Assignment-5 ITI-2180 Foundations of Data and Database Management

1. Chapter 6 Hands-on Assignment (problems A – J: see below) – 50 points

Problem A: Create a sequence for populating the Customer# column of the CUSTOMERS table. When setting the start and increment values, keep in mind that data already exists in this table. The options should be set to not cycle the values and not cache any values, and no minimum or maximum values should be declared.

Problem B: Add a new customer row by using the sequence created in Question 1. The only data currently available for the customer is as follows: last name = Shoulders, first name = Frank, and zip = 23567.

Problem C: Create a sequence that generates integers starting with the value 5. Each value should be three less than the previous value generated. The lowest possible value should be 0, and the sequence shouldn’t be allowed to cycle. Name the sequence MY\_FIRST\_SEQ.

Problem D: Issue a SELECT statement that displays NEXTVAL for MY\_FIRST\_SEQ three times. Because the value isn’t being placed in a table, use the DUAL table in the FROM clause of the SELECT statement. What causes the error on the third SELECT?

Problem E: Change the setting of MY\_FIRST\_SEQ so that the minimum value that can be generated is –1000.

Problem F:

JustLee Books wants to enhance their order tracking capabilities by adding functionality that allows every order to have a status. This status is a identified by a single character:

'P' the order is being *p*rocessed (the default value)

'B' the order is *b*ack-ordered

'C' the order has been *c*ancelled

'S' the order has been *s*hipped

Change the structure of the ORDERS table to allow the order status by adding a column called *ostatus*. Provide a default value and a check constraint.

Problem G:

JustLee Books wants to keep a history of every order, from order placement through shipping, including if it is back-ordered or cancelled. Create a history table called OHISTORY that has the following fields (no columns can allow nulls):

*oseq#* - a chronological numerical value that is the primary key

*order#* - the order number with a foreign key to the ORDERS table

*ostatus* – must be one of the ostatus values specified above

*sdate* – the date/time when the status changed

Problem H: Create a sequence called *ohistory\_oseq#\_seq* for the OHISTORY table that will be used for the *oseq#* column when a new history record is inserted. Start this value at 1000.

Problem I: Create an index on the order# column of the OHISTORY table

Problem J: Prepare and execute a data dictionary (metadata) query to show the table name, index name, and column name for all indexes on the OHISTORY table.