## **Team Name:** Team Project 9

## **Team Members and Contributions**

* Aimee Valladares - Primarily created the Data Dictionary in Excel and subsequently updated it. Primarily wrote and updated the 3-End User sample queries. Compiled the primary data sets from labor statistics that would later be inserted into our database. Helped create the rest of the insert statements in SQL. Wrote the high level description of what the data information means as well as the purpose of what our database is for. Created the Idea Expo poster and PowerPoint presentation. Helped present our database during the Idea Expo.
* Alec Lewis - Helped create the Data Dictionary and the ER Diagram as well as establish the business rules for reference. Helped research for the data source for our database. Helped resolve issues with our SQL code when running the queries. Helped present our database during the Idea Expo.
* Martin Butcher - Created the database, the tables, and some of the insert statements in SQL as well as helped resolve issues for our SQL code. Is responsible for the screen snapshots of our database running and test queries. Documented known issues and problems with our database. Turned in our material for the Idea Expo. Helped present our database during the Idea Expo.
* Zack Retamozo - Helped create the Data Dictionary and 3-End User sample queries. Created the first draft of the ER Diagram in the Dia Diagram.

## **Project Description:**

The purpose of our database is to help educate students on possible career options they can pursue when taking into account certain information. Our database will be able to determine what career options are available to a student when taking into account their major. This will allow the student to and figure out what is the best career option with the information provided by our database. This database consists of 6 tables: Concentration, Location, Major, Profession, Student, and Wage. The database will take into account information corresponding to the student from the Concentration, Major, and Student tables while the database will contain information pertinent for job analysis in the Location, Profession, and Wage tables.

In our database, each of our tables contains different information. The concentration table contains three attributes: the concentration ID number, the name of the concentration, and the corresponding major ID number. The primary key of this table is the concentration ID number. This table has one foreign key, the major ID number which is from the Major table. This table consists of 35 rows of information.

The major table contains three attributes: the major ID number, the name of the major, and whether or not the major has ABET accreditation (Y/N). The primary key for this table is the major ID number. This table does not have a foreign key. This table consists of 9 rows of information.

The student table contains five attributes: the student’s ID number, the student’s first name, the student’s last name, their corresponding major ID number, and their corresponding concentration ID number. The primary key for this table is the student’s ID number. This table has two foreign keys: the major ID number from the Major table and the concentration ID number from the Concentration table. The number of rows of information in this table because its source of information is from participation.

The profession table contains six attributes: the profession ID number, the name of the profession, what the typical entry-level education is, whether or not previous work experience is needed (Y/N), the forecast of the change in the number of people employed (aka the job outlook), and the corresponding major ID number. The primary key for this table is the profession’s ID number. This table has one foreign key the major ID number from the Major table. This table consists of 96 rows of information.

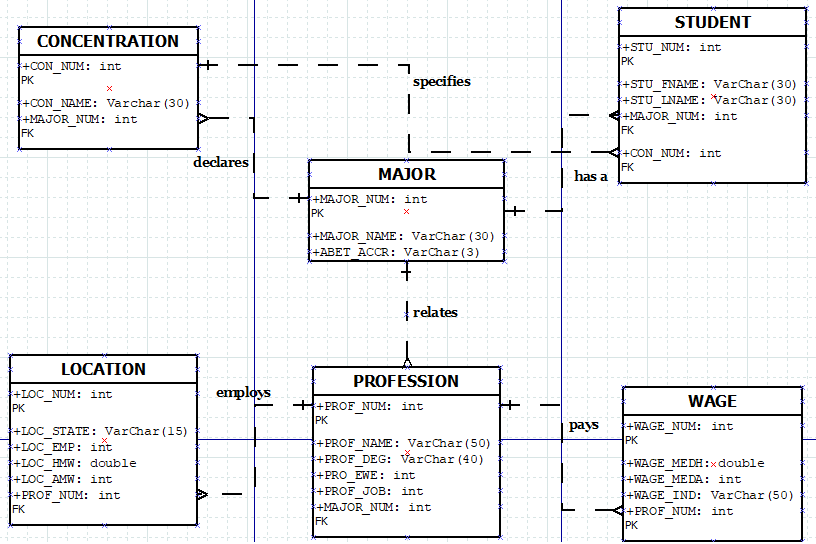
The location table contains six attributes: the location’s ID number, which state in the United States has the highest possibility of employment, the corresponding employment level, the hourly mean wage, and the annual mean wage, and the profession’s ID number. The primary key for this table is the location’s ID number. This table has one foreign key the profession ID number from the Profession table. This table consists of 96 rows of information.

Finally, the wage table contains five attributes: the wage ID number, the hourly median wage, the annual median wage, the industry with the highest levels of employment, and the profession ID number. The primary key for this table is the wage ID number. This table has one foreign key the profession ID number from the Profession table. This table consists of 96 rows of information.

