as a **composite** or **multi-valued** **primary key**), what you mean by *unique* values is that the combination of values be unique.

1c. A table can only have one primary key.

**Rules for Foreign Keys:**

2a. A foreign key references a primary key or column(s) with unique constraints (but referencing unique column(s) is not further considered in CS60).

2b. The datatype of the foreign key must be the same as the datatype for the primary key it references.

2c. The table and primary key that the foreign key references must exist at the time you set up the foreign key.

2d. The values in the foreign key are restricted to the set of values in the primary key it references, or the foreign key can be NULL. If you don’t want to allow a NULL in the FK, you’ll need to set a separate NOT NULL constraint. A foreign key constraint by itself allows a NULL

2e. Foreign keys can have one column or more than one. If more than one, they’re referred to as **composite or multi-valued foreign keys**. They will reference primary keys with the same number of columns as the foreign key. A two-column foreign key cannot reference a one-column primary key, nor can a one-column foreign key reference a two-column primary key. If a foreign key has more than one column, there is an order to those columns when the key is set up, and the order will match the order of any multi-column primary key.

2e. The column name(s) of the foreign key are usually the same as the column name(s) of the referenced primary key, but having the same names isn’t a requirement for FKs to reference PKs.

**Data types useful in this project:**

char(n) is a fixed-length string of 8-bit ASCII characters with length n.

varchar2(n) is a variable-length string of 8-bit ASCII characters with a maximum length of n-characters.

Number(n) is a whole number with a maximum of n decimal digits. The decimal digits are separately stored in about 4 bits each. “about” is used because additional rules arise for whole numbers that end in zeros, like 1000000.

1. The following Student table is empty and has a name, the column storing the Student ID has a name and a datatype. Can you make the Student\_ID the primary key? If not, why not?

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First Datatype |
| varchar2(7) | Varchar2(20) | Varchar2(20) |

2. The following table has a primary key as noted. Can you insert the first row? If not, why not? By a null value, I mean that no value has been assigned.

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| Null value | Jones | Jack Datatype  Constraint(s) |
| varchar2(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

3. The following table has a primary key as noted. Can you insert the first row? If not, why not?

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| 1234567 | Null value | Jack Datatype  Constraint(s) |
| varchar2(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

4. The following table has a primary key as noted, and the 1st row has been inserted. Can you insert the 2nd row of raw data? If not, why not?

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| 1234567 | Yi | Jack |
| 1234567 | Sanchez | Antonio Datatype  Constraint(s) |
| varchar2(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

5. The following table enrolls students by recording their student ID and three things about the section of a course they’re enrolling in: The Department\_Name, Course\_Number, and Section\_Letter (this is not how we record a section at SMC). Can you insert the first row? If not, why not?

Enroll

|  |  |  |  |
| --- | --- | --- | --- |
| Student\_ID | Department\_Name | Course\_Number | Section\_Letter |
| 1234567 | Math | 101 | A |
| varchar2(7) | Varchar2(20) | Varchar2(4) | Char(1) |
| Primary Key | Primary Key | Primary Key | Primary Key |

4-column primary key

6. The following table is described above. Can you insert the 2nd and 3rd rows shown below? If not, why not?

Enroll

|  |  |  |  |
| --- | --- | --- | --- |
| Student\_ID | Department\_Name | Course\_Number | Section\_Letter |
| 1234567 | Math | 101 | A |
| 1234567 | Math | 201 | A |
| 2345678 | Math | 101 | A |
| varchar2(7) | Varchar2(20) | Varchar2(4) | Char(1) |
| Primary Key | Primary Key | Primary Key | Primary Key |

4-column primary key

7. The following table has been created without any keys. Based on the raw data, what column or combination of columns could be a primary key?

PURCHASE **table**

|  |  |  |
| --- | --- | --- |
| Column\_1 | Column\_2 | Column\_3 |
| A | A | B |
| B | NULL | A |
| A | B | B |
| C | E | C |

8. The following table is SMC’s way to handle enrollment. The table is empty (the column names and datatypes are actually stored elsewhere in the database). **No other tables exist**. Can you make the Student\_ID column be a foreign key? If not, why not?

