INTRO:

1. 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.
   1. CREATE SEQUENCE Customers\_seq
2. 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.
   1. INSERT INTO Customers (customer#, lastname, firstname, zip)
3. 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.
   1. CREATE SEQUENCE MY\_FIRST\_SEQ
4. 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?
   1. SELECT MY\_FIRST\_SEQ.NEXTVAL
5. Change the setting of MY\_FIRST\_SEQ so that the minimum value that can be generated is -1000.
   1. ALTER SEQUENCE MY\_FIRST\_SEQ
6. A new table has been requested to support tracking automated emails sent to customers. Create the table and add data as described below.

• Tablename: email\_log

• Columns: emailid (numeric), emaildate (datetime), customer# (numeric)

• Primary key: emailid column, define as an Identity Column

• Add the following data rows and display resulting rows (if any errors occur, explain

why the error is expected)

1. Emaildate = current date, customer# = 1007

2. Emailid = specify to use the column default value, emaildate = current date,

customer# = 1008

3. Emailid = 25, emaildate = current date, customer# = 1009.

1. CREATE TABLE email\_log
2. INSERT INTO email\_log (emaildate, customer#)
3. Create a private synonym that enables you to reference the MY\_FIRST\_SEQ object as

NUMGEN.

* 1. CREATE SYNONYM NUMGEN

1. Use a SELECT statement to view the CURRVAL of NUMGEN. Delete the NUMGEN synonym and MY\_FIRST\_SEQ.
   1. SELECT NUMGEN.CURRVAL
2. Create a bitmap index on the CUSTOMERS table to speed up queries that search for customers based on their state of residence. Verify that the index exists, and then delete the index.
   1. CREATE BITMAP INDEX CUSTOMER\_STATE\_IDX
3. Create a B-tree index on the customer’s Lastname column. Verify that the index exists by querying the data dictionary. Remove the index from the database.
   1. CREATE INDEX customers\_last\_idx