**Final Project**



The above ER diagram was designed based on the following description:

An auction Web site has items for sale that are provided by sellers. Each item is associated with one or more categories, has an opening price, a description, and an ending time. Further, each item may or may not be featured.  Sellers also have the option to promote their items as special sale items.  Each special sale item has a sale starting and ending timestamp as well as a special sale price for the sale period.  Customers submit bids. The highest, earliest bid submitted before the ending time is the winning bid and the item is sold to the bidder. Each seller must pay the auction company 5% of the winning bid. The auction company wants to be able to analyze the sales behavior of its customers and sellers and so must keep track of all bids and sales.

The auction company should also provide online mechanisms for sellers/buyers to enter comments about transactions and buyers/sellers involved in the specific transactions.  The auction company should also provide online mechanisms to keep track of disputes between sellers and buyers as well as bid retractions.  A detailed description should accompany any disputes and similarly a reason must be provided for each bid retraction. The auction company has the right to disqualify persons with a large number of retractions and/or disputes from the auction web site.

This ER diagram was mapped into a set of database tables (with the appropriate normalization/denormalization) as shown below:

**CREATE TABLE users**

**( userID NUMBER**

**CONSTRAINT pk\_users PRIMARY KEY, -- uses user\_id sequence number**

**email VARCHAR2(50),**

**lname VARCHAR2(20),**

**fname VARCHAR2(25),**

**street VARCHAR2(50),**

**city VARCHAR2(25),**

**state VARCHAR2(2),**

**zip VARCHAR2(10),**

**status CHAR(1) -- 'G' or 'B'**

**);**

**CREATE TABLE sellers**

**( userID NUMBER CONSTRAINT fk\_sellers**

**REFERENCES users(userID) ON DELETE CASCADE,**

**creditCardType VARCHAR2(10),**

**creditCardNumber VARCHAR2(16),**

**expiration VARCHAR2(6),**

**bank VARCHAR2(20),**

**accountNo VARCHAR2(25),**

**CONSTRAINT pk\_sellers PRIMARY KEY (userID)**

**);**

**create table buyers**

**( userID NUMBER CONSTRAINT fk\_buyers**

**REFERENCES users(userID) ON DELETE CASCADE,**

**maxBidAmount NUMBER,**

**CONSTRAINT pk\_buyers PRIMARY KEY (userID)**

**);**

**CREATE TABLE items**

**( itemID NUMBER**

**CONSTRAINT pk\_items PRIMARY KEY, -- uses item\_id sequence number**

**name VARCHAR2(50),**

**description VARCHAR2(64),**

**openingPrice NUMBER(9,2),**

**increase NUMBER(9,2),**

**startingTime DATE,**

**endingTime DATE,**

**featured CHAR(1), -- 'Y' or 'N'**

**userID NUMBER CONSTRAINT fk\_items**

**REFERENCES sellers(userID) ON DELETE CASCADE**

**);**

**CREATE TABLE categories**

**( cID NUMBER**

**CONSTRAINT pk\_categories PRIMARY KEY, -- uses category\_id number**

**name VARCHAR2(20),**

**description VARCHAR2(128)**

**);**

**CREATE TABLE itemCategory**

**(**

**cID NUMBER CONSTRAINT fk\_itemcategory\_cID**

**REFERENCES categories(cID) ON DELETE CASCADE,**

**itemID NUMBER CONSTRAINT fk\_itemcategory\_itemID**

**REFERENCES items(itemID) ON DELETE CASCADE,**

**CONSTRAINT pk\_itemcategory PRIMARY KEY (cID, itemID)**

**);**

**CREATE TABLE promotions**

**(**

**itemID NUMBER**

**CONSTRAINT fk\_promotions REFERENCES items(itemID) ON DELETE CASCADE,**

**startingTime DATE,**

**endingTime DATE,**

**salePrice NUMBER(9,2),**

**CONSTRAINT pk\_promotions PRIMARY KEY (itemID,startingTime)**

**);**

**CREATE TABLE bids**

**(**

**userID NUMBER CONSTRAINT fk\_bids\_userID**

**REFERENCES buyers(userID) ON DELETE CASCADE,**

**itemID NUMBER CONSTRAINT fk\_bids\_itemID**

**REFERENCES items(itemID) ON DELETE CASCADE,**

**price NUMBER(9,2),**

**timestamp DATE,**

**CONSTRAINT pk\_bids PRIMARY KEY (userID, itemID, price)**

**);**

**CREATE TABLE retractions**

**(**

**retractionTimestamp DATE,**

**userID NUMBER CONSTRAINT fk\_retractions\_userID**

**REFERENCES users(userID) ON DELETE CASCADE,**

**itemID NUMBER CONSTRAINT fk\_retractions\_itemID**

**REFERENCES items(itemID) ON DELETE CASCADE,**

**reason VARCHAR2(128),**

**CONSTRAINT pk\_retractions PRIMARY KEY (retractionTimestamp, userId, itemID)**

**);**

**CREATE TABLE sales**

**(**

**itemID NUMBER CONSTRAINT fk\_sales\_itemID**

**REFERENCES items(itemID) ON DELETE CASCADE,**

**sellerUserID NUMBER CONSTRAINT fk\_sales\_sellerUserID**

**REFERENCES sellers(userID) ON DELETE CASCADE,**

**buyerUserID NUMBER CONSTRAINT fk\_sales\_buyerUserID**

**REFERENCES buyers(userID) ON DELETE CASCADE,**

**price NUMBER(9,2),**

**settlementdate DATE,**

**CONSTRAINT pk\_sales PRIMARY KEY (itemID)**

**);**

Here is the details of your final project.

Given the nature of the database, as a class, design a data mart and the corresponding star schema. This is group-based. Once the data mart is approved by the instructor, each student will need to deliver the following individually.

Deliverable #1 - Create the data mart by creating and populating the fact and dimensional tables. This is individual based and each student needs to submit the scripts/programs/SQL statements for populating fact and dimension tables. Do not create any indexes at this point. Data (in the comma delimited format) will be provided. Deliverable #1 is due week #2.

Deliverable #2 - After the data mart is designed and approved, the instructor will design a number of queries. Each student will need to write SQL statements to answer the queries. Deliverable #2 is due week #3.

Deliverable #3

* What indexes should you create to improve the performance of the queries you wrote for deliverable #2? Why?
* Can you rewrite the queries you wrote for deliverable #3 to improve their performance? Why or why not?

Deliverable #3 is due week #4.