# **Database Design Term Project**

**Student Name 1: Sneha Sahani Student ID: C0740756**

**Student Name 2: Vineela Thakkalapally Student ID: C0747617**

# **SUBMISSION:**

# **Submit your project as pdf or MS Word (without compression) Plus the file of query you used to analyze and create your database. You have to state your approach clearly on your report.**

# **PLEASE don’t use winRaR.**

# **Note: This project must consist of your own work completed during this semester**

# **Part 1:**

# **Requirements Analysis**

# Overview of Company and Product(s):

Provide a one page document entitles "Company Overview" that identifies:

a. Your name & student number

b. Your company

c. Type of company (example wholesaler or retailer)

d. Short description of the company

e. Product

i. Identify at least 6 characteristics (attributes), such as id or code, type, description, size, color, price, and so on

Customer Sales Invoice:

Modify the sample Customer Sales Invoice so that it reflects your company and the company's product. Although the company (and website) can advertise several products.

**Company Overview**

1. **Student Name 1: Sneha Sahani Student ID: C0740756**

**Student Name 2: Vineela Thakkalapally Student ID: C0747617**

1. **Sugar Dough**
2. **It’s a Retailer**
3. **Welcome to Sugar Dough Bakery, Toronto’s best bakery, stop in to check out our sweet treats!**

**Our bakery and store-front hours are Monday-Friday 8:30-8:00pm and Saturday’s 10:00-4:00pm. Our phone and order hours are Monday-Friday 8:30-8:00pm and Saturday’s 10:00-4:00pm. Whether you are searching for a special birthday cake or simply looking for a sweet ending to an everyday meal, Sugar Dough offers something for everyone. This is a one of a kind bakery, where the aroma of**

**home-style baking weakens even the strongest will. Sugar Dough offers a wide variety of freshly baked goods daily; gourmet cookies, fresh apple cake, cheesecake slices, chocolate eclairs, strawberry torte, pound cake, lemon bars, decadent brownies, shortbread, and over 20 flavors of cupcakes that no one can resist.**

1. **Product:**
   * 1. **Product\_id**
     2. **Product\_Name**
     3. **Product\_Description**
     4. **Product\_Categroy**
     5. **Product\_Quantity**
     6. **UnitPrice**

# **Part 2:**

# **Conceptual Design**

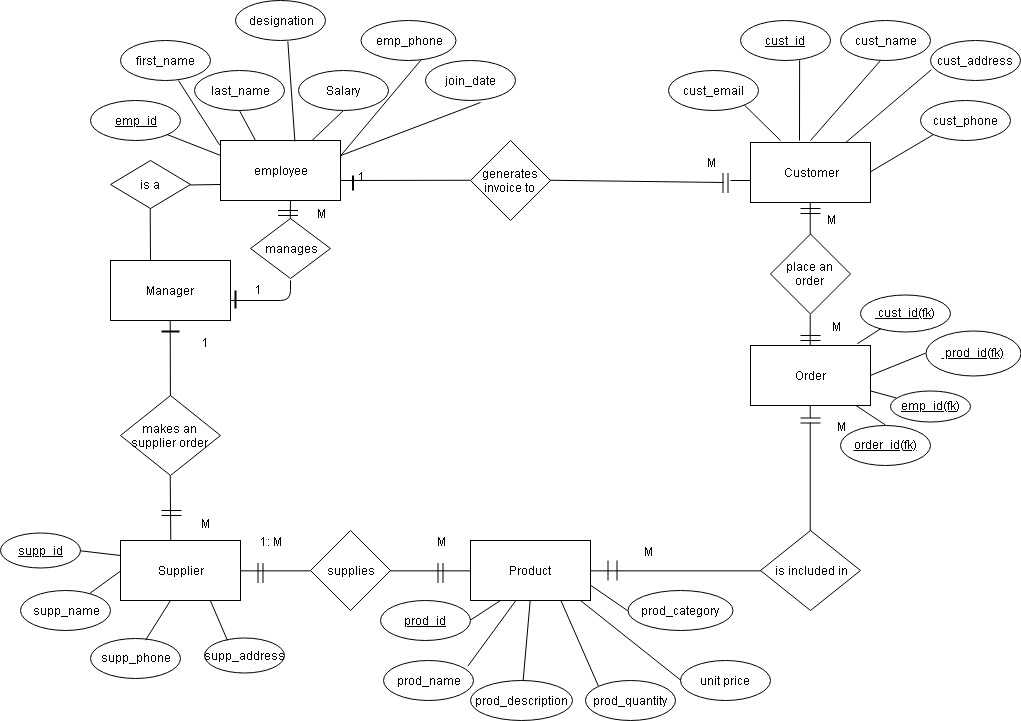
1. Create an ER diagram based on the customer invoice. Only include the entities identified

