CPSC 304 Project Demo Checklist

Group Name:

Group Members Present:

TA Name: **Date/Time:**

1) (10 points) project and selection
   - user must be able to specify the selection constant (eg. select from Items where quantity >= __) and the attribute to be returned (eg. select ___ from Items)
   - rerun with at least one other constant (eg. change from quantity to price) and attribute

2) (10 points) join
   - join at least two tables and performs a meaningful query (eg. join the Customers and the Transactions table to find the phone numbers of all customers who has purchased a specific item)

3) (10 points) division
   - show the result of a meaningful query (eg. find all the customers who bought all the items)
   - prove that your division results change based on the data either by inserting a new tuple (eg. into the Items table), or deleting an existing tuple (eg. from the Transactions table)

4) (10 points) aggregation
   - show the result of a meaningful query (eg. find the most expensive item)
   - rerun with at least one other aggregation, either on the same query (eg. the least expensive item), or a completely different query (eg. find the first customer)

5) (20 points) nested aggregation with group-by
   - show the result of a meaningful query (eg. the average number of items purchased per customer)
   - show the actual query or view in the code
   - rerun with at least one other nested aggregation

6) (10 points) deletion
   - case 1: deletion causing cascades (students need to explain policy regarding blocking and show code)
   - case 2: deletion without causing cascades

7) (10 points) update
   - update a value that violates some constraint (this is different from the type checking in the UI and must be checked in the database)
   - correct the value and update again

8) (20 points) GUI
   - type checking exists at least for some fields
   - error messages are located at visible and logical places
   - overall look of the UI

9) (30 points) extra features
   - eg. using Bootstrap to create a nice UI could give you 5 points max; an impressive UI with robust type checking and error handling everywhere could give you an additional 5 points max
   - eg. using a simple Trigger could give you 5 points max; using an interesting Trigger that would cause cascading effects could give you an additional 5 points max
   - eg. implementing a login system could give you 5 points max; checking for user privileges using the login ID could give you an additional 5 points max
   - eg. using Privileges on each table in the database could give you 5 points max
   - eg. using any databases in the cloud like Amazon Web Services could give you 10 points max

Total: / 130