## **Data Dictionary:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table Name: | Attribute Name: | Contents: | Type: | Format: | Range: | Required: | PK or FK | FK Referenced Table | Number of Rows |
| Major | MAJOR\_NUM | Major ID | Integer | 999 | 999 - 100 | Yes | PK |  | 9 |
|  | MAJOR\_NAME | Name | VarChar(30) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | ABET\_ACCR | ABET Accreditation | VarChar(3) | XXXXXXXXXX | XXX | Yes |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| Profession | PROF\_NUM | Profession ID | Integer | 999999999 | 999999999 - 100000000 | Yes | PK |  | 96 |
|  | PROF\_NAME | Profession Name | VarChar(50) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | PROF\_DEG | Typical entry-level education | VarChar(40) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | PROF\_EWE | Is Previous Work Experience Needed? | VarChar(4) | XXX | Xxx | No |  |  |  |
|  | PROF\_JOB | Job Outlook | Integer | 99 | 1 - 99 | Yes |  |  |  |
|  | MAJOR\_NUM | Major ID | Integer | 999999999 | 999999999 - 100000000 | Yes | FK | MAJOR |  |
|  |  |  |  |  |  |  |  |  |  |
| Location | LOC\_NUM | Location ID | Integer | 999999999 | 999999999 - 100000000 | Yes | PK |  | 96 |
|  | LOC\_STATE | Location in the US (What state?) | VarChar(15) | XXXXXXXXXX | Xxxxxxxxxxxxxxx | Yes |  |  |  |
|  | LOC\_EMP | Employment Level | Integer | 999999999 | 999999999 - 100000000 | Yes |  |  |  |
|  | LOC\_HMW | Hourly Mean Wage | Double | 99.99 | 00.00 - 99.99 | No |  |  |  |
|  | LOC\_AMW | Annual Mean Wage | Integer | 999999999 | 999999999 - 100000000 | No |  |  |  |
|  | PROF\_NUM | Profession ID | Integer | 999999999 | 999999999 - 100000000 | Yes | FK | PROFESSION |  |
|  |  |  |  |  |  |  |  |  |  |
| Concentration | CON\_NUM | Concentration ID | Integer | 999999999 | 999999999 - 100000000 | Yes | PK |  | 35 |
|  | CON\_NAME | Concentration Name | VarChar(30) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | MAJOR\_NUM | Major ID | Integer | 999999999 | 999999999 - 100000000 | Yes | FK | MAJOR |  |
|  |  |  |  |  |  |  |  |  |  |
| Student | STU\_NUM | Student ID | Integer | 999999999 | 999999999 - 100000000 | Yes | PK |  | 10 |
|  | STU\_FNAME | Student's First Name | VarChar(30) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | STU\_LNAME | Student's Last Name | VarChar(30) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | MAJOR\_NUM | Major ID | Integer | 999999999 | 999999999 - 100000000 | Yes | FK | MAJOR |  |
|  | CON\_NUM | Concentration ID | Integer | 999999999 | 999999999 - 100000000 | Yes | FK | CONCENTRATION |  |
|  |  |  |  |  |  |  |  |  |  |
| Wage | WAGE\_NUM | Wage ID | Integer | 999999999 | 999999999 - 100000000 | Yes | PK |  | 96 |
|  | WAGE\_MEDH | Hourly Median Wage (50th percentile) | Double | 99.99 | 00.00 - 99.99 | No |  |  |  |
|  | WAGE\_MEDA | Annual Median Wage (50th percentile) | Integer | 999999999 | 999999999 - 100000000 | No |  |  |  |
|  | WAGE\_IND | Top Industry | VarChar(50) | XXXXXXXXXX | Xxxxxxxxxxxxxxxxxxxxx | Yes |  |  |  |
|  | PROF\_NUM | Profession ID | Integer | 999999999 | 999999999 - 100000000 | Yes | FK | PROFESSION |  |

## **ER Diagram:**



## **Source Code Files**

-- Create CONCENTRATION table

create table CONCENTRATION(

CON\_NUM int not null,

CON\_NAME var char(30),

MAJOR\_NUM int,

primary key (CON\_NUM),

foreign key (MAJOR\_NUM) references MAJOR(MAJOR\_NUM)

);

-- Create LOCATION table

create table LOCATION(

LOC\_NUM bigint(20) not null,

LOC\_STATE varchar(25),

LOC\_EMP int,

LOC\_HMW double,

LOC\_AMW int,

PROF\_NUM int,

primary key (LOC\_NUM),

foreign key (PROF\_NUM) references PROFESSION(PROF\_NUM)

);

-- Create MAJOR table

create table MAJOR(

MAJOR\_NUM int not null,

MAJOR\_NAME varchar(30) not null,

ABET\_ACCR varchar(3),

primary key (MAJOR\_NUM)

);

-- Create PROFESSION table

create table PROFESSION(

PROF\_NUM int not null,

PROF\_NAME varchar(50) not null,

PROF\_DEG varchar(40) not null,

PROF\_EWE varchar(30) not null,

PROF\_JOB int not null,

MAJOR\_NUM int,

primary key (PROF\_NUM, MAJOR\_NUM),

foreign key (MAJOR\_NUM) references MAJOR(MAJOR\_NUM)

);

