



Topedia Online Forum¹

Database Design

Project Implementation Report



²

Table of Contents

Project Summary	3
User Introduction	3
Business Rules:	3
Entity and Attribute Tables	4
AddressEntity	4
Access_Type	4
UserEntity	4
Courier	5
Employee	5
Post_Category	5
Status_Type.....	6
Content_Post	6
Post_Reply	7
Product_Category.....	7
Product	7
Vendor.....	8
Product_Supply	8
User_Own_Coupon.....	9
Payment_Method_Code.....	9
Payment_Method	10
Purchase	10
Purchase_Line.....	11
Delivery_Status.....	11
Shipping_Status	11
Software Infrastructure.....	13
SQL Queries for creating tables and inserting data	13
CREATING TABLES	13
INSERTING TABLE.....	19
OUTPUT DISPLAY	29
Major Data Questions.....	38
Interface Display	47
Login Page	47
Sign Up Page.....	48
User Information Management	49
Create Post	50
Post Reply	51
Shopping Cart	53
Coupon Management	55
Purchase History	56
User Information Management	57
Purchasing List.....	59
Content Post Management	59
Product Replenishment Management	60
Key Performance Indicator	61
Package Shipping Management	62
Benefit Management.....	65

Project Summary

The project aims to design a database system for an online forum called ToyPedia. Even though several functionalities, like posting and replying, already equipped by the original database system, the system need an update due to issues of budget overrun and the burden brought by increasing number of users. To satiate the requirements, a new database has been proposed and then developed. This updated forum allows its users to post content that is relevant with toy, to review the content that posts by another user, and to buy product that is available in the forum shopping zone. Specifically, new functionality like purchasing has been introduced which is being expected to bring extra profit to the forum management group.

User Introduction

In this report, users can be divided into two groups, which are user or consumer group and administration group. Each of group is a composite of other different user members.

1. User Group
 - Normal User who has limited privileges
 - VIP User who enjoy all the available benefits
2. Administration Group
 - Employee who has the lowest access right
 - Sales Manager who can manipulate selling price of product and implement marketing strategy by distributing coupons to different targeted users.
 - Administrator is the employee who has the highest security right to access the data stored in the database.

Business Rules:

The following are the business rules that were followed while designing this database

- a) Each employee has a unique discount rate when they purchase an item.
- b) The VIP User will not be charged of delivery fee.
- c) A product can be supplied by many vendors and, meanwhile, a vendor can supply many products.
- d) A user must provide his/her credit/debit card details in order to purchase a product
- e) The delivery fee and tax will alter according to the address each user has.

Entity and Attribute Tables

AddressEntity

Stores the address of the user, employee, courier, and vendor.

AddressEntity	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	addressID	varchar(5)	NOT NULL	Unique identifier for this address
	streetNo	varchar(10)	NOT NULL	Street Number of the address
	streetName	varchar(50)	NOT NULL	Street Name of the address
	contactPhone	varchar(15)	NOT NULL	Phone number
	city	varchar(20)	NOT NULL	City in the address
	stateName	varchar(20)	NOT NULL	State in the address
	country	varchar(50)	NOT NULL	Country in the address
	zipcode	varchar(10)	NOT NULL	Zipcode in the address

Access_Type

Stores different access type of the user and employees as well as the corresponding explanation.

Access_Type	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	accessType	varchar(2)	NOT NULL	Numeric number which represents type of user or employee
	typeName	varchar(25)	NOT NULL	Name of the type of user or employee
	at_description	varchar(100)		Description of access type

UserEntity

Store the detailed information of each user.

UserEntity	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	userID	varchar(5)	NOT NULL	Unique identifier of this table
Foreign Key	addressID	varchar(5)	NOT NULL	Identifier for the address of the user
	fName	varchar(15)	NOT NULL	First name of the user
	mName	varchar(10)		Middle name of the user
	lName	varchar(15)	NOT NULL	Last name of the user
	nName	varchar(20)	NOT NULL	Nick name of the user
	email	varchar(30)	NOT NULL	Email id of the user
	hashedPassword	varchar(20)	NOT NULL	Password of the user
	accountCreateDate	datetime	NOT NULL	Date when the account was created
	profilePicture	varchar(60)		Profile picture of the user

Foreign Key	accessType	varchar(2)	NOT NULL	Access type of the user
-------------	------------	------------	----------	-------------------------

Courier

Stores the detailed information of each courier.