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| varchar2(7) | char(4) |

9. Now I’m adding a 2nd table. Both tables are empty. Can I get Student\_ID in Enroll table to reference Student\_ID in Student? If not, why not?

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| varchar2(7) | char(4) |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First Datatype |
| Number(7) | Varchar2(20) | Varchar2(20) |

10. I’ve changed a datatype. Now can Student\_ID in Enroll reference Student\_ID in Student? If not, why not?

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| Varhar2(7) | char(4) |
| FK? |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First Datatype  Constraints |
| Varchar2(7) | Varchar2(20) | Varchar2(20) |
|  |  |  |

11. Student table now has a PK shown. Can Student\_ID in Enroll reference Student\_ID in Student? If not, why not?

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| Varchar2(7) | Varchar2(20) |
| FK? |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| Varchar2(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

12. I’m changing the name of Student\_ID in Enroll to be ID. Can column ID in Enroll reference Student\_ID in Student?

Enroll

|  |  |
| --- | --- |
| ID | Section\_Number |
| Char(7) | Varchar2(20) |
| FK? |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| Char(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

13. I’m changing the name of ID back to Student\_ID in Enroll. Student\_ID in Enroll is now a foreign key to reference Student\_ID in Student.

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| Varchar2(7) | char(4) Can this 1st row in Enroll be inserted? |
| 1234567 | 4119 |
| FK |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| Char(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

14. Same tables as before. Can the 1st row in Enroll be inserted?

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| varchar2(7) | char(4) This row? |
| NULL | 4119 |
| Foreign Key |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First |
| Varchar2(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

15. Same tables as before except the row beginning with NULL does not appear in Enroll. I’ve added a row to Student.

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| Varchar2(7) | char(4) Can this row be inserted? If not, why not? |
| 234567 | 4119 |
| Foreign Key |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First This row has been inserted. |
| 1234567 | Yi | Yon |
| Char(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

16. Student table exists as shown below with one row in Student. Can the 1st row in Enroll be inserted?

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| varchar2(7) | char(4) Can this row be inserted? |
| NULL | 4119 |
| Foreign Key |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First This row has been inserted. |
| 1234567 | Yi | Yon |
| Varchar2(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

17. Student table exists as shown below with one row in the Student table. Enroll is empty.

Enroll

|  |  |
| --- | --- |
| Student\_ID | Section\_Number |
| Char(7) | char(20) Can this row be inserted? |
| 1234567 | 4119 |
| Foreign Key |  |

Student

|  |  |  |
| --- | --- | --- |
| Student\_ID | Student\_Last | Student\_First This row has been inserted. |
| 1234567 | Yi | Yon |
| Char(7) | Varchar2(20) | Varchar2(20) |
| Primary Key |  |  |

18. The Enroll table is the same as you saw earlier; it stores enrollment data in a school that identifies a section of a course by the three values: Department\_Name, Course\_Number, and Section\_Letter, and it identifies a course by two values: Department\_Name and Course\_Number. The Course table stores data about courses. A course can have no sections, one section, or many sections in a particular semester.

How many primary keys does the Enroll table have? How many primary keys does the Course table have?

Can the three columns Department\_Name , Course Number, and Section\_Letter in Enroll be a composite foreign key that references the composite primary key in Course?

Enroll

|  |  |  |  |
| --- | --- | --- | --- |
| Student\_ID | Department\_Name | Course\_Number | Section\_Letter |
| 1234567 | Math | 101 | A |
| 1234567 | Math | 201 | A |
| 2345678 | Math | 101 | A |
| Varchar2(7) | Varchar2(20) | Varchar2(4) | Char(1) |
| Primary Key | Primary Key | Primary Key | Primary Key |
|  | ? | ? | ? |

4-column primary key

Course

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Department\_Name | Course\_Number | Course\_Title | Course\_Description | Credit\_Hours |
| Math | 101 | Algebra I |  | 3 |
| Math | 201 | Calculus I |  | 3 |
| Math | 301 | Differential Equations I |  | 3 |
| Varchar2(7) | Varchar2(20) | Varchar2(35) | Varchar2(300) | Number(2,1) |
| Primary Key | Primary Key |  |  |  |

2-column primary key