**Top of Form**

**Database Project**

### ****Divided into 5 parts****

**The objective of this project is to build a database system from an existing application that uses simple tables.**

**The** “Barkley Brothers Real Estate” firm has contracted you and your team to convert the simple table base system described in the document “Barkley\_Brothers\_Real\_Estate\_Proj.docx”

**Database Project Part 1**

You and your team need to analyze the project requirements specified in the document “Barkley\_Brothers\_Real\_Estate\_Proj.docx”.

**Database Project Part 2**

Data Flow Diagram (DFD) – Use the Oracle Data Modeler

Logical data Model (also called Conceptual Design) expressed in an Entity Relationship Diagram (ERD) and identification of all business rules – Use the Oracle Data Modeler

1. A DFD Diagram created using the Oracle Data Modeler.
2. An Entity Relationship Diagram (ERD) using the Oracle Data Modeler
3. Identify all business rules.

**Database Project Part 3**

**Relational Database Design**

Relational Database Design - Convert the ER Disgram to a relational diagram.

The input to Part 3 is the Logical Data Model (ERD) produced in Part 2.

You will proceed to translate the ERD to a Relational Diagram containing tables and referential integrity constraints (Primary Keys, Foreign Keys, Check, and Unique constraints). The ERD will be converted using the following steps:

1. Create table for each entity; include single-valued attributes. Choose key.

2. Create table for each weak entity type; include single-valued attributes.

Include key of owner as a foreign key in the weak entity. Set key as foreign key of owner plus local, partial key.

3. For each 1:1 relationship, add a foreign key to one of the entities involved in the relationship (a foreign key to the other entity in the relationship).\*\*

4. For each 1:N relationship, add a foreign key to the entity on the N-side of the relationship (to reference the entity on the 1-side of the relationship).\*\*

5. For each M:N relationship, create a new table. Include a foreign key for each participant entity, in the relationship. The key for the new table is the set of all such foreign keys.

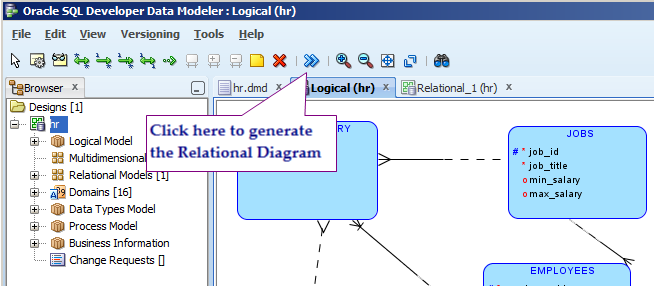
6. For each multi-valued attribute, construct a separate table. Repeat the key for the entity in this new table. It will serve as both the key for this table as well as a foreign key to the original table for the entity.

**NOTE: \*\* The relationship is converted to a relation or table whenever it has attributes that belong to the relationship**.

Once the above process is completed proceed to build the ERD Diagram using the Oracle data Modeler.

Build the Dependency Diagrams.

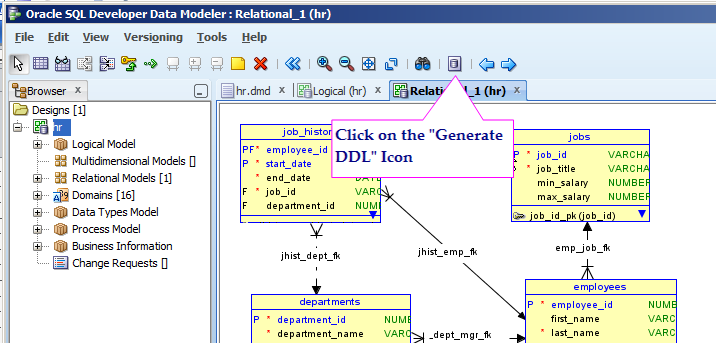
Once you complete the Dependency Diagrams you need to go back to the Oracle Data Modeler and verify your ERD. If you see a need to make an adjustment, this is the time to do it.



Verify that the relational model generated by the Oracle Data Modeler matches the contents of the Dependency diagrams you have built.

Finally you can have the Oracle Data Modeler create your initial SQL DDL script to build your Database Schema. To do this, activate the “Relational Model” pane and you will see an icon in the Icon bar that looks like a data storage symbol with the label “Generate DDL”. Click on it and you will be presented with the DDL File Editor window. Click on “Generate” and follow the menus to generate your DDL script.

Verify that the DDL generated is correct and complete and make the necessary adjustments and add the NON UNIQUE indexes for all the foreign key attributes.



**Deliverables for Database Project Part 3:**

All the deliverables to be merged into a single document named: LASTNAME\_FIRSTNAME\_DB\_Proj\_Part3.docx

1. A Dependency Diagram
2. The Relational Database Diagram.
3. The database script to create your database schema (Generated from the Oracle Data Modeler). The script contain the SQL statements to create the Database schema and is named:

PART3.SQL

1. A Section explaining what you have learned and observed in the process of completing the Part 3 of your Database LAB Project.

