NYU – TANDON SCHOOL OF ENGINEERING
CS-GY 6083 - B, SPRING 2020
<u>Principles of Database Systems</u>

Project Part 1

In this first part of the project, you will focus on designing a suitable relational schema that can be used to store the data in the system. In the second part of the project, you will then build a web-accessible frontend that will allow users to use the service via their browsers. Both parts of the project may be done individually or in teams of two students (NO more than two students per group). You will receive an email from the TAs asking you to sign up as a group. Note that the second part of the project builds on top of this first part so you cannot skip this project part 1.

Project Guidelines: In this first part of the project, you will design the relational database (using Oracle Data Modeler) that stores all the data about business case detailed below. You should use your own database system (preferably MySQL) on your laptop or an internet-accessible server. Following is the business case and a list of steps for this part of the project. Note that in this first part, you will only deal with the database side of this project - a suitable web interface will be designed in the second project.

Business Case:

We Do Secure (WDS) is one of the largest life insurance provider company in the United States. Recently WDS has expanded its operation to offer Auto and Home insurance to their customers. After a careful analysis, the business team has decided to build a separate information systems for its Auto and Home insurance services. The project team has articulated a product scope to create a relational database that will incorporate schema design with all related entities and relationships among them to meet WDS's new business services.

The business analyst has provided following details/requirements for the database design.

a) WDS intend to have all customers' details that includes customer's full name, address, gender, marital status, and customer type. The possible data value of gender must be either "M", or "F" representing "Male" or "Female" respectively. A customer may choose not to provide gender data. The possible data value of marital status must be either "M", "S", or "W", representing "Married", "Single", and "Widow/Widower" respectively. The customer can be Automobile Insurance customer or Home Insurance customer or both. The customer type data value "A" represents Automobile Insurance customer and "H" represents Home insurance customer.

b) For Home Insurance customer, WDS intend to store home insurance policy start date and end date, home insurance premium amount, and home policy insurance status. If home insurance policy term is current, status column should have value "C", and if it is expired, it should have value "P". WDS wants to keep all previously expired policy data in the same table.

c) More than one home can be insured under each home insurance policy. For each home insured, WDS intend to store home purchase date, home purchase value, home area in Sq. Ft., Type of home (one of the value as, S,M,C,T representing Single family, Multi Family, Condominium, Town house respectively). In order to decide appropriate home insurance premium, WDS intending to store four parameters namely: Auto Fire notification, Home Security System, Swimming Pool, and Basement. Please note following possible values for each of the parameters.

Parameter	Possible Value	Meaning
Auto Fire Notification	1	There is automatic fire notification to the fire department
Auto Fire Notification	0	There is NO fire notification to the fire department
Home Security System	1	The home security system is installed and monitored
Home Security System	0	The home security system is not installed or not monitored
Swimming Pool	"U"	Underground swimming pool
Swimming Pool	"O"	Overground swimming pool
Swimming Pool	"I"	Indoor swimming pool
Swimming Pool	"M"	Multiple swimming pool
Swimming Pool	null	No swimming pool
Basement	1	There is a basement
Basement	0	There is NO basement

d) Each home insurance policy will generate one or more invoice on different invoice date. Invoice should include payment due date, and invoice amount.

e) WDS accepts home insurance premium in installments and each payment should have stored with payment date and method of payment. WDS accepts payment either as "PayPal", "Credit", "Debit", "Check"

f) For Auto Insurance customer, WDS intend to store auto insurance policy start date and end date, auto insurance premium amount, and auto policy insurance status. If auto insurance policy term is current, status column should have value "C", and if it is expired, it should have value "P". WDS wants to keep all previously expired policy data in the same table.

g) Each Auto insurance policy will generate one or more invoice on different invoice date. Invoice should include payment due date, and invoice amount.

h) WDS accepts Auto insurance premium in installments and each payment should have stored with payment date and method of payment. WDS accepts payment either as "PayPal", "Credit", "Debit", "Check"

i) For each of Auto Insurance policy, one or more vehicles can be insured. For each insured vehicle, WDS intend to store vehicle VIN (vehicle identification number), vehicle make-model-year, and status of the vehicle. The status of the vehicle can be one of "L", "F", or "O" representing "Leased", Financed", "and Owned'.

j) For each vehicle insured, WDS intend to store details of drivers. There may be more than one drivers for each vehicle insured. For each driver, WDS intend to store driver's license number, name, and birthdate.

For this project, identify appropriate entities, identify relationship among entities, attributes of each entity and their datatype- size-constraints, design and implement a centralized relational database system that collects relevant data for WDS.

For any business rule not defined, please state all assumptions you have made (if any) while designing and implementing the database.

Submission:

Submit following deliverables towards project part 1 assignment

a) Create a logical E-R model for database schema with appropriate relationships amongst them. Submit screenshot.
b) Create a relational model, depicting all entities, attributes (name, type, size, and mandatory/optional, primary key), relationships (foreign keys). Submit screenshot.
c) Use relational model to create the schema, and their objects, primary keys, foreign keys, and other constraints. Submit DDL code.
d) Write commands and apply necessary CHECK constraints to apply defined business rules to enforce data consistency. Submit DDL code.
e) Populate meaningful sample data for all entities (10 to 15 records par entity). Submit DML code.
f) List total number of records populated for each entity (just record counts, not full data set). Submit COUNT (*) query and result for each table.
g) Write data dictionary queries that details all tables, columns-dataype-size-mandatory/optional, constraints, and attribute comments of schema objects. Submit data dictionary queries and their corresponding results.
h) Write SQL queries using each of following,
   Q1) Table joins with at least 3 tables in join
   Q2) Multi-row subquery
   Q3) Co-related subquery
   Q4) SET operator query
   Q5) Query with any analytical function or in line view or WITH clause

   For each of above queries use proper column alias, built-in functions, appropriate sorting and submit following three items for each of above queries.
   A1) Select query
   A2) Result of the query
   A3) Statement about what information you intend to achieve through the query

Submit a properly documented description and justification of your entire design (no more than half page), assumptions made, if any along with deliverables as above. This should be a paper in a single PDF document (Times New Roman fonts, 12pts, single spacing) with all screenshots inserted within. On cover page clearly state course, section, submission date, student names and Ids on title page. If you are working in group, name and ID of both students must be listed. There should be table of content page after the cover page.