

RETAIL POINT OF SALE DATABASE

DATA MODELLING FOR INFORMATION PROFESSIONALS - 5707

Term Project Final Report-FALL 2021

BY :

Vamsi Krishna Bharghava Muktinutalapati

Hemanth Gupta Kollipara

Lakshmi Naga Sai Veera Durga Uppala

Shanmukha Srinivas Muddana

Retail Point of Sale Database Project

Project Description

Retail stores are the general locations where customers are convenient to reach out for purchasing the goods and products offered by different suppliers, different categories at one place. Retailing has become an integral part of modern society and many people depends on retail stores to lead a comfortable life. Due to the advancement of technology, many of the retail stores kicked-off selling their products through online with the help of logistics department to deliver it to their customer houses.

As the Retail industry is evolving largely in every nook and corner of the world, there exists an importance of customer data, supplier data, supply chain data, orders data, etc. Maintaining transaction data, invoices on paper which is written cannot be stored for longer time and maintaining higher number customers data is impossible. To make this automated and hassle free to organize and utilize the data, we have designed this retail point of sale database.

Point of sale is a type of teller machine which performs transactions from the customers by card or cash and records each unique customers transaction data at the back-end database. As large amounts of products got sold every day in retail stores which is primarily located in cities, this point-of-sale teller machine also plays a key role in automated adjusting of inventory quantities. This automated manipulation helps retail stores to reduce the staff efforts in maintenance and prediction of stock quantities.

In this project, we have designed to maintain customers data, Invoice data of the customers, supplier's data, products data, product category data and vendors data. Vendor is a person who operates point-of-sale teller machine and generates invoices as per the customer purchased products.

From this database, we can collect the data of the customers and the products purchased by them. As data plays major role in finding the patterns and in development of the business, this project will be helpful for all the retail companies to perform Analysis. This project will also lessen the human resources cost to companies and maintains the data accurately which is not possible with human beings.

This data can be accessed only by the authorized persons, and it can also store the past and present data which would be useful for comparison analysis. This method can also have data backup facilities during the times of data losses or breaches.

Objectives:

- To digitalize all the customer, supplier, product, and invoice data.
- Details of Products and their Product Categories
- Details of Customer details and their membership towards retail store
- Information of vendor and his invoice generations
- Invoice information combination with products, customer, and payment information.

- Quantity of stock available for each unique product in a store can be seen from the product table
- Types of different categories of products available in the retail store and that data can be retrieved from the product category table
- Supplier information and contact details can be retrieved from the supplier table.

Scope :

1. Vendor can access the shopping cart of customers, invoices, information about all the products.
2. Vendor can access all the information like How many Invoices he generated, and the total amount billed on each day and Month.
3. Vendor can access information about the products that are added by the customers in the shopping cart.
4. The Invoice shows the product id, names of products, and discount on each product (if any promotions available).
5. Store Owners can access the invoices and know the information of customers who are frequently visiting and loyal to the store to offer special discounts.
6. System will generate the unique invoice numbers for each purchase and will be different invoices for the same customer with different transactions.
7. Each product is mapped to it's category.
8. One supplier can supply one or more number of products and vendor can have many numbers of suppliers.
9. System will generate the unique id for the shopping cart, customers, vendors, and invoices

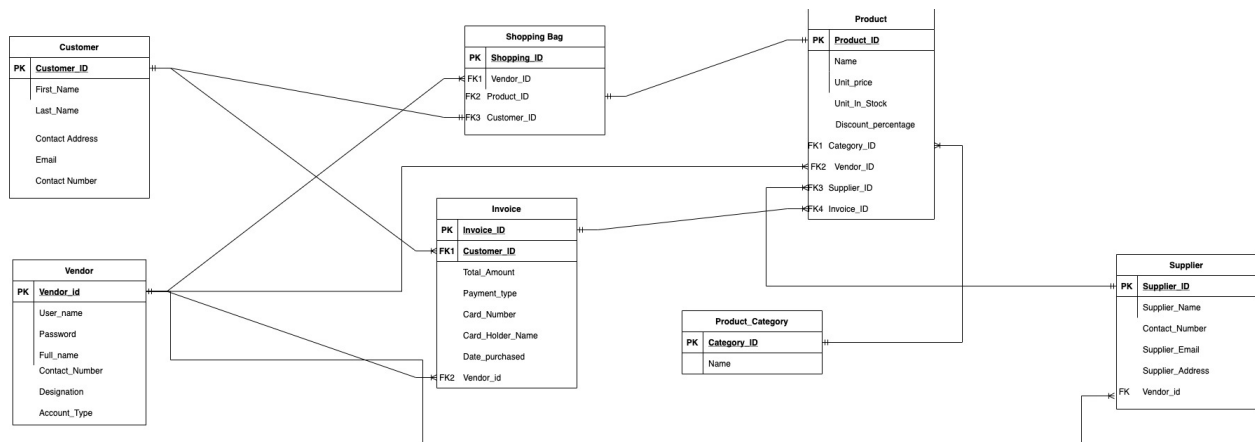
User Requirements:

1. Retail Store owner can calculate the total sales in a day from store.
2. Store owner can analyze data of sales from which the product category they are earning highest profits.
3. Store owner can monitor the stock quantities of each product from the product table.
4. Store owner can evaluate the performance of each vendor at pos as per the number of invoices generated in a day.
5. Store owner can maintain and monitor all the suppliers list from the supplier's table.
6. Price of each product can be traced from the product table.
7. Using databases, data can be stored, manipulated, and retrieved easily
8. Digitalization of records can be performed.
9. Vendor who operates the pos can be provided with restricted access to protect the transactional data.

Business Rules :

1. A customer can have many Invoices.
2. A vendor can process many Invoices.
3. A supplier can supply many products.
4. Each shopping ID can have one Invoice.
5. Unique Invoice is provided to one customer
6. Each product category can have many products
7. A customer can have only in e Shopping IDs
8. A vendor can process many shopping IDs.
9. A vendor can accept products from many suppliers
10. One shopping cart can have many products.