-- Create STUDENT table

create table STUDENT(

STU\_NUM int not null,

STU\_FNAME varchar(30) not null,

STU\_LNAME varchar(30) not null,

MAJOR\_NUM int,

CON\_NUM int,

primary key (STU\_NUM),

foreign key (MAJOR\_NUM) references MAJOR(MAJOR\_NUM),

foreign key (CON\_NUM) references CONCENTRATION(CON\_NUM)

);

-- Create WAGE table

create table WAGE(

WAGE\_NUM int not null,

WAGE\_MEDH double,

WAGE\_MEDA int,

WAGE\_IND varchar(50),

PROF\_NUM int,

primary key (WAGE\_NUM),

foreign key (PROF\_NUM) references PROFESSION(PROF\_NUM)

);

-- Insert Statements

-- CONCENTRATION insert

insert into CONCENTRATION

(CON\_NUM, CON\_NAME, MAJOR\_NUM)

values

(‘2001’, ‘Advanced Topics’, ‘001’);

-- LOCATION insert

insert into LOCATION

(LOC\_NUM, LOC\_STATE, LOC\_EMP, LOC\_HMW, LOC\_AMW, PROF\_NUM)

values

(‘3001’, ‘California’, ‘15930’, ‘25.72’, ‘52560’, ‘1001’);

-- MAJOR insert

insert into MAJOR

(MAJOR\_NUM, MAJOR\_NAME, ABET\_ACCR)

values

(‘001’, ‘Computer Science’, ‘Yes’);

-- PROFESSION insert

insert into PROFESSION

(PROF\_NUM, PROF\_NAME, PROF\_DEG, PROF\_EWE, PROF\_JOB, MAJOR\_NUM)

values

(‘1001’, ‘Audio and Video Equipment Technician’, “Bachelor’s Degree”, ‘None’, ‘13’, ‘001’);

-- STUDENT insert

insert into STUDENT

(STU\_NUM, STU\_FNAME, STU\_LNAME, MAJOR\_NUM, CON\_NUM)

values

(‘7023’, ‘Aimee’, ‘Valladares’, ‘001’, ‘2006’);

-- WAGE insert

insert into WAGE

(WAGE\_NUM, WAGE\_MEDH, WAGE\_MEDA, WAGE\_IND, PROF\_NUM)

values

(‘4001’, ‘21.04’, ‘43770’. ‘Finance and Insurance’, ‘1001’);

## **Instructions for running and using your database:**

In order to access our database you must first open the Linux terminal named PuTTY. After selecting PuTTY enter “ember.hpc.lab” (the phrase without the quotation marks) into the Host Name to open it. You will then have to log in with your username and then enter your password for ember.hpc.lab. Then you will enter “mysql -u [username] -p -h ember-db” replacing [username] with your username into the command line. You will then receive an Enter password: prompt, you will then type your password. An indicator of whether or not you entered the correct password is the MariaDB [(none)]> prompt appearing. To display all the databases you have access to enter “show databases”.

To access the database, type “avalladares7023\_PRJ”. Now you should have access to our career database. To view our tables, you can type the command “show tables”. To view the table structure for each of our tables, type “describe [Table Name];” To view each of the data sets for each of the tables, type “select \* from [Table Name]”. To run our sample end-user queries type in the SQL code from the Sample Queries section of this report.

## **Sample Queries:**

The first end-user query will allow the user to view the top 5 professions with the highest job outlook a student can possibly pursue when taking into account their major.

select pro.PROF\_NAME "Profession Name", pro.PROF\_DEG "Typical Entry-Level Education", pro.PROF\_EWE "Previous Work Experience Needed", pro.PROF\_JOB "Job Outlook", maj.MAJOR\_NAME "Major Name"

from PROFESSION pro, MAJOR maj

where pro.MAJOR\_NUM = maj.MAJOR\_NUM and maj.MAJOR\_NUM = 001

order by pro.PROF\_JOB desc

limit 5;

The second end-user query will allow the user to view professions with the highest level of employment (in the United States) and the corresponding wages (based on state data) when given a student’s major.

select loc.LOC\_STATE "State", loc.LOC\_EMP "Employment Level", loc.LOC\_HMW "Hourly Mean Wage", loc.LOC\_AMW "Annual Mean Wage", pro.PROF\_NAME "Profession Name", maj.MAJOR\_NAME "Major Name"

from LOCATION loc, PROFESSION pro, MAJOR maj

where loc.PROF\_NUM = pro.PROF\_NUM and pro.MAJOR\_NUM = maj.MAJOR\_NUM and maj.MAJOR\_NUM = 001

order by loc.LOC\_EMP desc;

The third end-user query allows the user to view the top 5 professions with the highest paying wages and biggest industry in regard to employment (based on national data) when taking into account the student’s major.

select maj.MAJOR\_NAME "Major Name", pro.PROF\_NAME "Profession Name", w.WAGE\_MEDH "Hourly Median Wage", w.WAGE\_MEDA "Annual Median Wage", w.WAGE\_IND "Top Industry"

from WAGE w, PROFESSION pro, MAJOR maj

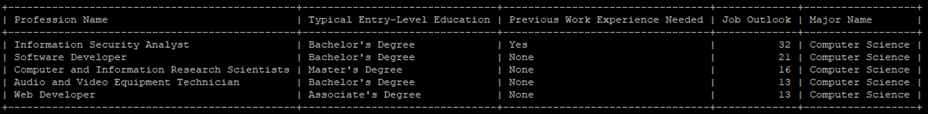
where pro.PROF\_NUM = w.PROF\_NUM and pro.MAJOR\_NUM = maj.MAJOR\_NUM and maj.MAJOR\_NUM = 001

