CS 325 - Homework 3

Deadline

11:59 pm on **Friday, September 24, 2021**.

Purpose

To see if you are gleaning some important concepts from required class reading, and to practice thinking about and writing relational operations, both "by hand" AND using SQL.

How to submit

Problems 1 and 2 will be completed on the course Canvas site.

For Problem 3 onward:

Each time you wish to submit, within the directory `325hw3` on nrs-projects.humboldt.edu (and at the nrs-projects UNIX prompt, **NOT inside** `sqlplus`!) type:

`~st10/325submit`

...to submit your current files, using a homework number of `3`.

(**Make sure** that the files you intend to submit are listed as having been submitted!)

Additional notes:

- You are required to use the HSU Oracle `student` database for **Problems 4 and 5** of this homework.

- **DB Reading Packet 3 and SQL Reading Packet 2,** on the course Canvas site, along with the posted in-class projections from the public course web site, are useful references for this homework.

- Feel free to add additional `prompt` commands to your SQL scripts as desired to enhance the readability of the resulting output.

Starting set of `select` style standards

- NOTE the following **course style standards** for SQL `select` statements:

  - In a SQL script, put a blank line BEFORE and AFTER each `select` statement, for better readability.

  - A `select` statement's `from` clause should ALWAYS start on a new line.

  - If a `select` statement has a `where` clause, that `where` clause should always start on a new line.

  - If a clause is longer than one line, INDENT the continuation on the next line by at least three spaces (so it is clear which clause it "belongs" to).

Problem 1

Correctly complete the "HW 3 - Problem 1 - Reading Questions for DB Reading Packet 3 - Intro to the

Relational Model", on the course Canvas site.

Problem 2

Correctly complete the "HW 3 - Problem 2 - Questions about relational operations", on the course Canvas site.

Setup for Problems 3 onward

Use `ssh` to connect to `nrs-projects.humboldt.edu`, and create, protect, and go to a directory named `325hw3` on nrs-projects:

```
mkdir 325hw3
chmod 700 325hw3
cd 325hw3
```

Put all of your files for this homework in this directory. (And it is from this directory that you should type `~st10/325submit` each time you would like to submit your files.)

Problem 3 - **NOTE! This problem does NOT use Oracle or SQL!*

(You are THINKING through this problem's parts from a purely relational algebra point of view! NO SQL, nothing being executed on a computer; you do these "in your head", "by hand"!)

Use `nano` (or `vi` or `emacs`) to create a file named `325hw3-3.txt` within directory `325hw3`:

```
nano 325hw3-3.txt
```

While within `nano` (or whatever), type in the following:

- your name
- `CS 325 - Homework 3 - Problem 3`
- the date this file was last modified

Then, answer each of the following questions, **preceding** each by the problem part it is the answer for.

When asked to give a relation in **tabular form** as your answer, do so by putting the attribute names on one line, a row of dashes , and then the attribute values neatly lined up underneath. For example:

```
stuff_id    stuff_name
---------------------
123         chocolate
124         otters
125         redwood trees
```

Consider the relations `Client`, `Video`, and `Rental` -- for this problem, assume that their current contents are:

the Client relation:

Cli_id	Cli_lname	Cli_fname	Cli_phone
000A	Alpha	Ann	000-0001
111B	Beta	Bob	111-1112
222B	Beta	Ann	222-2223
333C	Carlos	David	333-3334
444D	Delta	Edie	111-1112

the Video relation:

Vid_id	Vid_format	Vid_purchase_date	Vid_rental_price	Vid_length
00000D	DVD	11-JAN-2020	1.99	73
11111H	HD-DVD	22-FEB-2021	4.99	91
22222B	BluRay	03-MAR-2019	1.99	105
33333H	HD-DVD	22-FEB-2021	3.99	69
44444B	BluRay	04-APR-2017	0.99	91

the Rental relation:

Cli_id	Vid_id
111B	11111H
222B	00000D
222B	22222B
333C	22222B
333C	00000D
333C	11111H
000A	44444B

3 part a

Mentally perform a **relational projection** (by hand, NOT using SQL!) of the Vid_purchase_date and Vid_format attributes of Video, typing in **tabular form** the relation that should result into your file 325hw3-3.txt.