ERD Diagram:



Data Dictionary:

TABLE NAME	ATTRIBUTE NAME	CONTENTS	TYPE	FORMAT	RANGE	REQUIRED	PK OR FK	FK Referenced Table
------------	----------------	----------	------	--------	-------	----------	----------	---------------------

Customer	Customer_ID	Customer's ID Number	INT	999999	000000-999999	YES	PK	
	First_Name	Customer's First Name	VARCHAR(25)	Xxxxxxxxxx		YES		
	Last_Name	Customer's Last Name	VARCHAR(25)	Xxxxxxxxxx		No		
	Contact Address	Customer's Contact Address	VARCHAR(70)	Xxxxxxxxxx		YES		
	Email	Customer's Email	VARCHAR(25)	Xxxxxxxxxx		No		
	Contact Number	Customer's Contact Number	BIGINT	99999999999	0000000000-99999999999	YES		
Shopping_Bag	Shopping_ID	Shopping ID Number	INT	999999	000000-999999	YES	PK	Vendor Table
	Vendor_ID	Vendor ID Number	INT	999999	000000-999999	YES		
	Product_ID	Product ID Number	INT	999999	000000-999999	YES		
	Customer_ID	Customer ID Number	INT	999999	000000-999999	YES		
Product	Product_ID	Product ID Number	INT	999999	000000-999999	YES	PK	
	Name	Product Name	VARCHAR(20)	Xxxxxxxxxx		YES		
	Unit_price	Price of product	FLOAT	999999	000000-999999	YES		
	Unit_In_Stock	No.of Units availability	INT	999999	000000-999999	YES		
	Discount_percentage	Discount percentage of product	INT	999999	000000-999999	YES		
	Category_ID	Category ID Number	INT	999999	000000-999999	YES		
	Vendor_ID	Vendor ID Number	INT	999999	000000-999999	YES		
	Supplier_ID	Supplier ID Number	INT	999999	000000-999999	YES		
	Invoice_ID	Invoice ID Number	INT	999999	000000-999999	YES		
Vendor	Vendor_ID	Vendor ID Number	INT	999999	000000-999999	YES		

	User_name	User name of Vendor	VARCHAR(20)	Xxxxxxxx		YES		
	Password	Password of Vendor	VARCHAR(15)	Xxxxxxxx		YES		
	Full_name	Full name of Vendor	VARCHAR(50)	Xxxxxxxx		YES		
	Contact_Number	Contact number of Vendor	BIGINT	9999999999	0000000000-9999999999	YES		
	Designation	Designation of Vendor	VARCHAR(5)	Xxxxx		YES		
	Account_Type	Account type of Vendor	VARCHAR(5)	Xxxxx		YES		
Invoice	Invoice_ID	Invoice ID Number	INT	999999	000000-999999	YES	PK	Customer
	Customer_ID	Customer ID Number	INT	999999	000000-999999	YES	FK1	
	Total_Amount	Total amount of Invoice	FLOAT	999999	000000-999999	YES		
	Payment_Type	Payment type of Invoice	VARCHAR(5)	Xxxxx		YES		
	Card_Number	Card number of Customer	BIGINT	999999999999	0000000000-9999999999	YES		Vendor
	Card_Holder_Name	Name of Card Holder	VARCHAR(50)	999999999999	0000000000-9999999999	YES		
	Date_purchased	Purchased Date	DATE	MM-DD-YYYY		YES		
	Vendor_id	Vendor id number	INT	999999	000000-999999	YES	FK2	
Product_Category	Category_ID	Category ID number	INT	999999		YES	PK	
	Category Name	Name of Product	VARCHAR(20)	Xxxxxxxx		YES		
Supplier_Table	Supplier_ID	Supplier ID Number	INT	999999	000000-999999	YES	PK	Vendor
	Contact_Number	Contact Number of Supplier	BIGINT	999999999999	0000000000-9999999999	YES		
	Supplier_Email	Email of Supplier	VARCHAR(30)	Xxxxxxxx		YES		
	Supplier_Address	Address of Supplier	VARCHAR(50)	Xxxxxxxx		YES		
	Vendor_id	Vendor id number	INT	999999	000000-999999	YES	FK	

--	--	--	--	--	--	--	--	--

Query and It's operations for Customer Table

Customer Table:

```
create table Customer(
CustomerId int primary key identity(1000,1),
FirstName varchar(25),
LastName varchar(25),
ContactAddress Varchar(45),
Email varchar(30),
ContactNumber bigint
)
```

Insert Statements:

```
insert into Customer Values('Vamsi','Krishna','Denton','vkb@gmail.com',1234567891)
insert into Customer Values('Hemanth','Gupta','Denton','hg@gmail.com',128558867891)
insert into Customer Values('Durga','Uppala','Denton','du@gmail.com',6589745123)
insert into Customer Values('Dheeraj','Gammi','Texas','DJ@gmail.com',123658711)
insert into Customer Values('Ravi','Teja','California','RT@gmail.com',1285236891)
insert into Customer Values('Bala','Krishna','Hindupur','bk@gmail.com',189757891)
insert into Customer Values('Jai','Balayya','Banjarahills','nandamuri@gmail.com',11245891)
insert into Customer Values('NTR','Jr','JubileeHills','ntr9999@gmail.com',6984567891)
insert into Customer Values('NTR','Sr','Abids','Ntr@ntr.com',9999999991)
insert into Customer Values('Chiranjeevi','Konidela','Hyderabad','ck@gmail.com',187456991)
insert into Customer Values('kini','konna','London','pkb@gmail.com',3234567891)
insert into Customer Values('sai','Krishna','Denton','skb@gmail.com',1234547391)
insert into Customer Values('pranathi','nandu','California','pnk@gmail.com',1294567891)
insert into Customer Values('Ruthu','priya','Paris','rpk@gmail.com',1234567891)
insert into Customer Values('Mahesh','Babu','Gachibowli','mh@gmail.com',122566741)
insert into Customer Values('Pawan','Kalyan','Bhimavaram','pk@gmail.com',9369584491)
insert into Customer Values('Prabhas','Uppalapati','Rajahmundry','pu@gmail.com',1256978891)
insert into Customer Values('Nani','Krishna','Vijayawada','nkb@gmail.com',1234567891)
insert into Customer Values('Manu','Nk','Denton','mnk@gmail.com',9999967891)
insert into Customer Values('kelly','tisha','London','ktb@gmail.com',1199999991)
insert into Customer Values('Vivi','khanna','sydney','pkb@gmail.com',1230567891)
insert into Customer Values('Vanitha','Komna','Dallas','vab@gmail.com',1234567861)
insert into Customer Values('Pandu','K','guntur','pkg@gmail.com',9233567891)
insert into Customer Values('Geetha','Iana','London','lkb@gmail.com',1234567691)
```