in the entity & attribute identification:

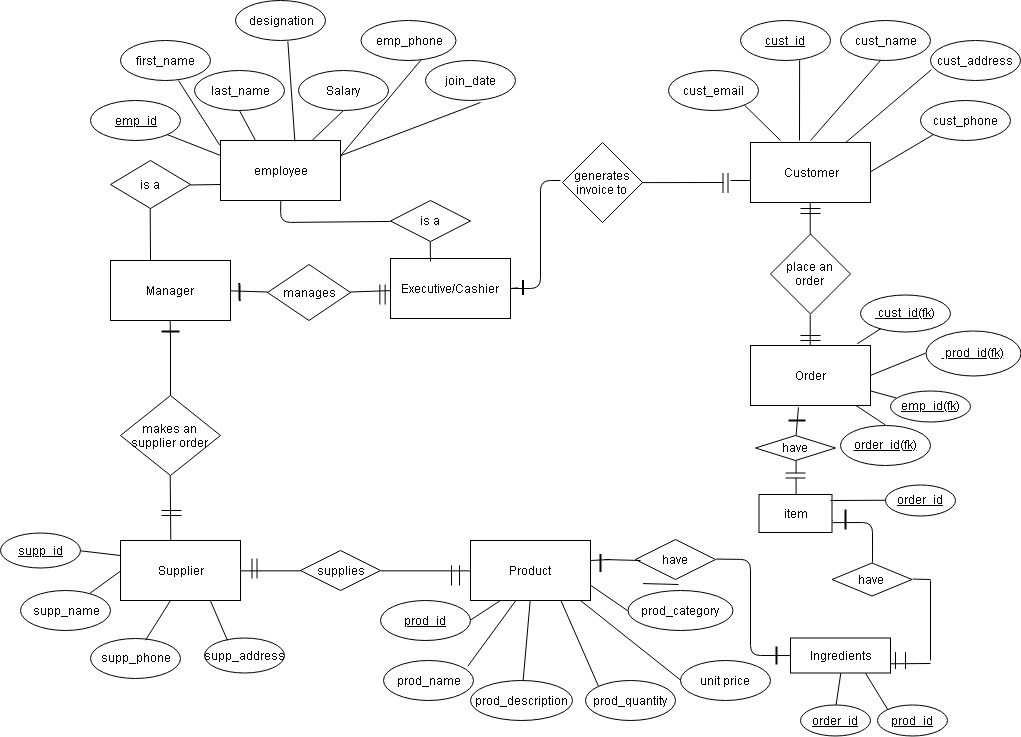
1. Entities
2. Attributes
3. Unique Identifiers
4. Named Relationships
5. Optionality and Cardinality
6. Foreign Keys
7. Create a second ER diagram that resolves many-to-many relationships

**ER DIAGRAM**

1.

****

2.