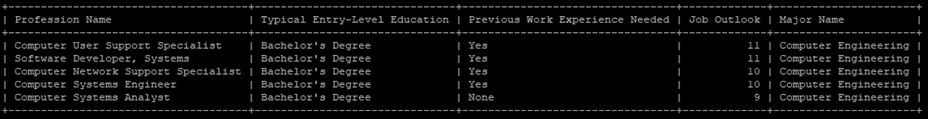
order by w.WAGE\_MEDA desc

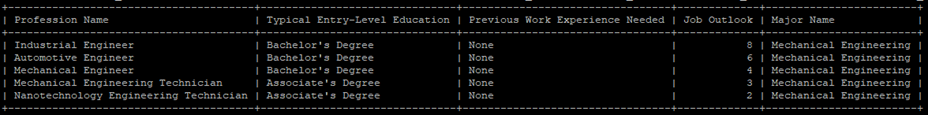
limit 5;

## **Screen Snapshots:**

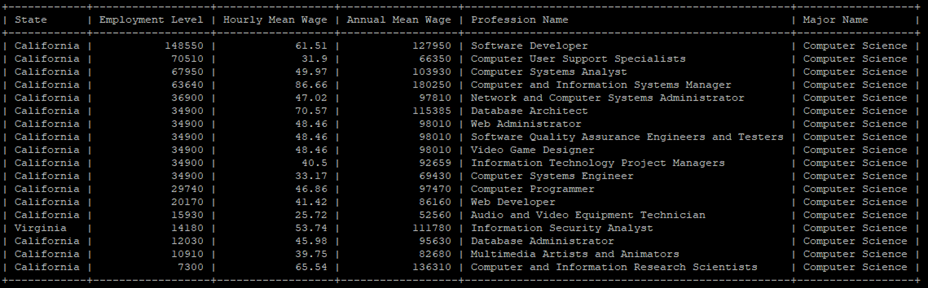
The following screenshots are of the first sample end-query running:



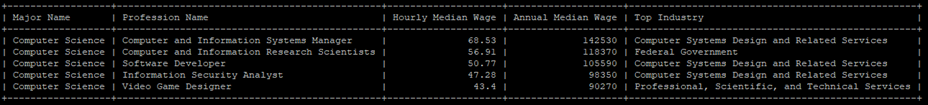


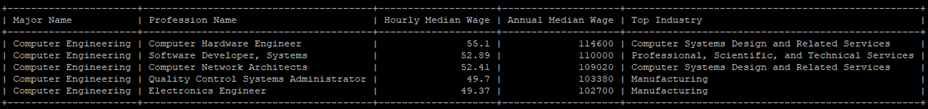


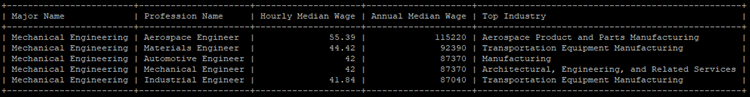
The following screenshots are of the second sample end-query running:



The following screenshots are of the third sample end-query running:







## **Known Issues:**

The database does not take into consideration the foreign job market and is focused entirely on the United States. As there are several leading industries overseas, it would be best to look into them if the database wishes to be further accessible. A possible solution for this issue is to create a new table that takes into account other countries with a notable presence in the job market with the corresponding employment levels, wages, and the type of industry.

Additionally, all majors listed are those offered at Florida Polytechnic University. If a different college wishes to use it, they may have additional majors that are not included in our database. As such, this database would be limited to use by students at this university. One possible solution is to insert other possible majors into the existing major table and making the necessary adjustments with the profession, location, and wage tables. Furthermore, because we are taking into account that other universities are involved in this we would create a University table.

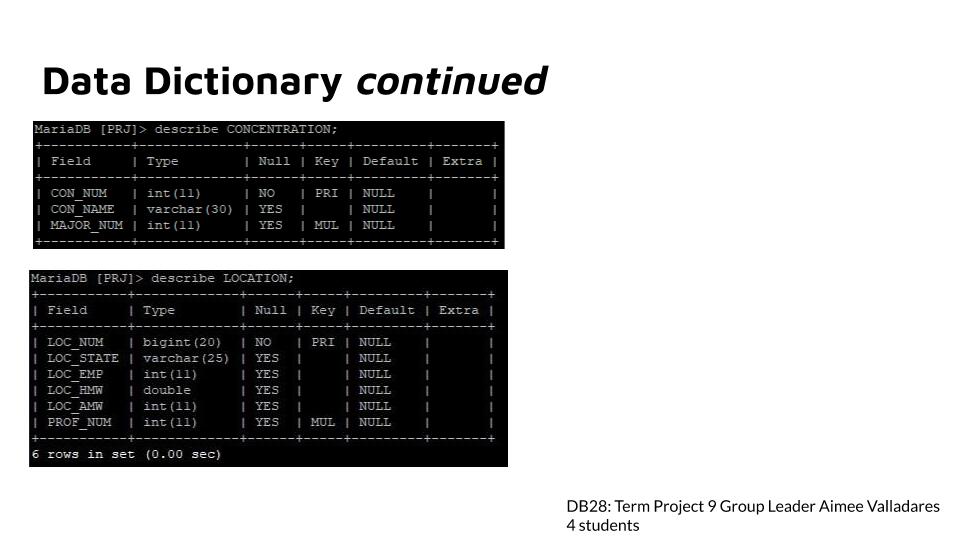
The profession table does not contain every possible job that relates to the majors as there are many more that could not be entered due to time constraint. A possible solution for this issue is to conduct further research on information of unaccounted professions.