Results:

Results		Messages				
	CustomerId	FirstName	LastName	ContactAddress	Email	ContactNumber
1	1000	Vamsi	Krishna	Denton	vkb@gmail.com	1234567891
2	1001	Sachin	Tendulkar	Mumbai	ten@gmail.com	2365897412
3	1002	Hemanth	Gupta	Denton	hg@gmail.com	128558867891
4	1003	Durga	Uppala	Denton	du@gmail.com	6589745123
5	1004	Dheeraj	Gammi	Texas	DJ@gmail.com	123658711
6	1005	Ravi	Teja	Califomia	RT@gmail.com	1285236891
7	1006	Bala	Krishna	Hindupur	bk@gmail.com	189757891
8	1007	Jai	Balayya	Banjarahills	nandamuri@g...	11245891
9	1008	NTR	Jr	JubileeHills	ntr9999@gma...	6984567891
10	1009	NTR	Sr	Abids	Ntr@ntr.com	999999991
11	1010	Chiranjee...	Konidela	Hyderabad	ck@gmail.com	187456991
12	1011	kini	konna	London	pkb@gmail.com	3234567891
13	1012	sai	Krishna	Denton	skb@gmail.com	1234547391
14	1013	pranathi	nandu	Califomia	pnk@gmail.com	1294567891
15	1014	Ruthu	priya	Paris	rp@gmail.com	1234567891
16	1015	Mahesh	Babu	Gachibowli	mh@gmail.com	122566741
17	1016	Pawan	Kalyan	Bhimavaram	pk@gmail.com	9369584491
18	1017	Prabhas	Uppalap...	Rajahmundry	pu@gmail.com	1256978891
19	1018	Nani	Krishna	Vijayawada	nkb@gmail.com	1234567891
20	1019	Manu	Nk	Denton	mnk@gmail.c...	9999967891
21	1020	kelly	tisha	London	ktb@gmail.com	119999991
22	1021	Vivi	khanna	sydney	pkb@gmail.com	1234567891
23	1022	Vanitha	Komna	Dallas	vab@gmail.com	1234567861
24	1023	Pandu	K	guntur	pkg@gmail.com	9233567891
25	1024	Geetha	lana	London	lkb@gmail.com	1234567691

Data Retrieval and Reports

There are few conditions used to work on the goals set using SQL queries, they are explained as follows:

1. To get the data of customer using Unique Customer ID.
2. To get the data of customer Email.
3. To find out all the customers in a particular locality or address.

EXPLANATION:

1. To get the data of customer using Unique Customer ID.

QUERY: `select * from customer where customerid=1000`

Result:

	CustomerId	FirstName	LastName	ContactAddress	Email	ContactNumber
1	1000	Vamsi	Krishna	Denton	vkb@gmail.com	1234567891

2. To get the data of customer Email and Contact Number.

Query: `select * from customer where Email='ten@gmail.com'`

Result:

Results Messages

	CustomerId	FirstName	LastName	ContactAddress	Email	ContactNumber
1	1001	Sachin	Tendulkar	Mumbai	ten@gmail.com	2365897412

3. To find out all the customers in a particular locality or address.

Query: `select * from customer where contactaddress='Denton'`

Results

Messages

	CustomerId	FirstName	LastName	ContactAddress	Email	ContactNumber
1	1000	Vamsi	Krishna	Denton	vkb@gmail.com	1234567891
2	1002	Hemanth	Gupta	Denton	hg@gmail.com	128558867891
3	1003	Durga	Uppala	Denton	du@gmail.com	6589745123
4	1012	sai	Krishna	Denton	skb@gmail.com	1234547391
5	1019	Manu	Nk	Denton	mnk@gmail.com	9999967891

Query and It's operations for Shopping Bag Table

Shopping Bag:

```
create table ShoppingBag(
ShoppingId int primary key identity(100,1),
VendorId int foreign key references Vendor(VendorId),
ProductId int foreign key references Product(ProductId),
CustomerId int foreign key references Customer(CustomerId),
)
```

Insert Statements:

```
insert into ShoppingBag values(500,200,1000)
insert into ShoppingBag values(501,201,1001)
insert into ShoppingBag values(502,202,1002)
insert into ShoppingBag values(503,203,1003)
insert into ShoppingBag values(504,204,1004)
```

Results:

	ShoppingId	VendorId	ProductId	CustomerId
1	100	500	200	1000
2	101	501	201	1001
3	102	502	202	1002
4	103	503	203	1003
5	104	504	204	1004

Data Retrieval and Reports

There are few conditions used to work on the goals set using SQL queries, they are explained as follows:

1. To get the data of Customer with the Shopping ID.
2. To get the Vendor details who performed that transaction at POS.
3. To Find out the category of the product purchased by the customer.