****

# **Part 3:**

**Logical Model**

Normalization & Relational Schema:

1. Create a document called "Database Normalization"
2. Normalize your database to 3NF
3. Modify the ER diagram if necessary
4. Include a relational schema for each entity

**DATABASE NORMALIZATION**

**1NF:**

employee [ emp\_id, first\_name, last\_name, designation, salary, emp\_phone, join\_date, order\_id, supp\_id, supp\_name, quantity, order\_confirmation, order\_date, supp\_phone, supp\_address, payement\_method, subtotal, total]

customer [ cust\_id, name, address, phone, email, payment\_method, tax, subtotal, total, date\_recieved, quantity, prod\_id, prod\_name, description, unit\_price, prod\_category]

**2NF:**

employee [ emp\_id, first\_name, last\_name, designation, salary, emp\_phone, join\_date]

suppliers [supp\_id, supp\_name, quantity, supp\_address, order\_date, supp\_phone, order\_confirmation]

customer [cust\_id, name, address, phone, email]

invoice\_table [order\_id, payment\_method, tax, subtotal, date\_recieved, quantity]

product [prod\_id, prod\_name, description, unit\_price, prod\_category]

**3NF:**

**employee [emp\_id, first\_name, last\_name, designation, salary, emp\_phone, join\_date]**

**customer [cust\_id, name, address, phone, email]**

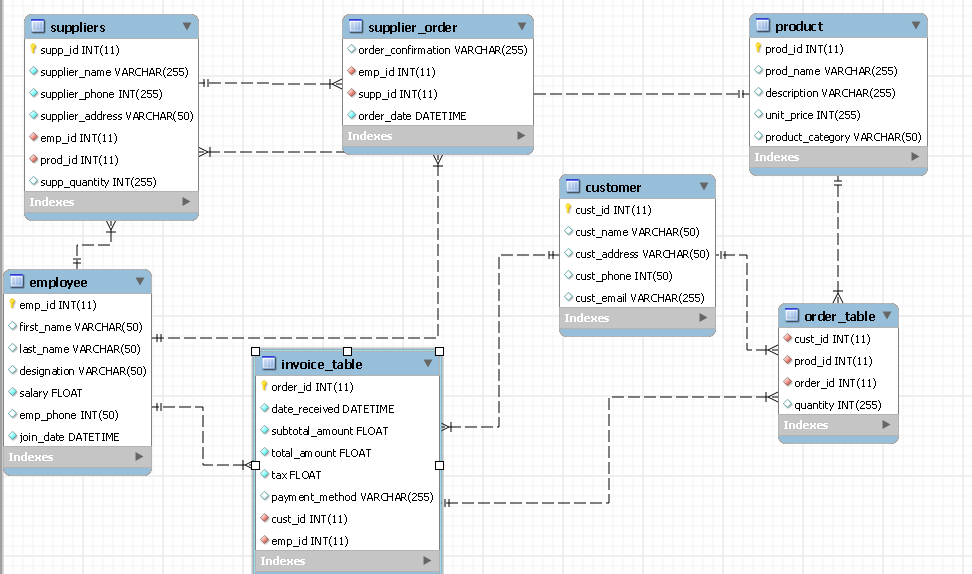
**product [prod\_id, prod\_name, description, unit\_price, prod\_category]**

**invoice\_table [order\_id, date\_recieved, subtotal, total, tax, payment\_method, cust\_id, emp\_id]**

**Order\_table [cust\_id, prod\_id, order\_id, quantity]**

**supplier [supp\_id, supp\_name, supp\_phone, supp\_address, emp\_id, prod\_id]**

**supplier\_order [order\_confirmation, emp\_id, supp\_id, order\_date]**



# **Part 4:**

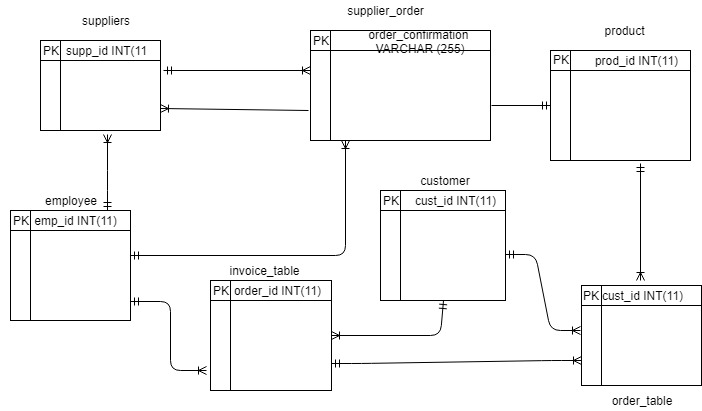
**Physical Model**

Physical Table Representation:

1. Convert the logical model to a physical model using a table format (not implementation)
2. Create a table for each physical database table

Include key type, optionality, column name, data type, length

**PHYSICAL MODEL**

****

# **Part 5:**

**Implementation of Physical Model**

Create Database Tables:

1. Construct CREATE statements for each table

2. Include IDENTITY and SEQUENCE columns (optional)

3. Construct INSERT statements and populate each table with at least 10 rows

## Identify Business Rules/Database Constraints:

1. Create a document called "Business Rules/Database Constraints"

a. List the physical database rules/constraints for:

* NOT NULL
* DEFAULT
* Primary keys
* Unique keys
* Foreign keys
* Check constraints

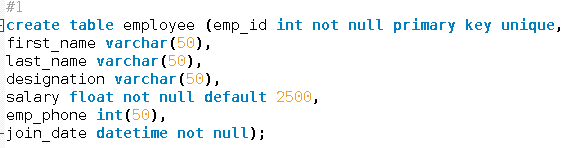
1. List at least 3 business rules that must be programmed
2. Implement the constraints into the database creation statements

## **Constraint Testing:**

Test at least 5 constraint and provide documentation that each constraint works.

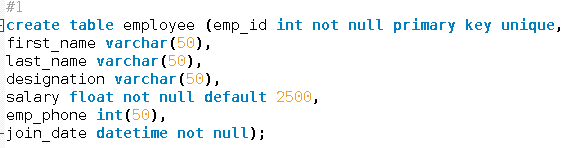
**BUSINESS RULES/DATABASE CONSTRAINTS**

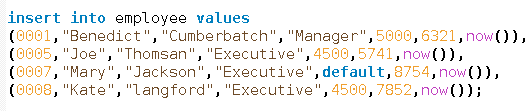
1. NOT NULL

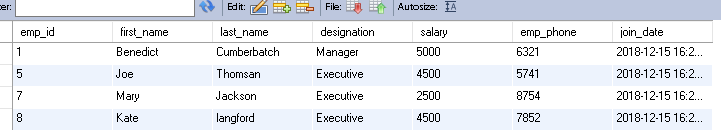




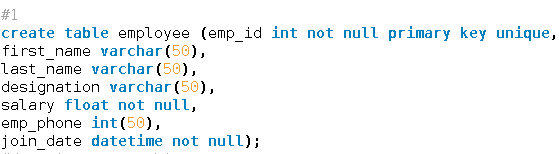
1. DEFAULT



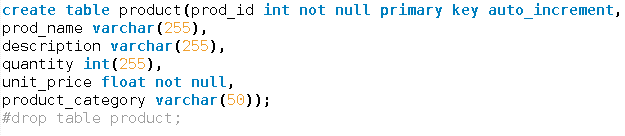




1. PRIMARY KEY

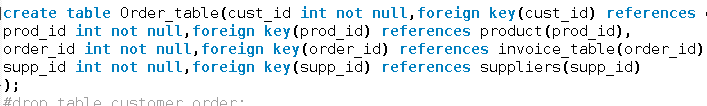


1. The PRIMARY KEY constraint uniquely identifies each record in a table.
2. Primary keys must contain UNIQUE values, and cannot contain NULL values.
3. A table can have only one primary key, which may consist of single or multiple fields
4. FOREIGN KEY

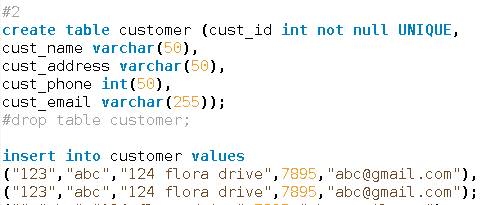


Foreign key is used to link two tables together.

Foreign key is a field in one table that refers to primary key in another table.



