CST2355 – Database Systems Lab Assignment 9

Student Name: \_\_Yanzhang Wu\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID: \_\_041056465\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

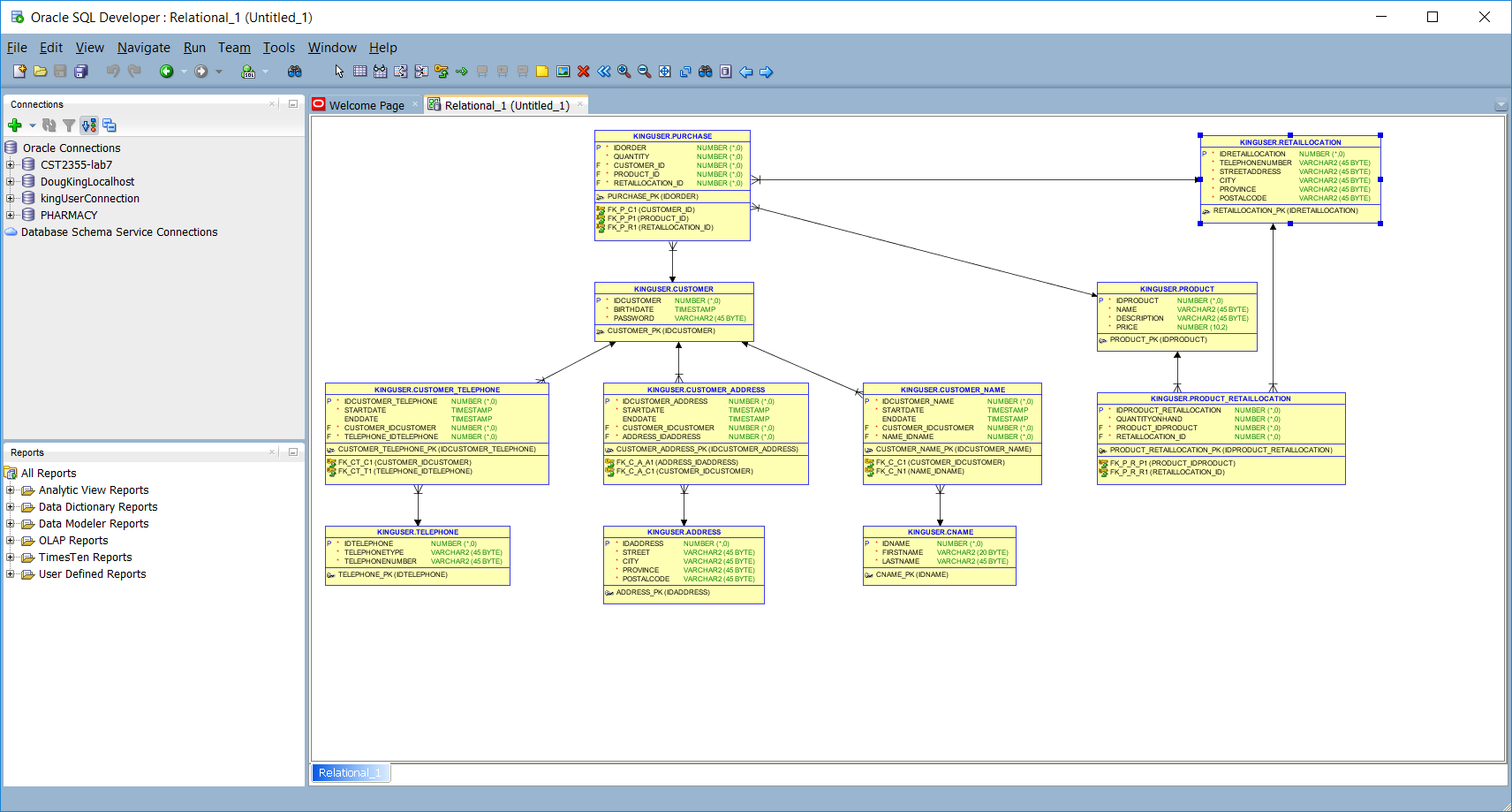
Student email: wu000298@algonquinlive.com\_\_\_\_\_\_

# Hand-in:

1. The lab assignment will be graded out of a maximum 4 points.
2. This template should be used to submit your lab assignment.
3. Make sure you have enough screenshots to completely document that you have completed all the steps.

# Activities (Steps):

1. We are going to create a package that shares items across a set of stored procedures and functions, based on the model that was used in lab 8: (see below)



* 1. First look at the tutorial at: <https://www.tutorialspoint.com/plsql/plsql_packages.htm> It contains an implementation of a package to manage customer records.
  2. Prepare an sql script called “lab9-sequences.sql” that contains the CREATE SEQUENCE statements to create sequences **that will be used in this lab when** **inserting new entries in each of the tables in your schema**. Choose appropriate starting values so that the existing data is not in conflict with the new numbers. (e.g., start them all at 100?) Run the script to create the sequences.
     1. Provide a screenshot showing the contents of your script.

Graphical user interface, application

Description automatically generated Graphical user interface, text, application, email

Description automatically generated Graphical user interface, text, application

Description automatically generated Graphical user interface, text, application

Description automatically generated

* + 1. Provide a screenshot showing the successful running of the sequences script.

Graphical user interface, text, application, email

Description automatically generated Text

Description automatically generated

1. Create a package in the *yourName*User schema called customer\_pkg with the following specification:
   1. The package should contain three stored procedures that each use an IDCUSTOMER IN parameter along with the appropriate IN parameters to update the underlying tables.
      1. change\_telephone()
      2. change\_address()
      3. change\_name()
   2. The package should also contain a function called new\_customer() that returns an INTEGER containing the IDCUSTOMER for a new customer record. The new\_customer() function should have mandatory fields for birthdate and password, and optional fields for each of the data fields in the telephone, address, and name tables.
   3. The package should also contain a function called get\_age() that returns an INTEGER containing the age in years (rounded down to the nearest integer value), for a given IDCUSTOMER.
   4. Provide a screenshot or screenshots showing the package **specification** (just the package specification – not the entire package body) below:
2. Provide the package body for customer\_pkg using the following criteria.
   * 1. Each procedure should insert a new telephone/address/name as appropriate
     2. Each procedure should update the entry in the relationship association table to have the sysdate timestamp as the enddate for the current entry (that is the one with NULL enddate gets updated to have enddate as sysdate). If there is no previous related item (i.e., no current entry with a NULL enddate) then this step should get skipped.
     3. Each procedure should insert a new record in the relationship association table that has the startdate as sysdate and a NULL enddate.
     4. The new\_customer() function should create the customer record along with the telephone, address, and name records as required. If all the non-key fields in a telephone, address or name record would be null, then the associated record should not be created. [**NOTE: The new\_customer function should use the three stored procedures in the package to create the related records.]**
     5. The get\_age() function should returns an INTEGER containing the age in years (rounded down to the nearest integer value), for a given IDCUSTOMER.
     6. Provide screenshots for the three stored procedures below:

Timeline

Description automatically generatedText

Description automatically generatedText, application

Description automatically generated

* + 1. Provide a screenshot showing the new\_customer function below:

Text

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

* + 1. Provide a screenshot showing the get\_age function below:

Graphical user interface, text, application, email

Description automatically generated

1. Once you have embedded all of your screenshots, submit the file in Brightspace and you’re done!