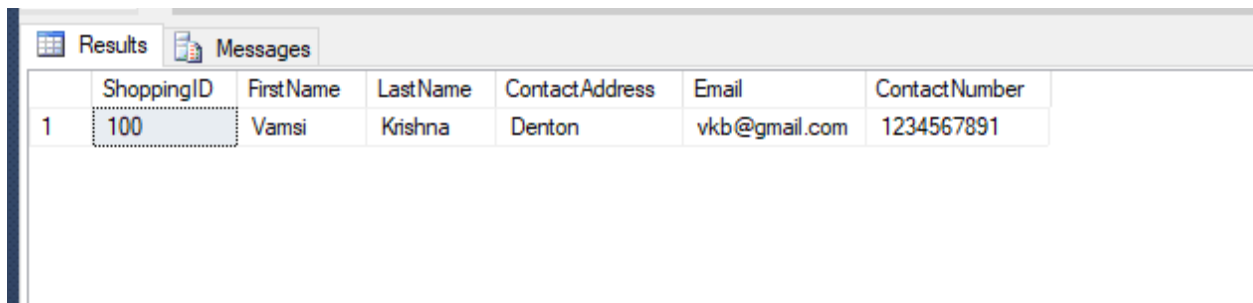
EXPLANATION:

1. To get the data of Customer with the Shopping ID.

Query: `select ShoppingID,FirstName,LastName,ContactAddress,Email,ContactNumber from customer join ShoppingBag on customer.CustomerId=ShoppingBag.CustomerId where shoppingid=100`

To get the data of customer using shoppingID, we have used join to get the data related to shopping ID. Above is the query used. We have retrieved the customer details of shopping ID=100.

Result:



The screenshot shows a SQL Server query results window with two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with 7 columns: ShoppingID, FirstName, LastName, ContactAddress, Email, and ContactNumber. The first row of data shows ShoppingID 100, FirstName Vamsi, LastName Krishna, ContactAddress Denton, Email vkb@gmail.com, and ContactNumber 1234567891. The ShoppingID cell in the first row is highlighted with a dashed border.

	ShoppingID	FirstName	LastName	ContactAddress	Email	ContactNumber
1	100	Vamsi	Krishna	Denton	vkb@gmail.com	1234567891

2. To get the Vendor details who performed that transaction at POS.

Query: `select shoppingid,vendor.UserName as VendorName,vendor.FullName as VendorFullName,vendor.Designation,vendor.AccountType from vendor join ShoppingBag on ShoppingBag.VendorId=Vendor.VendorId where shoppingid=100`

For this, we have used join between Shopping Bag and Vendor Table. In the above mentioned query, we will find the vendor information who performed shoppingId = 100.

Results:

Results		Messages			
	shoppingid	VendorName	VendorFullName	Designation	AccountType
1	100	John Woo	John Woo Reddy	Trash Head	Contract

3. To Find out the product information purchased by the customer.

Query:

Select

shoppingid,product.ProductName,Product.UnitPrice,Product.DiscountPercentage,Product.InvoiceId from product join ShoppingBag on product.ProductId=ShoppingBag.ProductId where shoppingid=100

In this, we have used Join to find out the product information purchased by customer using shipping id as reference. In the above example, we have used shoppingid =100 as reference. Below is the result for better understanding.

Result:

Results		Messages			
	shoppingid	ProductName	UnitPrice	DiscountPercentage	InvoiceId
1	100	Tolenol	25	12	10000

Query and It's operations for Vendor Table

Vendor Table:

```
create table Vendor(  
VendorId int primary key identity(500,1),  
UserName varchar(25),  
Password varchar(25),  
FullName varchar(25),  
ContactNumber bigint,  
Designation varchar(25),  
AccountType Varchar(25)  
)
```

Insert Query:

```
insert into vendor values('John Woo','Walmart@123','John Woo Reddy',5698745615,'Trash  
Head','Contract')  
insert into vendor values('Steve','Costco@123','Steve Sastri', 9658987456,'Store Manager','Full-  
Time')  
insert into vendor values('Robert','Target@123','Robert Naidu',8632548715,'Store  
Owner','Owner')  
insert into vendor values('Robinson','Coop@123','Robinson  
Chowdary',9325415615,'Supervisor','Contract')  
insert into vendor values('Alyssa','Kroger@123','Alyssa  
Gupta',7412535615,'Supplier','Contract');
```

Results:

Results		Messages					
	VendorId	UserName	Password	FullName	ContactNumber	Designation	AccountType
1	500	John Woo	Walmart@123	John Woo Reddy	5698745615	Trash Head	Contract
2	501	Steve	Costco@123	Steve Sastri	9658987456	Store Manager	Full-Time
3	502	Robert	Target@123	Robert Naidu	8632548715	Store Owner	Owner
4	503	Robinson	Coop@123	Robinson Chowdary	9325415615	Supervisor	Contract
5	504	Alyssa	Kroger@123	Alyssa Gupta	7412535615	Supplier	Contract

Data Retrieval and Reports

There are few conditions used to work on the goals set using SQL queries, they are explained as follows:

1. Total Number of Invoices generated by each Vendor on a day.
2. Total Amount Billed by each vendor on a day.

Explanation:

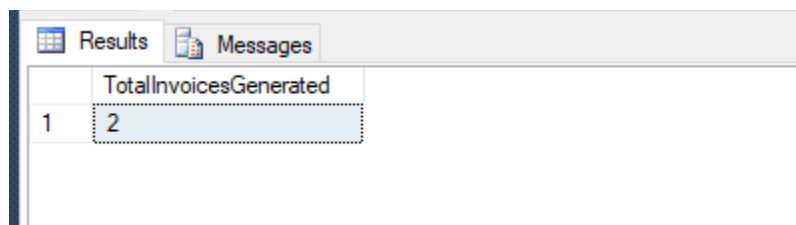
1. Total Number of Invoices generated by each Vendor on a day.