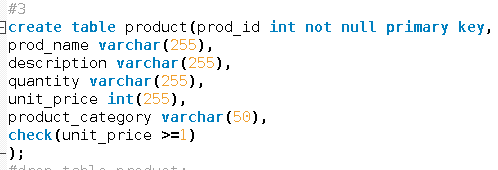
1. UNIQUE KEY

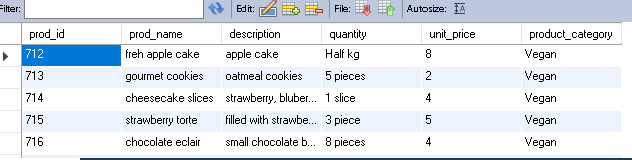




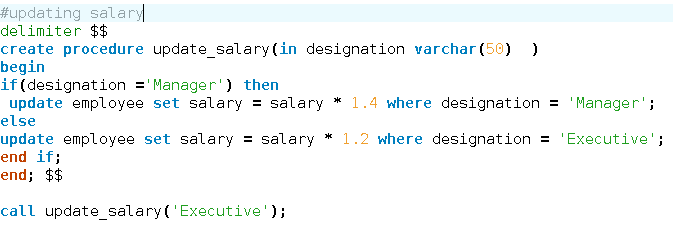
1. CHECK CONSTRAINTS

Check constraint will not allow the customer to add value less than 1

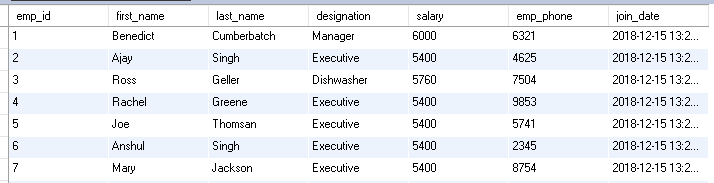




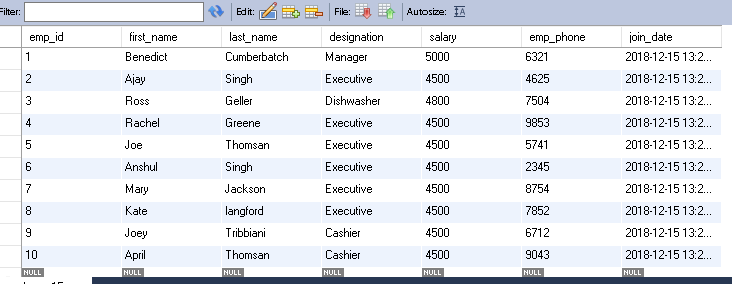
**Procedure for updating salary**



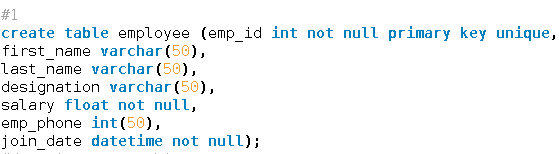
**Updated salary**



**Older salary**

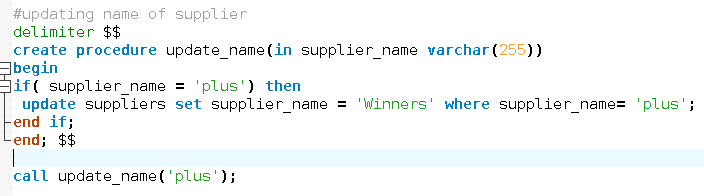
****

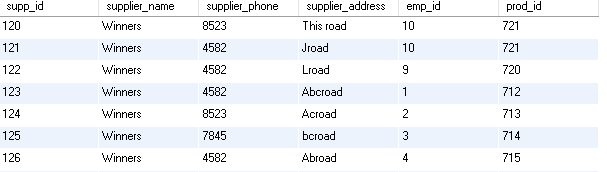
**Value must not be null**





**Procedure for updating supplier\_name:**





# **SUBMISSION:**

# **Submit your project as pdf or MS Word.(without compression)**

PLEASE don’t use winRaR.

A proper report has to be submitted including all the section with all the code, queries, etc.

You must email your project by Dec 14th 2018. Late submission will be 25% mark deduction per day.