# CS374 -- ASSIGNMENT #1

## NAME: Aaron Borjas DUE: 17 Sep Received: .

**GRADE:**

|  |  |  |
| --- | --- | --- |
| **CATEGORY** | **POINTS** |  |
| EX01\_01 |  | 10 |
| EX01\_02 |  | 10 |
| EX01\_03 |  | 22 |
| EX01\_04 |  | 15 |
| EX01\_05 |  | 10 |
| EX01\_06 |  | 28 |
| EX01\_07 |  | 15 |
|  |  |  |
| **TOTAL** |  | 100 |

## READING: Elmasri Chapter 1 and Chapter 2

## EXERCISES:

**EX01\_01 –** Exercise 1.8, page 28. You should be able to come up with at least 5 queries and 4 update operations. Make sure each table participates in at least one query and at least one update operation.

Queries:

1. Get a list of all the classes taught by Professor Anderson.
2. Check which classes Brown had an A in.
3. Look at the students in the Intro to Computer Science course in fall of 2007 and see if any student with a CS major failed.
4. Check how many credit hours each student has and see if they are under 12 credits or over 16 credits.
5. List the classes taunt in spring by Professor Knuth.

Update operations:

1. Add a new course, Assembly, to the course table with the Course\_number CS2780, Credit\_hours of 3, and a department of CS.
2. Add a new student with the last name Doe, Student\_number 23, and major in MATH
3. Give Doe, student\_number 23, a grade of B Intro to Computer Science section 92 and add this to grade\_report.
4. Make a prerequisite of CS2780 (assembly) CS1310 (into to computer science)

**EX01\_02 –** Exercise 1.10, page 28.

The relationships between the data in Figure 1.2 are as follows:

The students created in the Student table are referenced in the grade report, so that grades are assigned

to them for specific sections of a course.

Courses created in the course table have a name, course number, credit hours, and department. The

course number is important because it is how the course is identified in outside tables. The department of

a course is also important because students in the CS major should know what classes they need to take.

The section table indicates the section of a course, when it occurs, and who teaches it. The section

identifier is important to remember because it is included in the grade report table for student grades.

The grade report table helps compile some information from previous tables, namely the student table

and each student’s ID number, as well as the section table and the section identifier number mentioned

previously. It also adds a grade so the student can keep track of their grades.

Finally, the prerequisite table checks which classes must be taken before a given course. This table

mainly references the course number identifier found in the course table.

**EX01\_03 –** Exercise 2.14, page 56.

If I were designing a Web-based system to make airline reservations and sell tickets, I would choose the

three-tier DBMS architecture for web applications. The three-tier architecture is the best option because it provides an interface for the clients to peruse but doesn’t give them free access to the database (and other people’s flight/payment information). The two-tier architecture does not make sense for this type of system since it is not Web-based and requires extra software for the client to access the database. The basic client/server architecture is not optimal for the system we are creating because you have to define servers for specific tasks, such as managing files, web interface, and emails. Centralized DMBS architecture is definitely not a great choice, as user application programs, user interfaces, and DMBS functionality is confined to a specific number of computers, which can be overloaded by too much data, too many users, and other errors. Because of these reasons, the three-tier system is the best choice for maintaining an airline reservation and ticket selling system.

**EX01\_04 –** Design a relation that stores information about restaurants. Give the schema for that relation, including attribute names, the domain for each attribute, and key(s). What is the *arity* of the relation?

**Restaurant**

|  |  |  |  |
| --- | --- | --- | --- |
| Restaurant\_name | Restaurant\_rank (local) | Wait\_time | Daily\_income |

Restaurant\_name is a string of up to 50 characters

Restaurant\_rank is a local (city based) number, from 0-999.

Wait\_time is in minutes (once order is placed to when food is received)

Daily\_income is USD.

The arity of the relation is 4, since there are four attributes of the Restaurant table.

**EX01\_05 –** Give an instance for your restaurant table. The cardinality of your table should be greater than seven.

|  |  |  |  |
| --- | --- | --- | --- |
| Restaurant\_name | Restaurant\_rank | Wait\_time | Daily\_income |
| McDonald’s | 005 | 3 minutes | $1001 |
| Domino’s | 003 | 7 minutes | $1200 |
| Panda Express | 008 | 10 minutes | $800 |
| Mod Pizza | 007 | 8 minutes | $870 |
| Wendy’s | 001 | 5 minutes | $1500 |
| Starbucks | 002 | 2 minutes | $1400 |
| Dutch Bros | 004 | 3 minutes | $1100 |
| Taco Bell | 006 | 4 minutes | $950 |

**EX01\_06 –** Suppose you are designing a database for a restaurant supply company. You have a table for the different restaurants you supply ingredients to. Give schemas for two separate tables that a restaurant supply company might also need in their business.

**Orders**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Restaurant\_name | Order\_number | Order\_item | Order\_size | Order\_cost | Ship\_date |

Restaurant\_name is a string of up to 50 characters

Order\_number is a 5 digit number.

Order\_item is a string of 30 characters.

Order size is in pounds (lbs)

Order cost is in USD

Ship date is in American dating system, for example September 15, 2020 is 9/15/2020

**Supplies**

|  |  |  |  |
| --- | --- | --- | --- |
| Item\_name | Item\_stock | Outgoing | Incoming |

Item\_name is a string of 30 characters.

Item\_stock is in pounds (lbs)

Outgoing is in pounds (lbs) (pounds being send out)

Incoming is in pounds (lbs) (pounds being received)

**EX01\_07 –** Using your DBMS of choice (MySQL, SQL Server, Access, …), create your tables using the system’s user interface.