Courier	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	courierID	varchar(5)	NOT NULL	Unique identifier of the courier
Foreign Key	addressID	varchar(5)	NOT NULL	Identifier for the address of the courier
	companyName	varchar(20)	NOT NULL	Name of the courier company
	officialWebsite	varchar(100)	NOT NULL	Website of the courier company
	contactPhone	varchar(15)	NOT NULL	Phone number of the courier company

Employee

Stores the detailed information of each employee.

Employee	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	empID	varchar(5)	NOT NULL	Unique identifier of the employee
Foreign Key	addressID	varchar(5)	NOT NULL	Identifier for the address of the employee
	E_FName	varchar(15)	NOT NULL	First name of the employee
	E_MName	varchar(10)		Middle name of the employee
	E_LName	varchar(15)	NOT NULL	Last name of the employee
	E_email	varchar(20)	NOT NULL	Email id of the employee
	E_hashedPassword	varchar(20)	NOT NULL	Password of the employee
	E_accountCreateDate	datetime	NOT NULL	Date when the account was created
	E_profilePicture	varchar(60)		Profile picture of the employee
Foreign Key	accessType	varchar(2)	NOT NULL	Access type of the employee

Post_Category

Stores the elements that are relevant of posting a content.

Post_Category	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	categoryID	varchar(2)	NOT NULL	Unique identifier for the post category
	categoryName	varchar(25)	NOT NULL	Name of the post category
	categoryDescription	varchar(150)		Description of the post category

Status_Type

Stores the type of the status of the posts and reply.

Status_Type	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	statusID	varchar(2)	NOT NULL	Unique identifier for the type of status.
	statusDescription	varchar(50)		Description of the status type

Content_Post

Stores the elements that are relevant of replying a post.

Content_Post	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	postID	int	NOT NULL	Unique identifier of the post
Foreign Key	userID	varchar(5)	NOT NULL	Identifier of the user who made the post
Foreign Key	categoryID	varchar(2)	NOT NULL	Identifier of the post category of the post
	postTitle	varchar(50)	NOT NULL	Title of the post
	dateOfPost	datetime	NOT NULL	Date when the post was posted
	postContent	varchar(300)	NOT NULL	The content of the post
	picture1	varchar(50)		Stores the directory of this picture
	picture2	varchar(50)		Stores the directory of this picture
	numberOfView	integer	NOT NULL	Number of views for the post
	totalRatingScore	integer	NOT NULL	The total rating score is
	ratingUserNumber	integer	NOT NULL	The total number of users who have give rating point to the post.

	likes	integer	NOT NULL	Number of likes for the post
	dislikes	integer	NOT NULL	Number of dislikes for the post
Foreign Key	statusID	varchar(2)	NOT NULL	Identifier of the type of status of the post

Post_Reply

Stores the detailed information of replies for all the posts

Post_Reply	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	replyID	int	NOT NULL	Unique identifier of the reply.
Foreign Key	userID	varchar(5)	NOT NULL	Identifier of the user who made the reply
Foreign Key	postID	int	NOT NULL	Identifier of the post which got a reply
Foreign Key	statusID	varchar(2)	NOT NULL	Identifier of the status of the post
	dateOfReply	datetime	NOT NULL	Date when the reply was made
	content	varchar(120)	NOT NULL	Content of the reply
	pr_likes	integer	NOT NULL	Number of likes for the reply
	pr_dislikes	integer	NOT NULL	Number of dislikes for the reply

Product_Category

Stores the details of the type of products available in the forum.

Product_Category	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	productCategoryID	varchar(2)	NOT NULL	Unique identifier of the type of product
	productCategory	varchar(20)	NOT NULL	Name of the type of product
	pc_description	varchar(50)		

Product

Stores the details of the Product

Product	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	productID	varchar(5)	NOT NULL	Unique identifier of the product
Foreign Key	productCategoryID	varchar(2)	NOT NULL	Identifier of the type of product
	inventory	integer	NOT NULL	Quantity of the Stock remaining

	productName	varchar(50)	NOT NULL	Name of the product
	purchasePrice	numeric(8,2)	NOT NULL	Purchase price of the product
	salePrice	numeric(8,2)	NOT NULL	Sale price of the product
	productWeight	varchar(20)	NOT NULL	Weight of the product
	size	varchar(20)	NOT NULL	Size of the product in terms of height, breadth and width
	color	varchar(16)	NOT NULL	Primary color of the product
	title	varchar(50)	NOT NULL	Title of the product
	subtitle	varchar(80)	NOT NULL	Subtitle of the product
	p_description	varchar(150)	NOT NULL	Description of the product
	image1	varchar(60)		Image 1
	image2	varchar(60)		Image 2
	keyWord1	varchar(30)	NOT NULL	Keyword 1 of the product
	keyWord2	varchar(30)	NOT NULL	Keyword 2 of the product

Vendor

Stores the details of the vendors.