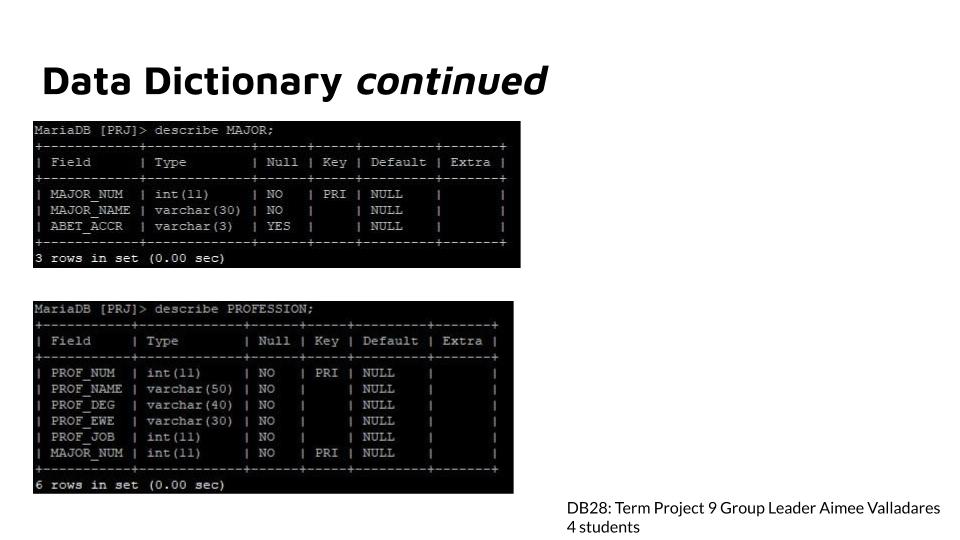
Finally, this database uses data that may very well change within a short period of time. In order to stay relevant, the database will have to be updated regularly. This issue can be solved by performing periodic updates or creating a program that automatically updates the database.

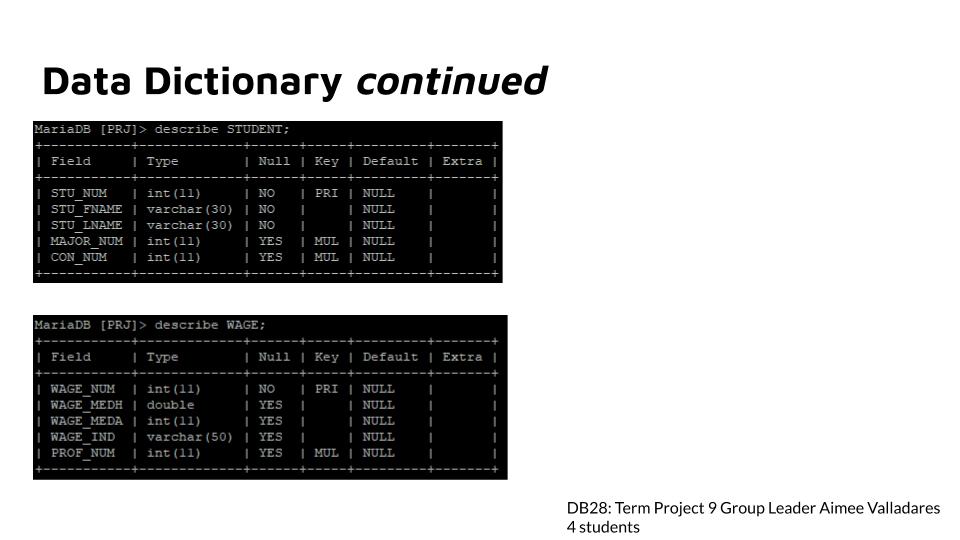
## PowerPoint Presentation:

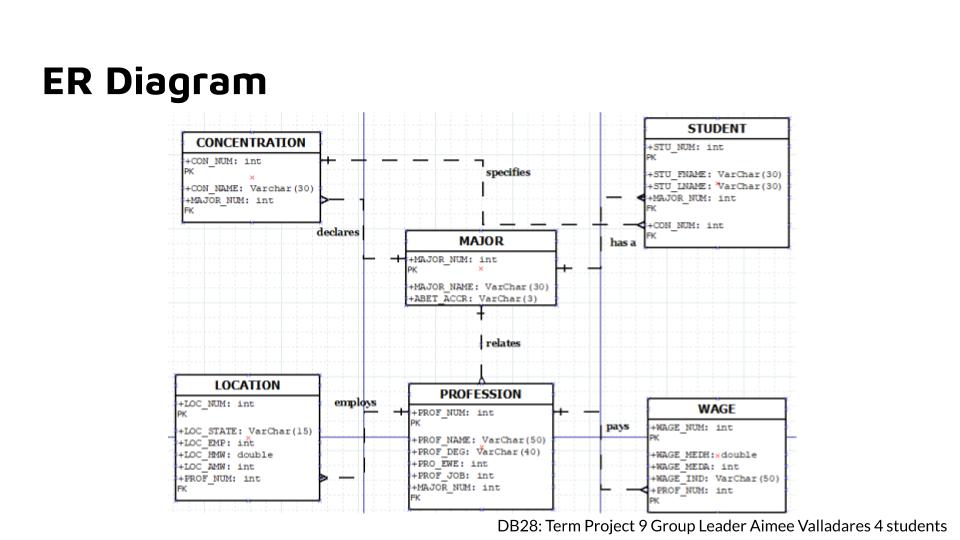




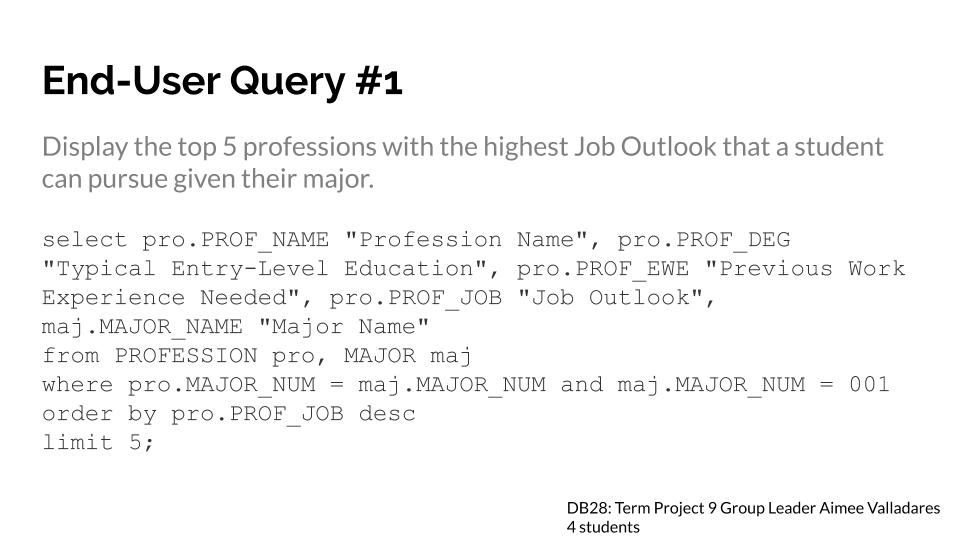


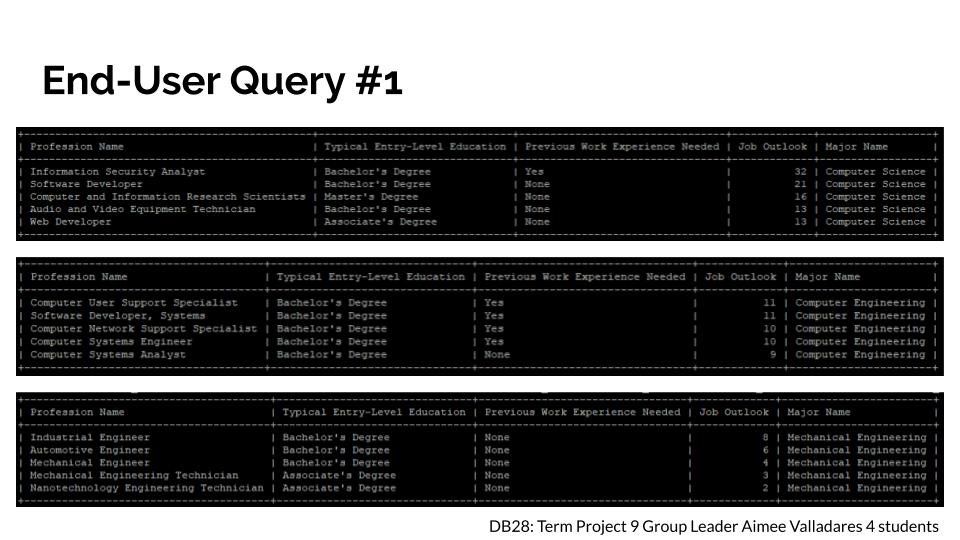


