FINAL PROJECT

Damir Amankos 210107051

Database for Cake shop

Description:

1. **"The System for Ordering" Database project: The prime motive of this database is to make the order transaction speedy and smooth for the customers, to maintain and secure the order records. The system is capable enough to book the cake as per the customer need and event.**

**A customer can register to purchase an item. The customer will provide the account number and bank name.**

**After registration, each customer will have a unique customer, user id, and password.**

**A customer can purchase one or more items in different quantities.**

End users: **Casual End Users:** programmers of the company **Parametric end users:** Reservation clerks basically check availability for a given request, check whether the requested product is in stock

**Parametric end users:** Clerks who are working at receiving end for company enter the product identifies via barcodes and descriptive information

## Data



through buttons to update a central database

the best solution is to partition tables by the **key** which differentiates stale from current data (such as **date, currency\_id** or things like that).

obsolescence:

Setting up daily obsolescence for on any rows retrieved by **queries**

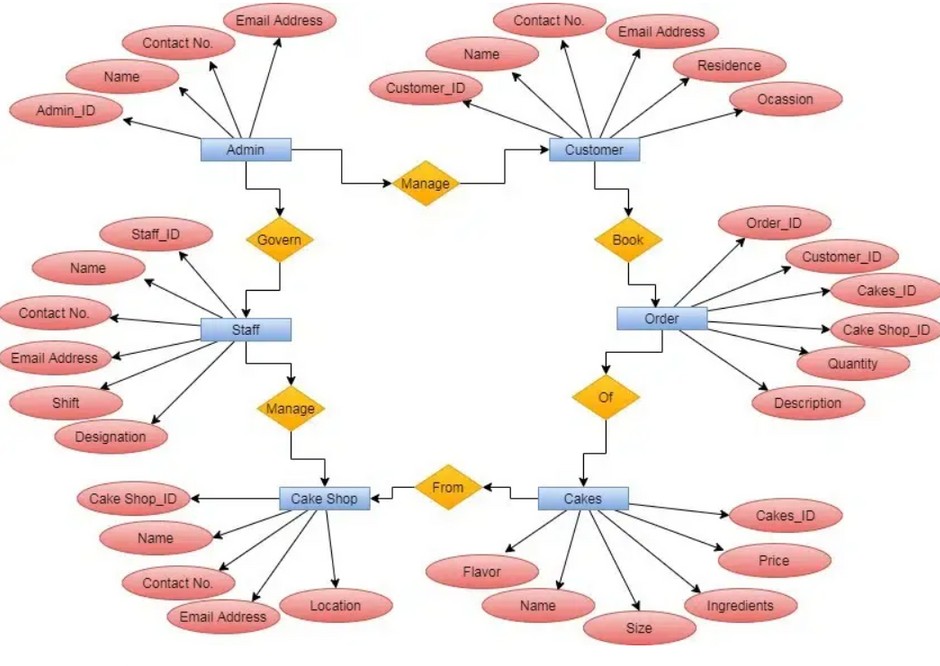


Project Idea: The Database System increasing Walmart Sales

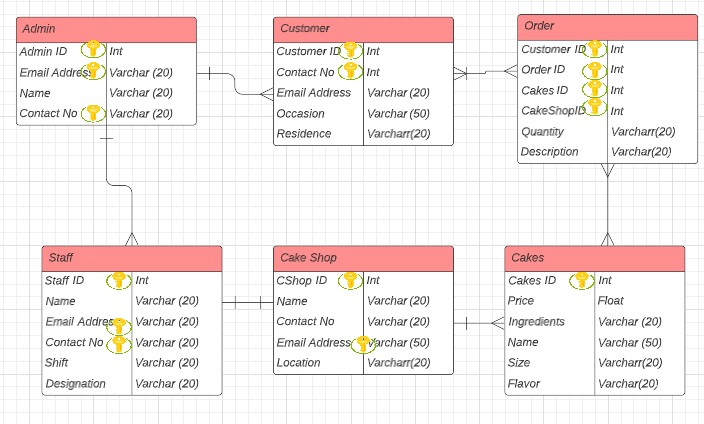
Environment

# Entity Relationship Diagram

Admin-Customer: 1 to Customer-Order: n to Order-Cakes: n to Cake shop-Cakes: 1 to Cake shop-staff: 1 to Admin-staff: 1 to



Relationships:



n n n n 1

n

# Normalization

### Functional dependencies:

1. CUSTOMER ID--->CONTACT NO, NAME, EMAIL ADDRESS, RESIDENCE, OCCASION

SUPERKEYS:(CUSTOMER\_ID); (CUSTOMER\_ID, CONTACT\_NO); (CUSTOMER\_ID,NAME); (CUSTOMER\_ID,EMAIL)

1. ORDER ID--->CUSTOMER ID, CAKES ID, CAKE SHOP ID, QUANTITY, DESCRIPTION

SUPERKEYS:(ORDER\_ID); (ORDER\_ID, CUSTOMER\_ID))

1. CAKES ID--->PRICE, INGREDIENTS, SIZE, NAME, FLAVOR

SIZE---> INGREDIENTS, PRICE

SUPERKEYS:CAKES\_ID

NON-PRIME--->NON-PRIME, SO IT IS TRANSITIVE DEPENDENCY

**table 1**

**table 2**

Size

### Name Flavor

Price Ingr

C\_ID

1. CAKE\_SHOP\_ID--->NAME, CONTACT\_NO, LOCATION, EMAIL ADDRESS NAME---> LOCATION

SUPERKEYS:(CAKE\_SHOP ID); (CAKE SHOP ID, NAME))