Vendor	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	vendorID	varchar(5)	NOT NULL	Identifier of this table
Foreign Key	addressID	varchar(5)	NOT NULL	The address that this vendor locates at
	vendorName	varchar(30)	NOT NULL	The name of the vendor
	officialWebsite	varchar(100)	NOT NULL	The official website of this vendor
	liasionContactPhone	varchar(15)	NOT NULL	The contact phone number of vendor's liaison
	V_accountCreateDate	datetime	NOT NULL	The date on which generates a vendor record

3

Product_Supply

Stores the details about the supply of the products from the vendor.

Product_Supply	Attribute Name	Field Type	IF NULL?	Explanation
----------------	----------------	------------	----------	-------------

Primary Key & Foreign Key	vendorID	varchar(5)	NOT NULL	Identifier of the vendor
Primary Key & Foreign Key	productID	varchar(5)	NOT NULL	identifier of the product
	replenish_quantity	integer	NOT NULL	The quantity of products supplied
Primary Key & Foreign Key	supplyStartDate	datetime	NOT NULL	The date when the supply started
	supplyEndDate	datetime		The date when the supply ended

Coupon

Stores the details of various coupons offered by the forum.

Coupon	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	couponID	varchar(5)	NOT NULL	Unique identifier of coupon
	couponName	varchar(20)	NOT NULL	Name of the coupon
	coupon_description	varchar(50)	NOT NULL	Description of the coupon
	discount_rate	numeric(5,2)	NOT NULL	Rate of discount applied to the coupon
	activeStartDate	datetime	NOT NULL	Active start date of the coupon
	activeEndDate	datetime	NOT NULL	Expiry date of coupon

User_Own_Coupon

Stores the details of the coupons that the users have.

User_Own_Coupon	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key & Foreign Key	couponID	varchar(5)	NOT NULL	Identifier of the coupon
Primary Key & Foreign Key	userID	varchar(5)	NOT NULL	Identifier of the user
	quantity	integer	NOT NULL	The number of coupons the user has
	ownDate	datetime	NOT NULL	The date when the user got it
	couponAvailable	varchar(3)	NOT NULL	Check if the coupon is available

Payment_Method_Code

Stores the type of payment methods (Credit Card or Debit Card)

Payment_Method_Code	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	paymentCodeID	varchar(2)	NOT NULL	Unique identifier of the type of payment

	codeName	varchar(20)	NOT NULL	Type of payment (i.e. Credit or Debit)
	code_description	varchar(50)		

Payment_Method

Stores the details of payment method used by the users.

Payment_Method	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	paymentMethodID	varchar(5)	NOT NULL	Unique identifier of the payment method
Foreign Key	userID	varchar(5)	NOT NULL	Identifier of the user
Foreign Key	paymentCodeID	varchar(2)	NOT NULL	Identifier of the type of payment
	cardNumber	varchar(16)	NOT NULL	Card number of the credit/debit Card
	expiryDate	varchar(5)	NOT NULL	Expiry date of the credit/debit card
	cardKeeper	varchar(50)	NOT NULL	Name of the card holder
	cvvNumber	varchar(3)	NOT NULL	CVV number of the card

Purchase

Stores the details of the purchase made by the users.

Purchase	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	purchasID	varchar(5)	NOT NULL	Unique identifier of the purchase
Foreign Key	payementMethodID	varchar(2)	NOT NULL	Identifier of the type of payment method
Foreign Key	userID	varchar(5)	NOT NULL	
Foreign Key	couponID	varchar(5)	NOT NULL	Coupon that user might use in the purchase order
	taxRate	numeric(5,2)	NOT NULL	
	deliveryFee	numeric(6,2)	NOT NULL	
	dateOfPurchase	datetime	NOT NULL	

Purchase_Line

Stores the list of products in a purchase.