**Database Project Part 4:**

**Database Physical Database Design, Optimization and Normalization, and Optimization.**

**Populate the Database tables (SQL INSERT Statements)**

The input to this Part 4 is the Relational Database Diagram and the PART3.SQL script generated in PART 3.

**Deliverable for Lab Project Part 4:**

There are two deliverables for Lab Project Part 4:

* 1. The Database Individual project report. This document contains your final report with all the requirements from Part 1, Part 2, Part 3, and Part 4. The Name of this document should be:

LASTNAME\_FIRSTNAME\_DB\_Proj.docx

* 1. The PART4.sql script

This script contains the SQL DDL statements to create your database schema, the SQL DML statements to populate data in your tables, and the queries required as part of this deliverable.

**Part 4 Requirements:**

1. Make sure the tables have Primary Keys (PK’s), the Natural Keys are enforced by a UNIQUE key constraint. Create CHECK constraints where needed. The constraints that cannot be implemented using primary keys (PK’s), foreign keys (FK’s), UNIQUE constraints (UK’s), and CHECK constraints need to be enforced using triggers.
2. Write done a justification for the PK’s, FK’s, UNIQUE constraints, and CHECK constraints.
3. Create a non-unique INDEX for all the foreign keys to optimize natural joins.
4. Analyze each table to make sure that they are in 3rd Normal Form and BCNF Normal Form (See Chapter 6 and the Normalization presentation).
5. Write the level of normalization for each table and state in your report why it is an acceptable level of normalization.
6. Create Oracle Sequences. One for each table that has a surrogate key. The name should be TableName\_ID\_SEQ.
7. Make sure the table’s Primary keys are Surrogate or artificial Keys.
8. Create a UNIQUE and a NOT NULL constraint for all the NATURAL KEYS that were previously the PK’s. They are now ALTERNATE KEYS and these constraints make sure that this important business rule is enforced through the life of the system.
9. Create UNIQUE constraints and NOT NULL constraint for all other alternate keys in your database schema.
10. Create INSERT statements to populate the tables with at least 15 rows of data. See the HR and the Beer Drinkers databases for examples.
11. In the INSERT statements, the surrogate key must be created using the Oracle sequence created in for the table (Sequence\_Name.NEXTVAL). See the “Beer Drinkers Database Schema” example where **Sequence\_Name.NEXTVAL** is userd to generate the surrogate key value.
12. You need to avoid referential integrity constraint errors by INSERTing the data into the PARENT tables first, then you INSERT into the CHILD tables. Another strategy used in the script **“Beer\_Drinkers\_schema\_w\_triggers.sql” is** to DISABLE the foreign keys (referential integrity constraints) before you insert the data and then proceed to ENABLE back the foreign keys.
13. Submit at least 10 SELECT statements including at least 3 that involve JOINs, 3 that involve subqueries, and 2 involving GROUP BY/HAVING statements.
14. The SQL script for Part 4 should be named:

PART4.sql

**Database Project Part 5**

**More on SQL Select, SQL DML, Row Level Triggers, and Final Project Report**

**Deliverable for Lab Project Part 5:**

There are two deliverables for Lab Project Part 5:

* 1. The Final Database Individual Project report. This document contains your final report with all the requirements from Part 1, Part 2, Part 3, Part 4 and Part 5. The Name of this document should be:

**GroupName\_DB\_Proj.docx**

* 1. The PART5.sql script

This script contains the SQL DDL statements to create your database schema, the SQL DML statements to populate data in your tables, other PL/SQL and triggers, and the queries required as part of this project.

**Part 5 Requirements:**

1. Write a Database Triggers to write into a history table the rows from your most critical table before an update. This will be a row level trigger that will save the rows that are being updated before the update occurs. For example, for a table EMPLOYEES, you add a table EMPLOYEES\_HIST that has the same schema as the table EMPLOYEES an contains the row that is about to be changed (the one contained in the “:old” record in a before update row level trigger).
2. Add audit columns to at least three tables in your database schema. The new columns will be CREATED\_DATE, CREATED\_BY, UPDATED\_DATE, UPDATED\_BY. The values for these three columns need to be generated by a ROW LEVEL TRIGGER after all INSERTS and UPDATES. See the **script “beer\_Drinkers\_schema\_w\_triggers.sql”.**
3. SQL SCRIPT for Part 5:

Part5.SQL

1. Final Project Report: A compilation of all the activities and scripts from Parts 1 to 5 with explanation for all the work that has been accomplished. This report is a project completion report written as if you’re presenting your work to the project sponsor.

Turn in a single MS Word document using APA Style in-text citations and references, with appendixes as needed. The name of the final report document must be as follows:

**GroupName\_DB\_Proj.docx**

