

Project 1

RESTAURANT MANAGEMENT SYSTEMS

Introduction

Jacob decides to open up a little restaurant that sells his favourite foods like Chicken Biryani, Parotta and Chicken fry, Mutton Pepper fry etc. The restaurant has captured some very basic data from their few months of operation but have no idea how to use their data to help them to run the business.

Problem Statement

Jacob wants to use the data to answer a few simple questions about his customers, especially about their visiting patterns, how much money they've spent and also which menu items are their favourite. Having this deeper connection with his customers will help him deliver a better and more personalised experience for his loyal customers. He plans on using these insights to help him decide whether he should expand the existing customer loyalty program - additionally he needs help to generate some basic datasets so his team can easily inspect the data without needing to use SQL. Jacob has provided a sample of his overall customer data to write SQL queries to help him answer his questions! Jacob has shared with you 4 key datasets for this case study:

- Customer
- Sales
- Menu
- Members

Advantages

- The system automates the manual procedure of managing restaurant activities.
- Customers can view their daily food items and order easily.
- The system is convenient and flexible to be used.
- It saves their time, efforts, money and resources.

Disadvantages

- The admin has to manually keep updating the information by entering the details in the system.
- This project we are using java swing. So it is support only desktop application. It is not update for internet application.

Project 1

System Specifications

H/W Requirement

- Core i3 processor
- 2GB Ram
- 20GB of hard disk space in terminal machines
- 1 TB hard disk space in Server Machine

S/W Requirement

- Windows 7 or above operating system
- JDK 1.8 and above.
- MySQL workbench

The Programmer help to the Jacob for his Restaurant Management Systems. He decide to create a database called RESTAURANT. Hence inside the database the programmer create above 4 Tables which has the following attributes:

TABLE - CUSTOMER

FIELD(OR)COLUMN	DATATYPE	CONSTRAINT
CUSTOMER_ID	VARCHAR	PRIMARY KEY
CUSTOMER_NAME	VARCHAR	NOT NULL
GENDER	VARCHAR	NOT NULL
EMAIL	VARCHAR	NOT NULL
ADDRESS	VARCHAR	NOT NULL
MOBILE_NUMBER	BIGINT	NOT NULL

TABLE – SALES

FIELD(OR)COLUMN	DATATYPE	CONSTRAINT
CUSTOMER_ID	VARCHAR(45)	FOREIGN KEY
ORDER_DATE	DATE	NOT NULL
PRODUCT_ID	INTEGER	FOREIGN KEY