3 part b

Mentally perform a **relational selection** (by hand, NOT using SQL!) of the rows of Video for which the Vid_format is 'BluRay', typing in **tabular form** the relation that should result into your file 325hw3-3.txt.

3 part c

Mentally perform a **relational natural join** (by hand, NOT using SQL!) of the relations `Client` and `Rental` on the attribute `Cli_id`, (using the join condition `Client.Cli_id = Rental.Cli_id`), typing in **tabular form** the relation that should result into your file `325hw3-3.txt`. Note that you only need to give the **final** resulting relation, **not** the intermediate steps along the way.

Submit your file `325hw3-3.txt`.

Problem 4

YOU **ARE** USING ORACLE and SQL FOR THIS PROBLEM.

(Note: this problem uses the tables created by the **set-up-ex-tbls.sql** script that we used in the Week 4 Lab, also posted on the public course web site in the "References" section. If you did not run this script in your Oracle account during that lab, you will want to do so before finishing this problem.)

Consider the **Additional Notes - Starting set of select style standards** section on p. 1 of this homework handout, -- it includes our beginning set of **course style standards** for SQL `select` statements.

The following SQL `select` statements are syntactically correct, but are **NOT** meeting these course style standards:

```
select * from empl where salary between 1500 and 2500;
select job_title,
salary from
empl where job_title = 'Clerk' and salary > 1200;
select empl_last_name, job_title, salary from empl,
dept where empl.dept_num = dept.dept_num and dept_loc in ('Dallas',
'Chicago');
```

Use `nano` (or `vi` or `emacs`) to create a file named `325hw3-4.sql`:

```
nano 325hw3-4.sql
```

While within `nano` (or whatever), type in the following within one or more SQL **comments**:

- your name
- `CS 325 - Homework 3 - Problem 4`
- the date this file was last modified

Then:

- use `spool` to start writing the results for this script's actions into a file `325hw3-4-out.txt`

- include a `spool off` command, at the BOTTOM/END of this file.

And now, between these `spool` commands, **RETYPE** the "ugly" `select` statements above so that they **meet** the course style standards for SQL `select` statements.

The results of these `select` statements will not change, but you should find your retyped `select` statements much easier to read.

Submit your files `325hw3-4.sql` and `325hw3-4-out.txt`.

Problem 5

YOU **ARE** USING ORACLE and SQL FOR THIS PROBLEM.

(Note: this problem uses the tables that you created in Homework 2, the tables `Client`, `Video`, and `Rental`.)

Use `nano` (or `vi` or `emacs`) to create a file named `325hw3-5.sql`:

`nano 325hw3-5.sql`

While within `nano` (or whatever), type in the following within one or more SQL **comments**:

- your name
- `CS 325 – Homework 3 – Problem 5`
- the date this file was last modified

Then:

- use `spool` to start writing the results for this script's actions into a file `325hw3-5-out.txt`

- include a `spool off` command, at the BOTTOM/END of this file. Type your answers to the parts below BEFORE this spool off command!

NOTE!!! READ THIS!!!

Now, within your file `325hw3-5.sql`, add in statements for the following, **PRECEDING** EACH with a SQL*Plus `prompt` command noting what problem part it is for.

5 part a

Write a SQL statement to perform a **"pure" relational projection** of the client first names from the `client` table.

5 part b

Write a SQL statement to perform a **"pure" relational projection** of the video rental price and video format columns only (and in that order) from the `video` table.

5 part c

Write a SQL statement to perform a **relational selection** of rows of the `video` table for videos with a length of less than 90 minutes.

5 part d

Write a SQL statement to perform a **relational selection** of rows of the `rental` table for rentals involving the video with ID of `'11111H'`.

5 part e

(OK, maybe just this ONCE...) Write a SQL statement to perform a **Cartesian product** of the `client` and `rental` tables.

5 part f

Write a SQL statement to perform an **equi-join** of the `video` and `rental` tables.

5 part g

Write a SQL statement to perform an **equi-join** of the `client` and `rental` tables.

Submit your files `325hw3-5.sql` and `325hw3-5-out.txt`.