Query: `select count(InvoiceId) as TotalInvoicesGenerated from Invoice join Vendor on Invoice.VendorId=Vendor.VendorId where Vendor.VendorId=500 and DatePurchased='2021-12-07'`

In the above Example Query, we have joined the Invoice and Vendor Tables and found that Vendor who bears an ID=500 has performed 2 transactions on '2021-12-07'.

This will help the owner to understand the work done by each vendor and can assess the Required Labor as per the Performed Transactions and Business requirement.

Result:



	TotalInvoicesGenerated
1	2

2. Total Amount Billed by each vendor on a day.

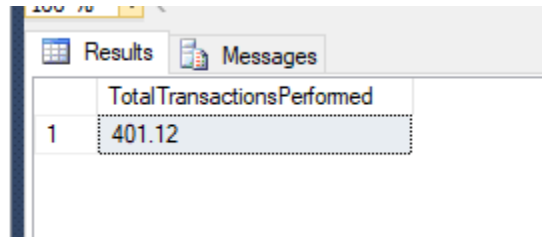
Query: `select sum(TotalAmount) as TotalTransactionsPerformed from Invoice join Vendor on Invoice.VendorId=Vendor.VendorId where Vendor.VendorId=500 and DatePurchased='2021-12-07'`

In the above Query example, we have joined the the Invoice and Vendor Tables and found that Vendor who bears an ID=500 has performed total transactions

Amount on '2021-12-07' is \$401.12.

This will help the Retail store owner to know the sales done by each vendor at POS on each day.

Result:



	TotalTransactionsPerformed
1	401.12

Query and It's operations for Product Table

Product Table:

```
create table Product(  
  ProductId int primary key identity(200,1),  
  ProductName varchar(25),  
  UnitPrice float,  
  UnitInStock int,  
  DiscountPercentage float,  
  CategoryId int foreign key references ProductCategory(CategoryId),  
  VendorId int foreign key references Vendor(VendorId),  
  SupplierId int foreign key references Supplier(SupplierId),  
  InvoiceId int foreign key references Invoice(InvoiceId)  
)
```

Insert Statements:

```
INSERT into product values('Tolenol',25.00,250,12.0,1,500,800,10000)
INSERT into product values('Biscuits',15.00,25,5.0,2,501,801,10001)
INSERT into product values('Sofaset',400.00,25.0,5.0,3,502,802,10002)
INSERT into product values('Dishes',35.00,25,10.0,4,503,803,10003)
INSERT into product values('Lipstick',18.00,250,10.0,5,504,804,10004)
INSERT into product values('TeddyBear',10.00,300,5.0,6,500,805,10005)
INSERT into product values('Speakers',40.00,100,15.0,7,501,800,10043)
INSERT into product values('Fish',5.00,25,2.0,8,502,801,10044)
INSERT into product values('Juice',10.00,250,2.0,9,503,802,10045)
INSERT into product values('Milk',5.00,350,12.0,10,504,803,10046)
INSERT into product values('Night Dress',25.00,200,12.0,11,500,804,10047)
INSERT into product values('Plates',35.00,450,2.0,12,501,805,10048)
INSERT into product values('Bedsheet',25.00,50,10.0,13,502,800,10049)
INSERT into product values('Wires',15.00,250,1.0,14,503,801,10055)
INSERT into product values('Pedigree',55.00,260,10.0,15,504,802,10056)
INSERT into product values('Salads',15.00,20,2.0,16,500,803,10057)
INSERT into product values('Support Sticks',25.00,250,2.0,17,501,804,10058)
INSERT into product values('Swim Dress',35.00,250,22.0,18,502,805,10059)
INSERT into product values('Net',5.00,250,1.0,19,503,800,10060)
INSERT into product values('Shoes',45.00,350,20.0,20,504,801,10061)
INSERT into product values('Tyre',25.00,250,2.0,21,500,802,10062)
INSERT into product values('Dipers',25.00,50,15.0,22,501,803,10063)
INSERT into product values('Face wash',15.00,20,2.0,23,502,804,10064)
INSERT into product values('seeds',25.00,250,1.0,24,503,805,10065)
INSERT into product values('Grapes',7.00,100,10.0,25,504,800,10066)
```

Results:

	ProductId	ProductName	UnitPrice	UnitInStock	DiscountPercentage	CategoryId	VendorId	SupplierId	InvoiceId
1	200	Tolenol	25	250	12	1	500	800	10000
2	201	Biscuits	15	25	5	2	501	801	10001
3	202	Biscuits	15	25	5	2	501	801	10001
4	203	Sofaset	400	25	5	3	502	802	10002
5	204	Dishes	35	25	10	4	503	803	10003
6	205	Lipstick	18	250	10	5	504	804	10004
7	206	TeddyBear	10	300	5	6	500	805	10005
8	207	Speakers	40	100	15	7	501	800	10043
9	208	Fish	5	25	2	8	502	801	10044
10	209	Juice	10	250	2	9	503	802	10045
11	210	Milk	5	350	12	10	504	803	10046
12	211	Night Dress	25	200	12	11	500	804	10047
13	212	Plates	35	450	2	12	501	805	10048
14	213	Bedsheet	25	50	10	13	502	800	10049
15	215	Pedigree	55	260	10	15	504	802	10056
16	216	Salads	15	20	2	16	500	803	10057
17	217	Support Stic...	25	250	2	17	501	804	10058
18	218	Swim Dress	35	250	22	18	502	805	10059
19	219	Net	5	250	1	19	503	800	10060
20	220	Shoes	45	350	20	20	504	801	10061
21	221	Tyre	25	250	2	21	500	802	10062
22	222	Dipers	25	50	15	22	501	803	10063
23	223	Face wash	15	20	2	23	502	804	10064
24	224	seeds	25	250	1	24	503	805	10065
25	225	Grapes	7	100	10	25	504	800	10066

Data Retrieval and Reports

There are few conditions used to work on the goals set using SOL queries, they are explained as follows:

1. On which product, we are offering highest Discount to Customers?
2. Which Product is Costliest in our Retail Store?

Explanation:

1. On which product, we are offering highest Discount to Customers?

Query: `select * from product where DiscountPercentage=(select max(DiscountPercentage) from Product);`

To find out the Max offered discount row, we have used sub query concept to retrieve the data.