C\_ID Name Flavor

Size Price

Sh\_ID

**table 1**

**table 2**

name

email

loctn

c\_no

1. STUFF\_ID--->NAME, CONTACT\_NO, SHIFT, DESIGNATION, EMAIL ADDRESS DESIGNATION---> SHIFT

SUPERKEYS:(STUFF ID); (STUFF ID, NAME); (STUFF\_ID,CONTACT\_NO)

1. ADMIN\_ID--->NAME, CONTACT\_NO, EMAIL ADDRESS

**table 1**

**table 2**

### S\_ID Name Email Dsgntn Shift

SUPERKEYS:(ADMIN ID); (ADMIN ID, NAME); (ADMIN\_ID,CONTACT\_NO)

**create table Customer ( Customer\_ID VARCHAR(50), Name VARCHAR(50),**

**Contact\_No VARCHAR(50), Email Address VARCHAR(50), Residence VARCHAR(50), Occasion VARCHAR(50)**

Table

**);**

**insert into Customer (Customer\_ID, Name, Contact\_No, Email Address, Residence, Occasion) values ('223.144.94.79/25', 'Galvan Sea', '612-931-9772', 'gsea0@unblog.fr', '459 Crest Line Street', 'Action|Horror');**

**insert into Customer (Customer\_ID, Name, Contact\_No, Email Address, Residence, Occasion) values ('71.71.42.246/27', 'Danika Lockhart', '737-705- 6839', 'dlockhart1@elpais.com', '66 Almo Trail', 'Documentary');............................**

**create table Order (**

**Order\_ID VARCHAR(50),**

**Customer\_ID VARCHAR(50), Cakes\_ID VARCHAR(50),**

**Cake\_Shop\_ID VARCHAR(50), Quantity INT,**

**Description VARCHAR(50)**

**);**

**insert into Customer (Order\_ID, Customer\_ID, Cakes\_ID, Cake\_Shop\_ID, Quantity, Description) values ('7.170.171.212/14', '228.90.207.232/11', 'ac6:eba2:7966:3929:a8a3:937:a068:cd54/57', '2e10:5faa:afaa:f2ab:1416:d356:e56d:f204/47', 7, 'Horror|Thriller');**

**insert into Customer (Order\_ID, Customer\_ID, Cakes\_ID, Cake\_Shop\_ID, Quantity, Description) values ('205.207.245.69/25', '200.190.210.1/16', 'f445:582d:6514:6335:8bd9:9b08:6f32:2aab/15', '6de6:60a3:579f:defa:77ec:91fb:e7a:33ce/80', 10, 'Comedy|Crime|Horror|Thriller');**

**create table Admin (**

**Admin\_ID VARCHAR(50),**

**Contact\_No VARCHAR(50), Email\_Address VARCHAR(50), Name VARCHAR(50)**

**);**

**insert into Admin (Admin\_ID, Contact\_No, Email\_Address, Name) values ('152.54.14.234/18', '983-398-7120', 'tkensitt0@virginia.edu',**

**create table Cake Shop (**

**create table Cakes (**

**Cakes\_ID VARCHAR(50), Price VARCHAR(50),**

**Ingredients TEXT, Size VARCHAR(50), Name VARCHAR(50),**

**Flavor TEXT**

**);**

**insert into Customer (Cakes\_ID, Price, Ingredients, Size, Name, Flavor) values ('77.231.31.99/12', '$15.05', 'augue', '2XL', 'Treeflex', 'augue quam sollicitudin');**

**insert into Customer (Cakes\_ID, Price, Ingredients, Size, Name, Flavor) values ('115.220.42.170/18', '$21.13', 'vitae consectetuer eget', 'M', 'Fintone', 'quis tortor');**

**Cake\_Shop\_ID VARCHAR(50), Contact\_No VARCHAR(50), Email\_Address VARCHAR(50), Location VARCHAR(50),**

**Name VARCHAR(50)**

**);**

**insert into Cake Shop (Cake\_Shop\_ID, Contact\_No, Email\_Address, Location, Name) values ('120.161.59.175/23', '821-848-9157', 'abrinsford0@dagondesi '496 Sage Point', null);**

**create table Staff (**

**Staff\_ID VARCHAR(50),**

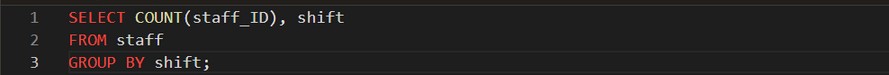
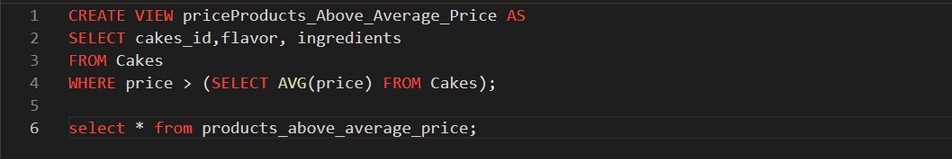
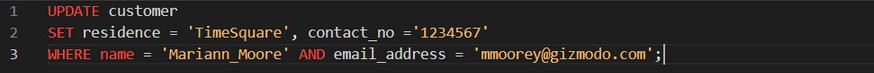
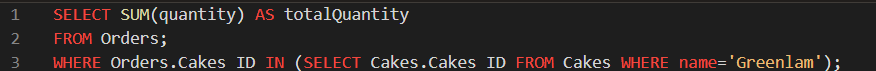
**Contact\_No VARCHAR(50), Email\_Address VARCHAR(50), Location VARCHAR(50),**

**Shift VARCHAR(50),**

**Designation VARCHAR(50)**

**);**

**insert into Staff (Staff\_ID, Contact\_No, Email\_Address, Location, Shift, Designation) values ('83.16.155.219/14', '317-573-4507', 'sbethune0@eventbrite.com', '45 Ilene Trail', 'Services', 'Weekly');**

 Query