Objective: Practice concepts of referential integrity constraints and foreign keys

Assignment: Answer each of the problems below.

Problem #1:
Consider the database state for the UNIVERSITY database shown below. The primary keys are underlined and the foreign keys (FK) are specified below. For each of the update operations (a)-(n), describe **all** integrity constraints that are violated by the operations. If no integrity constrains are violated, indicate that there are "no constraint violations". For each operation (a)-(n), assume that the database starts in the state shown below. For each violation, indicate the different ways of enforcing the constraint.

COURSES.dept is a FK referencing DEPT.abbr
DEPT.chair is a FK referencing FACULTY.pid
FACULTY.dept is a FK referencing DEPT.abbr
HAS_TAUGHT.pid is a FK referencing FACULTY.pid
HAS_TAUGHT.cnum is a FK referencing COURSES.ucnum

COURSES

ucnum	dept	dcnum	title
A4781	MATH	101	Intro to Calculus
B7288	MATH	317	Differential Equations
C9262	HIST	110	Early US History
C4196	HIST	318	Medieval Europe
D8017	CS	101	Intro to Programming
E7692	CS	285	Data Structures
F5293	CS	382	Algorithms

DEPT

abbr	name	chair	building
HIST	History	7788	Duff Hall
MATH	Math	2233	Skinner Hall
CS	Computer Science	3344	Burns Hall

FACULTY

pid	name	dept	specialty
1122	Homer Simpson	HIST	Renaissance Italy
2233	Marge Simpson	MATH	Calculus
3344	Bart Simpson	CS	Computer graphics
4455	Lisa Simpson	HIST	Early US History
5566	Maggie Simpson	HIST	Modern US History
6677	Smithers	MATH	String Theory
7788	Ned Flanders	HIST	Medieval Europe
8899	Krusty the Clown	CS	Bioinformatics
9900	Sideshow Bob	CS	Compiler optimization

HAS_TAUGHT

pid	cnum
1122	C9262
4455	C9262
2233	B7288
8899	E7692
1122	C4196
6677	A4781
9900	F5293
7788	C4196
3344	E7692

a) Insert <'G2485', 'HIST', '378', 'Modern US History'> into COURSES.
b) Insert <'H8451', 'INLS', '523', 'Database Systems'> into COURSES.
c) Insert <'5566', 'Barney Gumble', 'MATH', 'Geometry'> into FACULTY.
d) Insert <'5678', 'Kent Brockman', 'INLS', 'Archives'> into FACULTY.
e) Insert <'CHEM', 'Chemistry', NULL, 'Quimby Hall'> into DEPT.
f) Insert <NULL, 'MATH', '214', 'Linear Algegra'> into COURSES.
g) Insert <'9900', 'E7692'> into HAS_TAUGHT.
h) Insert <'9900', 'F5293'> into HAS_TAUGHT.
i) Insert <'7788', '4829'> into HAS_TAUGHT.
j) Delete the FACULTY tuple with pid='3344'.
k) Delete the COURSES tuple with ucnum='D8017'.
l) Delete the HAS_TAUGHT tuple with pid='1122' and cnum='C9262'.
m) Modify the chair attribute of the DEPT tuple with abbr='HIST' to be chair='4455'.
n) Modify the abbr attribute of the DEPT tuple with abbr='CS' to be 'COMPSCI'.

Problem #2:
Consider the following relations for a database about movies.

MOVIE (mpaa_num, title, runtime, production_co)
ACTOR (sag_num, fname, lname, bday)
STARS_IN (sag_num, mpaa_num, screen_time)
PRODUCTION_CO (tax_id_num, name, location)

For each movie an actor stars in, they have a certain amount of screen time. Each movie is produced by a production company. Production companies have a unique tax id number, a name, and an office location.

Specify the foreign keys for this schema. Clearly state any assumptions you make.

Grading:
I will be looking for accuracy, completeness, and clarity.

How to turn in your assignment:
Prepare your assignment using a word processor and drawing software. ALL assignments you submit should have your name, Onyen, homework number, and date at the top left of the first page.

Save your assignment into a SINGLE PDF file of no more than 3 pages total.

Name your file according to the following convention:

`youronyen-h2.pdf`

Replace *youronyen* with you actual Onyen (e.g. my assignment would be `rcapra-h2.pdf`). The character between *youronyen* and the "h2.pdf" part should be a single minus sign (not an underscore). You could also call this character a dash. There should be no spaces or other characters in the filename. Files with names that do not follow this convention will not be graded.

Submit your file electronically through the Sakai by going to the Assignments area and finding the "H2" assignment. After you think you have submitted the assignment, I strongly recommend checking to be sure the file was uploaded correctly and that you can view it by clicking on it from within Sakai. If I cannot open or view your file, I cannot grade it.

If for some reason you need to re-submit your homework file, you must add a version number to your filename so that we will know which file is the most recent. Sakai is configured so that it will only accept 3 total submissions. Use the following file naming convention if you need to re-submit:

Your first submission: `youronyen-h2.pdf`
Your second submission: `youronyen-h2-v2.pdf`
Your third submission: `youronyen-h2-v3.pdf`

Sakai is also configured with a due date and an "accept until" date. Submissions received after the due date (even just 1 minute!) will receive a 10% penalty per day. The "accept until" date is 5 days after the due date. Submissions will not be accepted after the "accept until" date and will have a score of zero recorded.