Result:

Results		Messages							
	ProductId	ProductName	UnitPrice	UnitInStock	DiscountPercentage	CategoryId	VendorId	SupplierId	InvoiceId
1	218	Swim Dress	35	250	22	18	502	805	10059

2. Which Product is Costliest in our Retail Store?

Query: `select * from product where UnitPrice=(select max(UnitPrice) from Product);`

To find out the Max Unit Price or Costliest Product in a store, we have used sub query concept to retrieve the data.

Result:

Results		Messages							
	ProductId	ProductName	UnitPrice	UnitInStock	DiscountPercentage	CategoryId	VendorId	SupplierId	InvoiceId
1	203	Sofaset	400	25	5	3	502	802	10002

Query and It's operations for Product Category Table

Product Category Table:

`create table ProductCategory(`

```
CategoryId int primary key identity(1,1),  
CategoryName varchar(30)  
)
```

Insert Statements:

```
insert into ProductCategory values('Pharmceuticals')  
insert into ProductCategory values('Food')  
insert into ProductCategory values('Furniture')  
insert into ProductCategory values('KitchenWare')  
insert into ProductCategory values('Cosmetics')  
insert into ProductCategory values('Toys')  
insert into ProductCategory values('Electronics')  
insert into ProductCategory values('Meat')  
insert into ProductCategory values('Beverages')  
insert into ProductCategory values('Dairy')  
insert into ProductCategory values('Clothing')  
insert into ProductCategory values('Disposables')  
insert into ProductCategory values('Travel')  
insert into ProductCategory values('Hardware')  
insert into ProductCategory values('Animal Food')  
insert into ProductCategory values('Perishable Goods')  
insert into ProductCategory values('Hiking')  
  
insert into ProductCategory values('Swim Wear')  
  
insert into ProductCategory values('Fishing')  
  
insert into ProductCategory values('Trekking')  
insert into ProductCategory values('Automobile')  
insert into ProductCategory values('Babycare')  
insert into ProductCategory values('BathProducts')  
insert into ProductCategory values('Grains')  
insert into ProductCategory values('Fruits')
```

Results:

	ProductId	ProductName	UnitPrice	UnitInStock	DiscountPercentage	CategoryId	VendorId	SupplierId	InvoiceId
1	200	Tolenol	25	250	12	1	500	800	10000
2	201	Biscuits	15	25	5	2	501	801	10001
3	202	Biscuits	15	25	5	2	501	801	10001
4	203	Sofaset	400	25	5	3	502	802	10002
5	204	Dishes	35	25	10	4	503	803	10003
6	205	Lipstick	18	250	10	5	504	804	10004
7	206	TeddyBear	10	300	5	6	500	805	10005
8	207	Speakers	40	100	15	7	501	800	10043
9	208	Fish	5	25	2	8	502	801	10044
10	209	Juice	10	250	2	9	503	802	10045
11	210	Milk	5	350	12	10	504	803	10046
12	211	Night Dress	25	200	12	11	500	804	10047
13	212	Plates	35	450	2	12	501	805	10048
14	213	Bedsheet	25	50	10	13	502	800	10049
15	215	Pedigree	55	260	10	15	504	802	10056
16	216	Salads	15	20	2	16	500	803	10057
17	217	Support Stic...	25	250	2	17	501	804	10058
18	218	Swim Dress	35	250	22	18	502	805	10059
19	219	Net	5	250	1	19	503	800	10060
20	220	Shoes	45	350	20	20	504	801	10061
21	221	Tyre	25	250	2	21	500	802	10062
22	222	Dipers	25	50	15	22	501	803	10063
23	223	Face wash	15	20	2	23	502	804	10064
24	224	seeds	25	250	1	24	503	805	10065
25	225	Grapes	7	100	10	25	504	800	10066

Data Retrieval and Reports

There are few conditions used to work on the goals set using SQL queries, they are explained as follows:

- 1.** To find out the category of each registered product in our retail store database.
- 2.** To find out the Largest amount of sales on which category of products as per the different seasons.

Explanation:

1. To find out the category of each registered product in our retail store database.

Query: select Product.ProductName,ProductCategory.CategoryName from product join ProductCategory on product.CategoryId=ProductCategory.CategoryId

To find out each product and its category, we have joined product and product category tables. Below is the result.

Result:

	ProductName	CategoryName
1	Tolenol	Pharmceuticals
2	Biscuits	Food
3	Biscuits	Food
4	Sofaset	Furniture
5	Dishes	KitchenWare
6	Lipstick	Cosmetics
7	TeddyBear	Toys
8	Speakers	Electronics
9	Fish	Meat
10	Juice	Beverages
11	Milk	Dairy
12	Night Dress	Clothing
13	Plates	Disposables
14	Bedsheet	Travel
15	Pedigree	Animal Food
16	Salads	Perishable G...
17	Support Stic...	Hiking
18	Swim Dress	Swim Wear
19	Net	Fishing
20	Shoes	Trekking
21	Tyre	Automobile
22	Dipers	Babycare
23	Face wash	BathProducts
24	seeds	Grains
25	Grapes	Fruits

2. To find out the Largest amount of sales on which category of products as per the different seasons.

This can be explained theoretically as this business requirements get changed dynamically. For example, during the time of Christmas, there will be a high business on cakes. This rise in business can be seen only during December last week and for the New year. This Question

can answer the sales performed by the retail store for each occasion on particular product category.