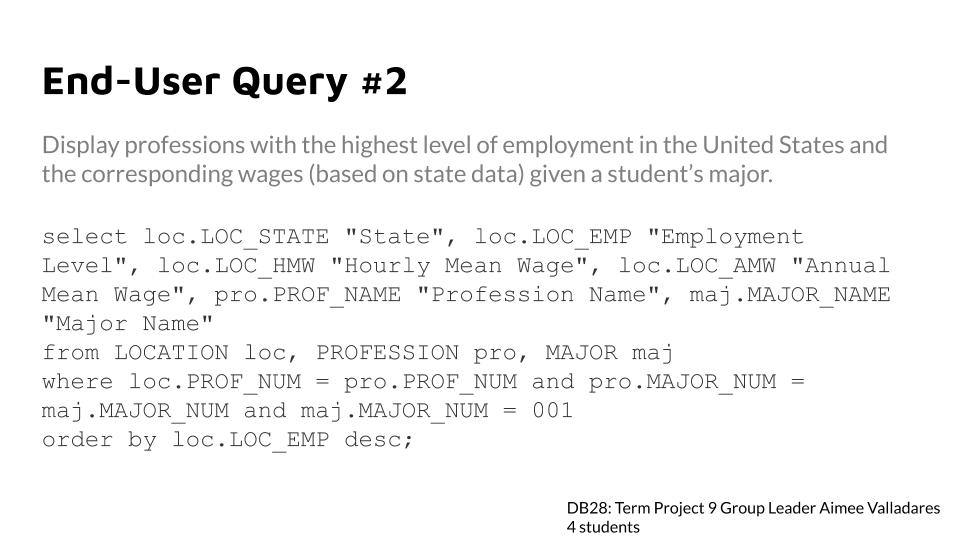


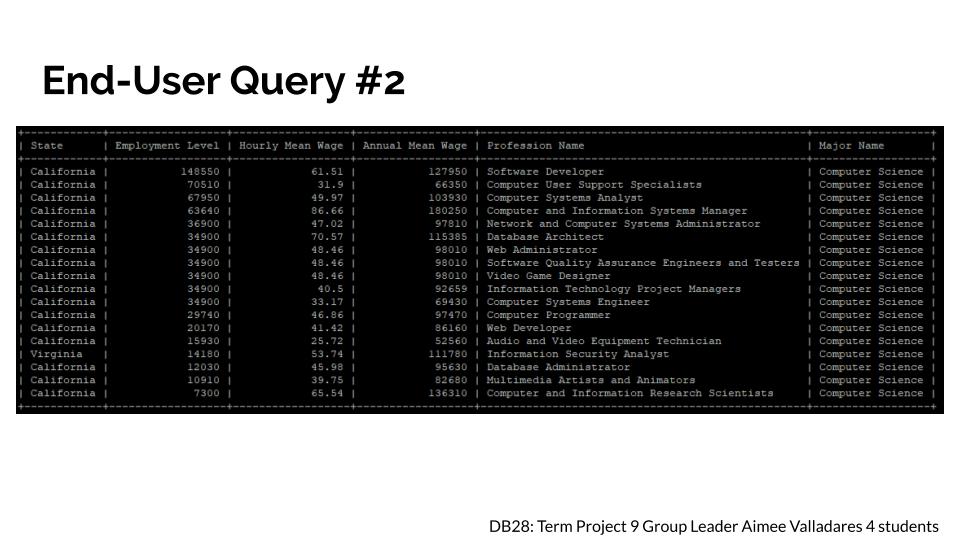


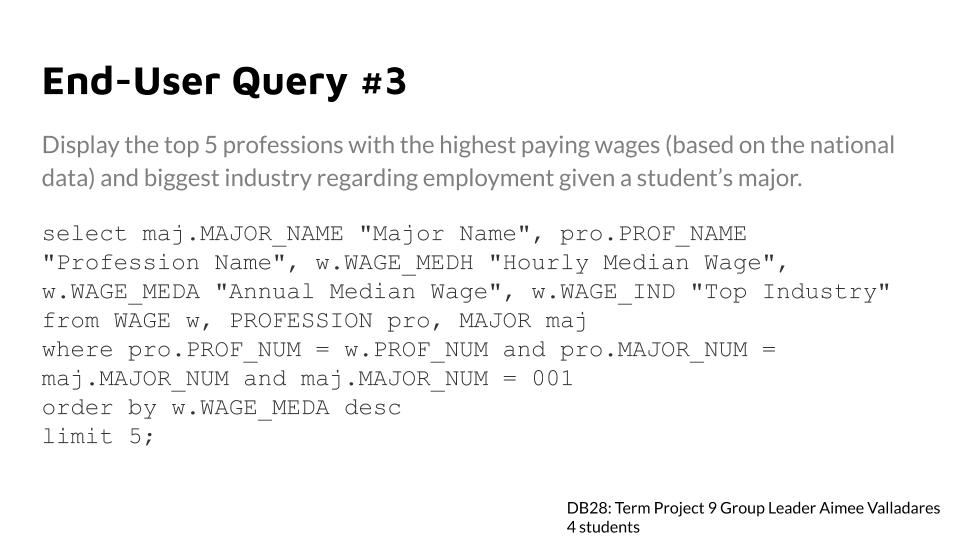


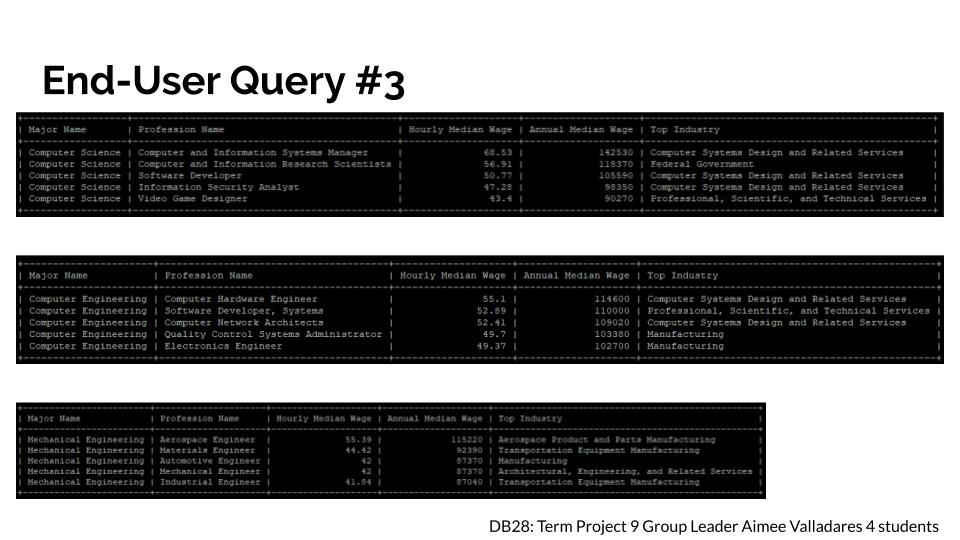


