

## COMP810 Data Warehousing and Big Data

### Lab – Week 3 (Data Warehousing)

Due: ---

On successful completion of this paper students will be able to:

- (a) create tables (schemas) using SQL,
- (b) Understand table (schemes) relationships – snowflake, star, constellation, and
- (c) Perform advanced SQL queries.

#### **Task 1.** (See Week1 lecture slides and file: Refresher Slides\_Creating Tables.pdf )

HINT: Parent tables need to be created before the child table

- 1- Write SQL code to create a table called **STUDENT** with the following attributes:

NAME	NULL	TYPE
STUDENT_NO (PK)	NOT NULL	NUMBER (6)
STUDENT_NAME	NOT NULL	VARCHAR2 (20)
STUDENT_ADDRESS		VARCHAR2 (50)

- 2- Write SQL code to create a table name **COURSE** with the following attributes:

NAME	NULL	TYPE
COURSE_NO (PK)	NOT NULL	NUMBER (3)
COURSE_NAME	NOT NULL	VARCHAR2 (40)
COURSE_DETAILS		VARCHAR2 (50)

- 3- Write SQL code to create a table name **GRADE** with the following attributes:

NAME	NULL	TYPE
STUDENT_NO (PK) (FK)		NUMBER (6)
COURSE_NO (PK) (FK)		NUMBER (3)
GRADE	NOT NULL	NUMBER (3)

Note: PK -> primary key; FK -> foreign key; the "GRADE" table has a composite primary key of (STUDENT\_NO, COURSE\_NO). However, each of these attributes "individually" is a foreign key that references the table that each is related to. Apply all necessary constraints (primary and foreign keys).

#### **Task 2.**

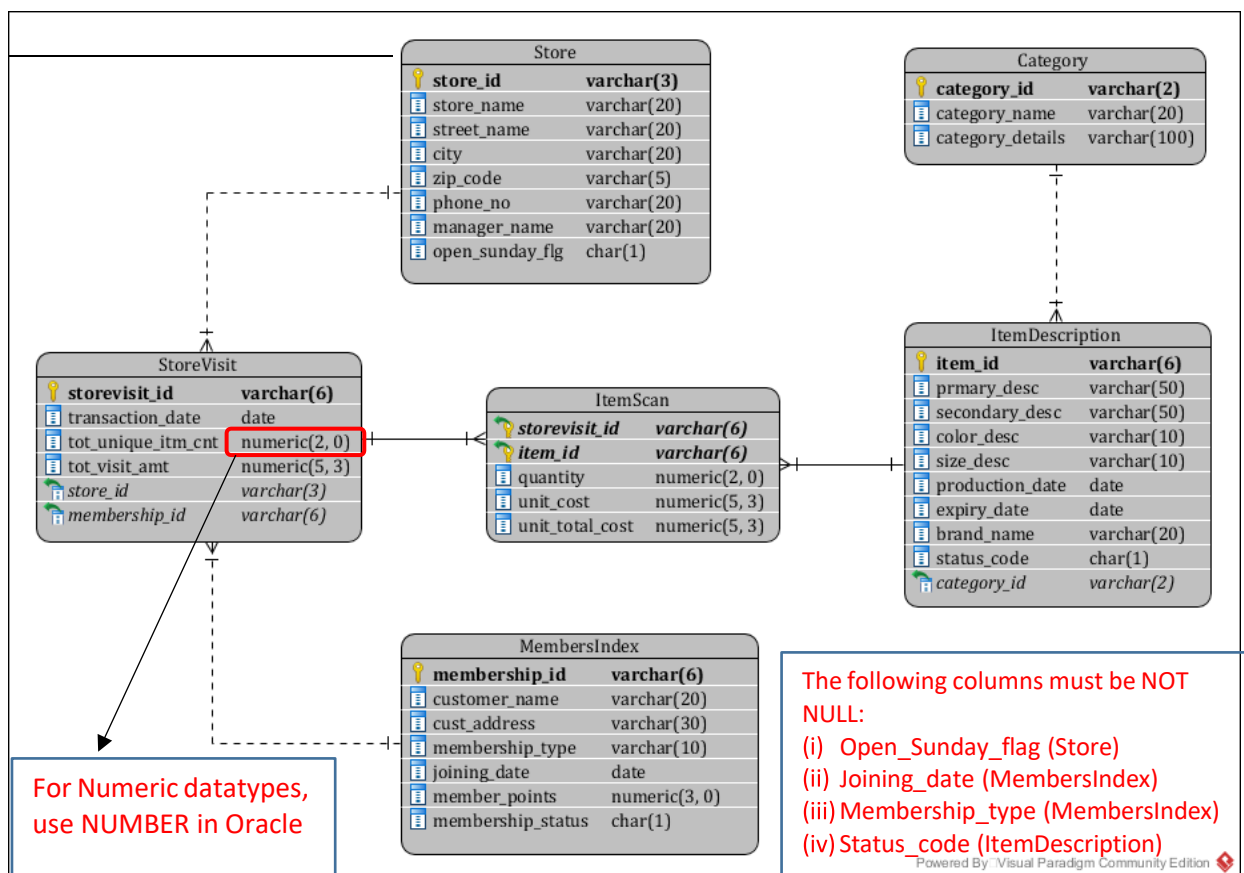
- 1. Run scripts HR.sql and Hotel\_DB.sql from Week 1. We will work with tables 'guest', 'hotel' and 'booking'.
- 2. Display the number of bookings made in each city.
- 3. Display the number of bookings made by each guest.
- 4. Display the number of bookings made by each guest in each city.

## Task 3. (See Week1 lecture slides and file: Refresher Slides\_Creating Tables.pdf )

Based on the ER model given below, **write SQL code** to create tables for each of the entities shown in the model. Apply all primary and foreign key constraints.

- Use naming rules and guidelines for table and column names. Define datatypes.
- Name all constraints (e.g. primary key, foreign key, NOT NULL) using a consistent naming convention<sup>1</sup>.
- Constraints can be defined at the time of table creation or after the table has created using ALTER TABLE command.
- Make sure you have created the parent table first before creating the child table. For example, you must create the **Store** table before creating the **StoreVisit** table.

Visual Paradigm Notation	
	Primary key
	Foreign key
	Both primary and foreign key
	Single line indicates both mandatory participation and maximum number of occurrences with the related entity is one.



<sup>1</sup> If you do not name your constraint, Oracle generates a name with the format SYS\_Cn, where n is an integer to create a unique constraint name.