As we haven't inserted sufficient amount of records to explain this with the date and seasons, we would like to explain the business concept with the theory.

Query and It's operations for Supplier Table

Supplier Table:

```
create table Supplier(  
SupplierId int primary key identity(800,1),  
ContactNumber bigint,  
SupplierEmail varchar(30),  
SupplierAddress varchar(50),  
VendorId int foreign key references Vendor(VendorId)  
)
```

Insert Statements:

```
insert into supplier values(865231478,'Kasia@amcor.com','716 W Ant St',500)  
insert into supplier values(569874512,'Anna@Mccormick.com','856 E praire St paris',502)  
insert into supplier values(63214789,'Angela@Philips.com','9220 E lincoln St',501)  
insert into supplier values(54712885,'Emily@apple.com','4325 S Oak St',503)  
insert into supplier values(96634525,'Jenn@samsung.com','143 washington st',504)
```

Results:

Results		Messages			
	SupplierId	ContactNumber	SupplierEmail	SupplierAddress	VendorId
1	800	865231478	Kasia@amcor.com	716 W Ant St	500
2	801	8536767211	Maisa@gmail.com	717 W Ant st	503
3	802	569874512	Anna@Mccormick.com	856 E praire St paris	502
4	803	63214789	Angela@Philips.com	9220 E lincoln St	501
5	804	54712885	Emily@apple.com	4325 S Oak St	503
6	805	96634525	Jenn@samsung.com	143 washington st	504

Data Retrieval and Reports

There are few conditions used to work on the goals set using SQL queries, they are explained as follows:

1. To get the information of supplier for each product in a store.
2. To maintain updated inventory in a store, this Query will help to maintain sufficient unit stock in a retail store

Explanation:

1. To get the information of supplier for each product in a store.

Query: select

productname,productid,supplier.SupplierId,supplier.SupplierEmail,supplier.ContactNumber
from product join supplier on Product.SupplierId=supplier.SupplierId

Result:

	productname	productid	SupplierId	SupplierEmail	ContactNumber
1	Tolenol	200	800	Kasia@amcor.com	865231478
2	Biscuits	201	801	Maisa@gmail.com	8536767211
3	Biscuits	202	801	Maisa@gmail.com	8536767211
4	Sofaset	203	802	Anna@Mccormick.com	569874512
5	Dishes	204	803	Angela@Philips.com	63214789
6	Lipstick	205	804	Emily@apple.com	54712885
7	TeddyBear	206	805	Jenn@samsung.com	96634525
8	Speakers	207	800	Kasia@amcor.com	865231478
9	Fish	208	801	Maisa@gmail.com	8536767211
10	Juice	209	802	Anna@Mccormick.com	569874512
11	Milk	210	803	Angela@Philips.com	63214789
12	Night Dress	211	804	Emily@apple.com	54712885
13	Plates	212	805	Jenn@samsung.com	96634525
14	Bedsheet	213	800	Kasia@amcor.com	865231478
15	Pedigree	215	802	Anna@Mccormick.com	569874512
16	Salads	216	803	Angela@Philips.com	63214789
17	Support Sti...	217	804	Emily@apple.com	54712885
18	Swim Dress	218	805	Jenn@samsung.com	96634525
19	Net	219	800	Kasia@amcor.com	865231478
20	Shoes	220	801	Maisa@gmail.com	8536767211
21	Tyre	221	802	Anna@Mccormick.com	569874512
22	Dipers	222	803	Angela@Philips.com	63214789
23	Face wash	223	804	Emily@apple.com	54712885
24	seeds	224	805	Jenn@samsung.com	96634525
25	Granes	225	800	Kasia@amcor.com	865231478

2. To maintain updated inventory in a store, this business will help to maintain sufficient unit stock in a retail store

As Transactions are performed at store by different vendors at POS for different customers, Unit stock will be subtracted as per the sales number of products for each quantity. As soon as number gets reduced, store owner will need to create a new order for the supplier in order to meet the business demand. As the considered diagram is limited, we couldn't show the query and results.

Query and It's operations for Invoice Table

Invoice Table:

```
create table Invoice(  
InvoiceId int primary key identity(10000,1),  
CustomerId int foreign key references Customer(CustomerId),  
TotalAmount float,  
PaymentType Varchar(25),  
CardNumber bigint,  
CardHolderName varchar(50),  
DatePurchased Date,  
VendorId int foreign Key references Vendor(VendorId),  
)
```