TABLE – MEMBERS

FIELD(OR)COLUMN	DATATYPE	CONSTRAINT
CUSTOMER_ID	VARCHAR(45)	PRIMARY KEY
JOIN_DATE	TIMESTAMP	NOT NULL

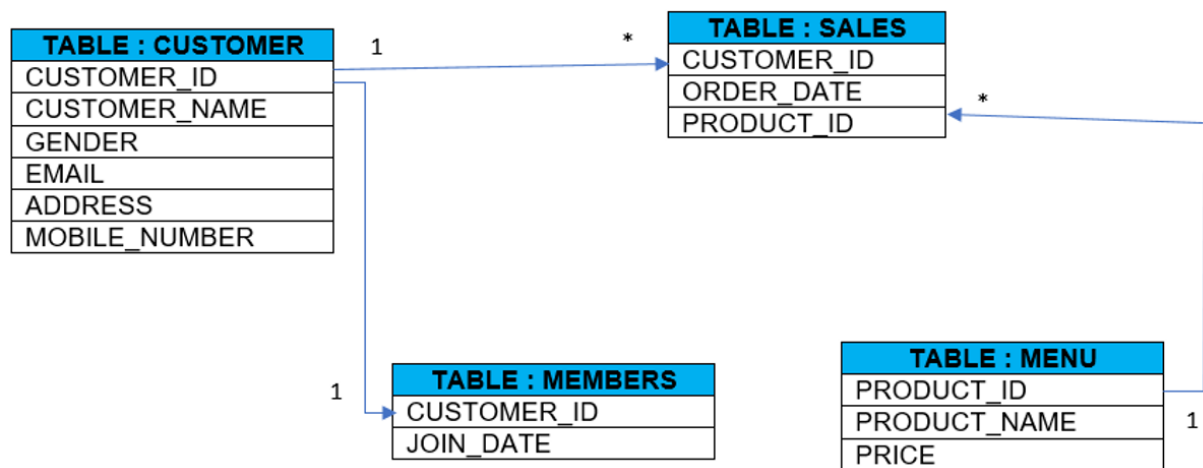
Project 1

TABLE – MENU

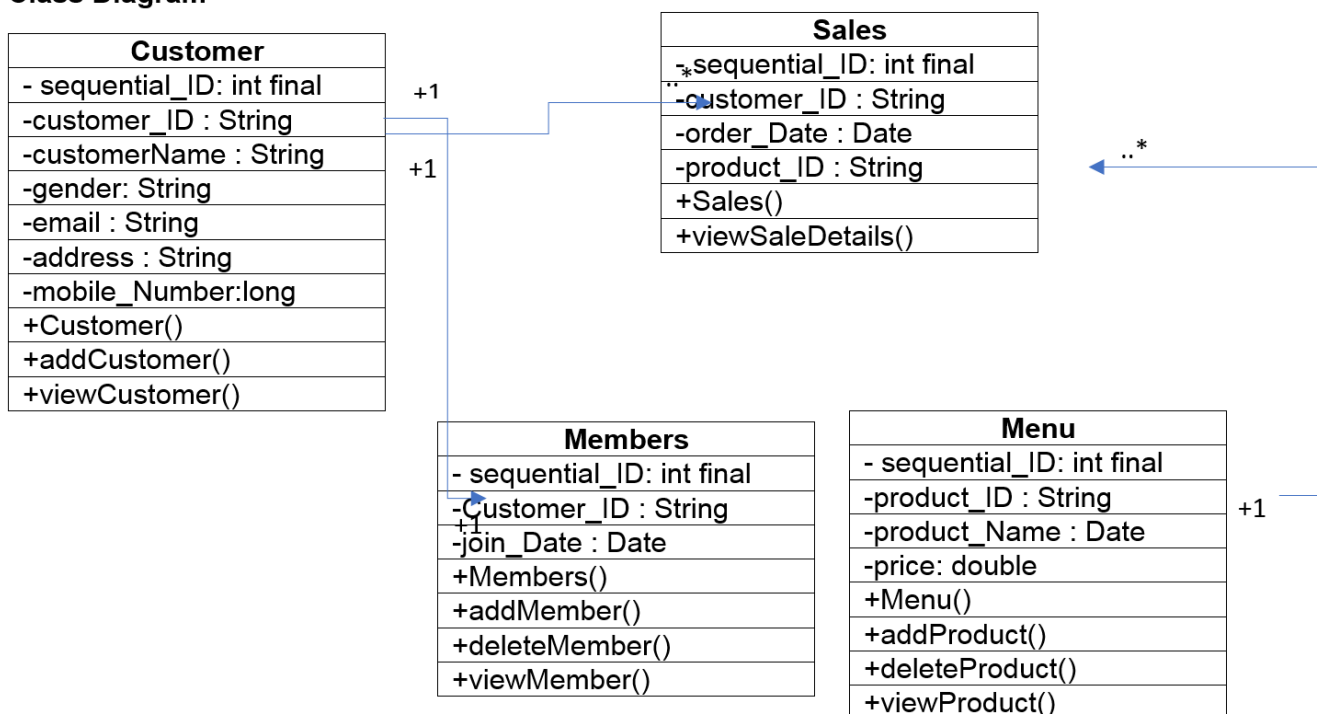
FIELD(OR)COLUMN	DATATYPE	CONSTRAINT
PRODUCT_ID	VARCHAR(45)	PRIMARY KEY
PRODUCT_NAME	VARCHAR(45)	NOT NULL
PRICE	DECIMAL	NOT NULL

You can inspect the Entity Relationship diagram and example data below

ER- DIAGRAM



Class Diagram



Project 1

The Project has 2 Module:

- 1) Admin Module
- 2) User Module

1) Admin Module

From the Admin Module, the Administrator access and control overall the Project Management. He Login to the Project. He maintain Customer, Sales, Members, Menu. In the Customer Module the Administrator Maintain Customer_Details. The Customer_ID is **UNIQUE** and Auto-Generated.

Through Sales Module the Administrator maintain Sales_Details. The SALE Table maps the **FOREIGN KEY REFERENCES** of the Customer_ID associate with the Customer Table. Also maps the **FOREIGN KEY REFERENCES** of the Product_ID associate with the Menu Table. The Customer to Sales and Menu to Sales Maintain One to Many Relationship.

Through Menu Module the Administrator Add the New Product. The Product_ID is **UNIQUE** and auto-generated. The administrator maintain the Product_Details.

Through the Members Module the Administrator Join the customers to the MEMBERS Table. The Customer and Members Table in the Database has One to One Relationship.

2) User Module

In the User Module the Customer Login to the Project through his CUSTOMER_ID. He view the Product and their prices from the Menu Table. The Customer order the Product. The Order_Date Column in the Customer Table should be CURRENT_SYSTEM_DATE.

Due to the Complexity of the Project the Programmer Divide the Project Module in the following Packages.

Packages

A) com.model

- 1) Customer
- 2) Sales
- 3) Members
- 4) Menu

B) com.dbutil

DBConnection – create a static method getConnection() which return the CONNECTION interface object.

Project 1

C) com.gui

- 1) LoginFrame
- 2) CustomerFrame
- 3) SalesFrame
- 4) MembersFrame
- 5) MenuFrame
- 6) ViewCustomer
- 7) ViewSales
- 8) ViewMembers
- 9) ViewMenu

D) com.sqlquery

In this package include list of sql files(DDL Statement-CREATE TABLE COMMANDS) and ER Diagram.

SWING.FRAME

- 1) LoginFrame
- 2) CustomerFrame
- 3) SalesFrame
- 4) MembersFrame
- 5) MenuFrame

SWING.TABLE

- 1) ViewCustomer
- 2) ViewSales
- 3) ViewMembers
- 4) ViewMenu

Note: Show the result tables in the swing table.