**CHAPTER 2 PROBLEM SET**

Take your Semester Project ERD and **implement the tables of your database in Oracle.** To begin, create a script in SQL Live, similar to the one used to make the College or Animal Schemas in class.

The first set of commands consist of DROP statements to drop previous versions of your tables. Will allow you to run your script more than once should you have some errors. You will DROP the tables in the ***reverse order*** from which you create them. (and the first time you run the DROP commands, you will get errors because the tables don’t exist yet. This is OK, and these errors will disappear on subsequent runs of the script.)

Next, list the CREATE statements for your tables. Primary key, foreign key, and check constraints should be **named out-of-line (table-level) constraints**. (Note: NOT NULL is the only type of constraint that must be a column-level constraint.)

Recall that before you can create a foreign key reference, the table to which it refers must already exist, and the field must be defined as a primary key. This means that you must create the ***parent table*** (the one with no foreign keys) first. Then you can create the child table, with foreign keys that references that parent table. Make sure you have the order of your CREATE statements correct or one or more tables will not be created. If your child table has more than one foreign key, then you will have to create all parent tables (not just one.)

Create a **sequence to use as the primary key** **for one of your tables**. Only create a sequence for one table, because if you use sequences for all primary keys, it will be very confusing.

Use comments to indicate different tasks, such as

**-- Dropping tables for multiple runs**

**-- Creating the SHIPS table**

At the end of the script, place one COMMIT command, as shown:

**COMMIT;**

**TURN IN A HARD COPY PRINTOUT of:**

1. the ***SQL script*** that creates all of the tables from your semester project ERD, and
2. a ***RUN of the script***, showing where all SQL statements run with no errors.
3. Turn in these pages plus your Entity-Relationship Diagram.

**SAMPLE of this homework (must all be printed). This shows you two tables and the sequence, but you must create all of your tables for your ERD and one sequence:**

-- Dropping TABLES and SEQUENCES for multiple runs

DROP TABLE hogwarts\_courses CASCADE CONSTRAINTS;

DROP TABLE hogwarts\_professors CASCADE CONSTRAINTS;

DROP SEQUENCE hogwarts\_professors\_seq;

-- Creating the Hogwarts\_Professors table

CREATE TABLE HOGWARTS\_PROFESSORS

(prof\_id NUMBER(6),

prof\_lastname VARCHAR2(50) NOT NULL,

prof\_firstname VARCHAR2(50),

prof\_email VARCHAR2(60),

CONSTRAINT hogwarts\_professors\_prof\_id\_pk PRIMARY KEY (prof\_id));

-- Creating the Primary Key sequence for Hogwarts\_professors

CREATE SEQUENCE hogwarts\_professors\_seq

START WITH 1

INCREMENT BY 1;

-- Creating the Hogwarts\_Courses table

CREATE TABLE Hogwarts\_courses

(cour\_id NUMBER(6),

cour\_name VARCHAR2(45) NOT NULL,

cour\_symbol VARCHAR2(45),

cour\_numOfStudents NUMBER(6),

prof\_id NUMBER(6),

CONSTRAINT hogwarts\_courses\_cour\_id\_pk PRIMARY KEY(cour\_id),

CONSTRAINT hogwarts\_courses\_prof\_id\_fk FOREIGN KEY (prof\_id)

REFERENCES hogwarts\_professors(prof\_id));

COMMIT;

**RUN OF SCRIPT:**

Table dropped.

Table dropped.

Sequence dropped.

Table created.

Sequence created.

Table created.

Statement processed.