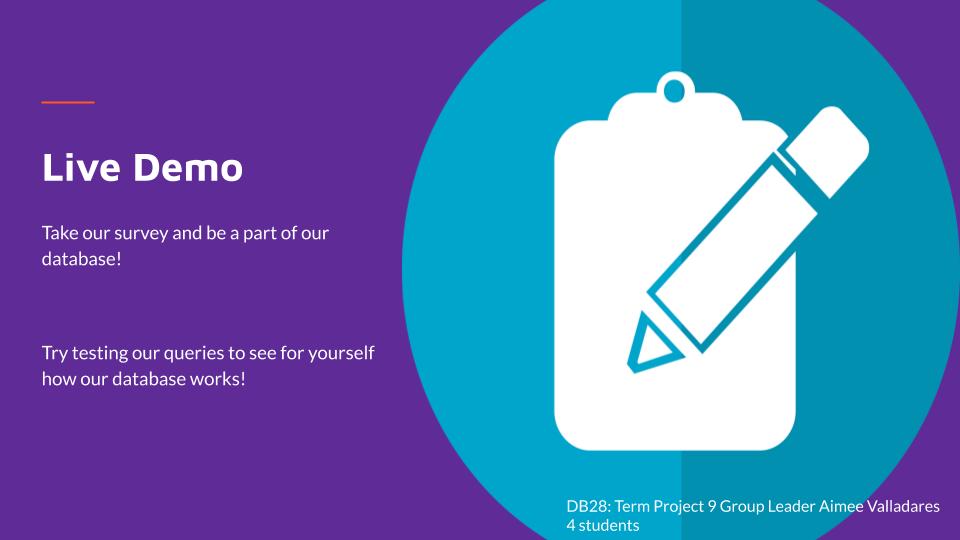


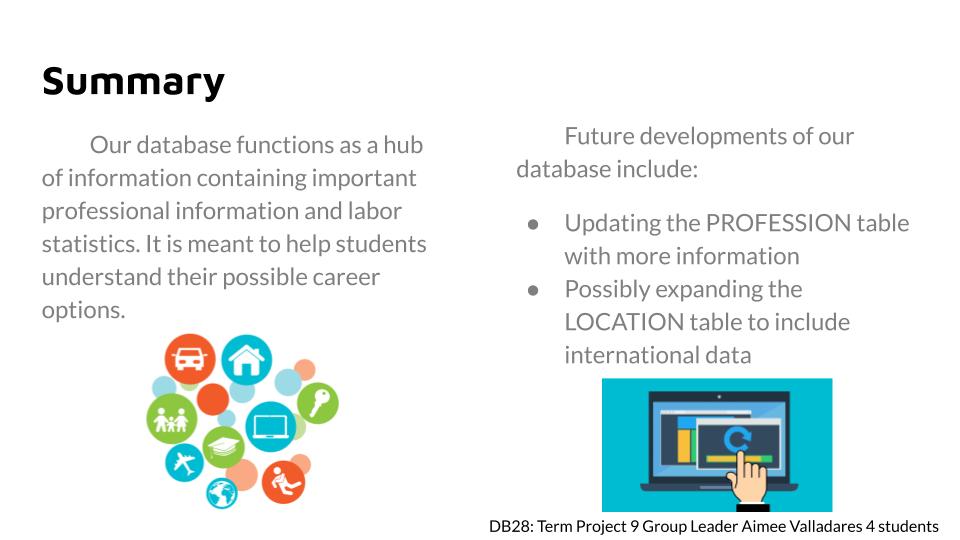














## **Idea Expo Poster:**

****