Insert Statements:

```
insert into invoice values(1000,200.56,'Amex Card',1234567891234999,'Vamsi','2021-12-07',500)
```



```

insert into invoice values(1001,450.56,'Discover Card',4292567823547865,'Hemanth','2022-10-09',501)
insert into invoice values(1002,550.64,'Freedom Card',9361346587650934,'Durga','2023-11-10',502)
insert into invoice values(1003,760.92,'Apple Card',5431986579864356,'Ravi Teja','2024-11-05',503)
insert into invoice values(1004,250.56,'Credit One',6571478635879743,'Bala Krishna','2028-06-06',504)
insert into invoice values(1005,670.56,'Amex Card',1964891234999,'Jai Ballaya','2024-08-03',501)
insert into invoice values(1006,230.00,'Sofi Card',7891435267458798,'Jr NTR','2029-10-03',502)
insert into invoice values(1007,270.56,'Discover Card',1234567899876543,'Sr NTR','2030-11-04',503)
insert into invoice values(1008,290.56,'Amex Card',6571564523541234,'Chiranjeevi','2028-01-05',504)
insert into invoice values(1009,310.96,'Apple Card',3456777891234999,'Kini Konna','2021-12-07',503)
insert into invoice values(1010,410.56,'Discover Card',6574568789554672,'Sai Krishna','2021-12-15',503)
insert into invoice values(1011,450.56,'Freedom Card',5565778922334455,'Pranathi Nandu','2021-12-07',504)
insert into invoice values(1012,268.76,'Discover Card',9988778800776655,'Ruthu Priya','2021-12-12',504)
insert into invoice values(1013,244.56,'Apple Card',7787669833542354,'Mahesh Babu','2021-12-07',504)
insert into invoice values(1014,566.78,'Amex Card',3354666744528898,'Pawan Kalyan','2030-02-02',504)
insert into invoice values(1015,212.45,'Amex Card',1234567891234999,'Prabhas','2021-05-28',504)
insert into invoice values(1016,313.46,'Discover Card',5634567891234999,'Nani','2033-11-11',504)
insert into invoice values(1017,222.56,'Amex Card',9856767891234999,'Manu','2022-12-07',504)
insert into invoice values(1018,234.76,'Apple Card',5578097533541234,'Kelly','2021-12-07',504)
insert into invoice values(1019,200.56,'Amex Card',3345667899861234,'Vivi','2021-12-07',504)
insert into invoice values(1020,540.56,'Discover Card',8965345487536874,'Vanitha','2021-12-07',504)
insert into invoice values(1021,640.56,'Amex Card',12345678912341254,'Pandu','2021-12-07',504)
insert into invoice values(1022,755.56,'Amex Card',123456789123568,'Geetha','2021-12-07',504)
insert into invoice values(1023,655.56,'Gift Card',123456789123568,'Ramya','2021-12-07',501)

```

Result:

Results		Messages							
	InvoiceId	CustomerId	TotalAmount	Payment Type	CardNumber	CardHolderName	DatePurchased	VendorId	
1	10000	1000	200.56	Amex Card	1234567891234999	Vamsi	2021-12-07	500	
2	10001	1000	200.56	Amex Card	1234567891234999	Vamsi	2021-12-07	500	
3	10002	1001	450.56	Discover Card	4292567823547865	Hemanth	2022-10-09	501	
4	10003	1002	550.64	Freedom Card	9361346587650934	Durga	2023-11-10	502	
5	10004	1003	760.92	Apple Card	5431986579864356	Ravi Teja	2024-11-05	503	
6	10005	1004	250.56	Credit One	6571478635879743	Bala Krishna	2028-06-06	504	
7	10043	1005	670.56	Amex Card	1964891234999	Jai Ballaya	2024-08-03	501	
8	10044	1006	230	Sofi Card	7891435267458798	Jr NTR	2029-10-03	502	
9	10045	1007	270.56	Discover Card	1234567899876543	Sr NTR	2030-11-04	503	
10	10046	1008	290.56	Amex Card	6571564523541234	Chiranjeevi	2028-01-05	504	
11	10047	1009	310.96	Apple Card	3456777891234999	Kini Konna	2021-12-07	503	
12	10048	1010	410.56	Discover Card	6574568789554672	Sai Krishna	2021-12-15	503	
13	10049	1011	450.56	Freedom Card	5565778922334455	Pranathi Nandu	2021-12-07	504	
14	10055	1017	222.56	Amex Card	9856767891234999	Manu	2022-12-07	504	
15	10056	1018	234.76	Apple Card	5578097533541234	Kelly	2021-12-07	504	
16	10057	1019	200.56	Amex Card	3345667899861234	Vivi	2021-12-07	504	
17	10058	1020	540.56	Discover Card	8965345487536874	Vanitha	2021-12-07	504	
18	10059	1021	640.56	Amex Card	12345678912341...	Pandu	2021-12-07	504	
19	10060	1022	755.56	Amex Card	123456789123568	Geetha	2021-12-07	504	
20	10061	1012	268.76	Discover Card	9988778800776655	Ruthu Priya	2021-12-12	504	
21	10062	1013	244.56	Apple Card	7787669833542354	Mahesh Babu	2021-12-07	504	
22	10063	1014	566.78	Amex Card	3354666744528898	Pawan Kalyan	2030-02-02	504	
23	10064	1015	212.45	Amex Card	1234567891234999	Prabhas	2021-05-28	504	
24	10065	1016	313.46	Discover Card	5634567891234999	Nani	2033-11-11	504	
25	10066	1023	655.56	Gift Card	123456789123568	Ramya	2021-12-07	501	

Data Retrieval and Reports

There are few conditions used to work on the goals set using SQL queries, they are explained as follows:

1. How many Invoices generated by the same customer in a month. This will help a store to manage whether customer is loyal?
2. Total Amount that store received in a form of cash on every day?

Explanation:

1. How many Invoices generated by the same customer in a month. This will help a store to manage whether customer is loyal?

Query:

```
select count(invoiceid) as NumberOfInvoices from invoice where CustomerId=1000 and Month(DatePurchased)=12
```

Result:

Results Messages	
	NumberOfInvoices
1	2

This can explain to store managers that how often the customer is visiting the store and performing the transactions. From this, stores can also offer special discounts to loyal customers by offering higher discounts.

2. Total Amount that store received in a form of card on every month?

Query:

```
select sum(TotalAmount) as TotalEarned from Invoice where paymenttype like '%card'and
Month(datepurchased)=12
```

Result:

Results Messages	
	TotalEarned
1	5336.64

Conclusion :

From this project, we can conclude that this database can be used in Retail stores to maintain their Transactions, Instock Quantities, Vendors, products and Suppliers data Robustly. For the Future work, this project can be extended by implementing Artificial Intelligence for maintaining inventories and to Update the Database transactions with the scheduled jobs. Apart from this manipulation techniques, we can deploy all the data to Cloud Computing for high reliability, scalability and Robustness of Data.

As Supply chain Industry and Retail Industries are ever green, these types of projects will be helpful in development and optimization of databases. This project will help Retail stores to perform their businesses efficiently and effectively.