Purchase_Line	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key & Foreign Key	purchaseID	varchar(5)	NOT NULL	Identifier of the purchase
Primary Key & Foreign Key	productID	varchar(5)	NOT NULL	Identifier of the product
	pl_quantity	integer	NOT NULL	Purchase quantity

Delivery_Status

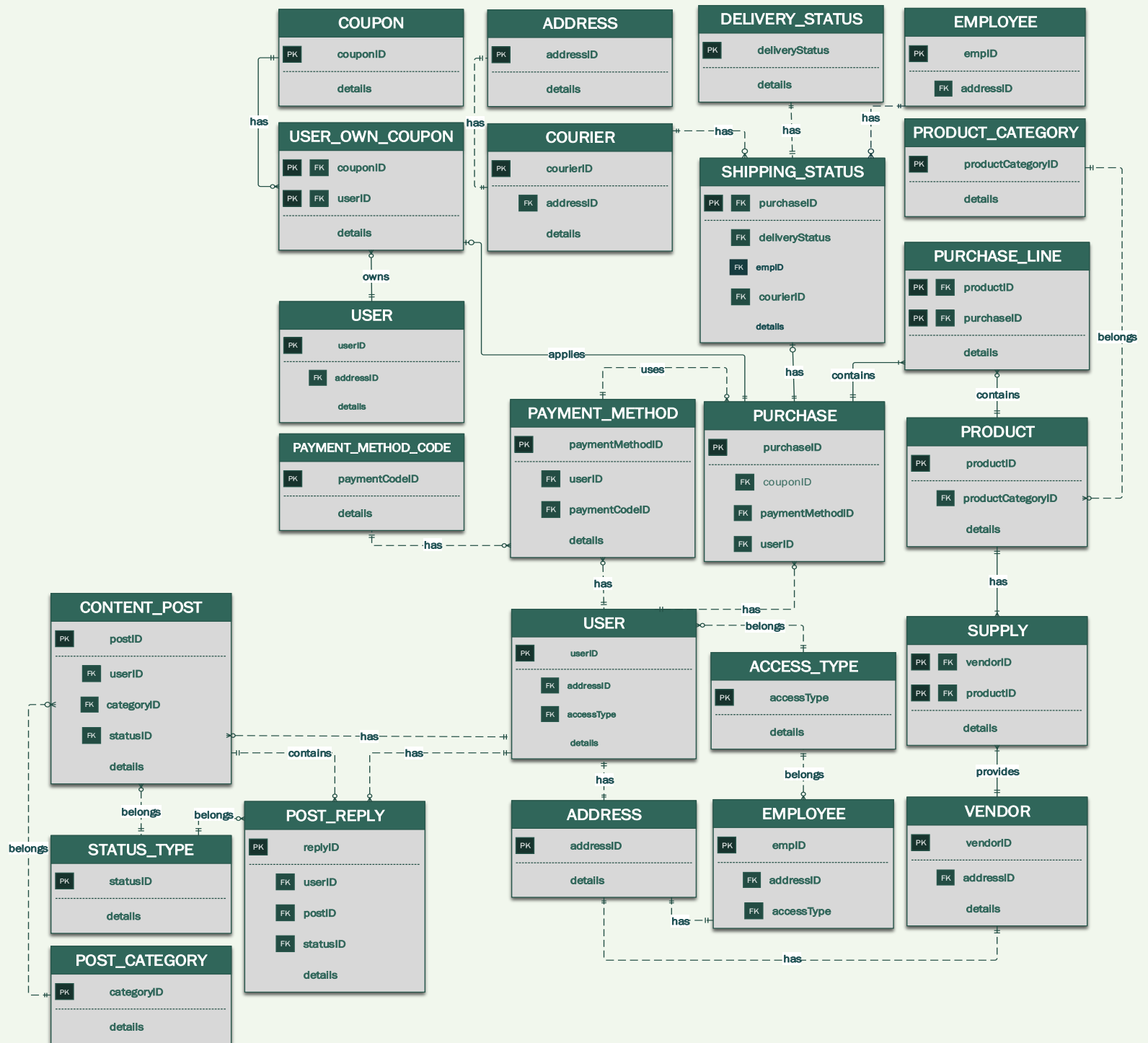
Stores the delivery status of the order made by the user.

Delivery_Status	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key	deliveryStatus	varchar(2)	NOT NULL	Delivery status of the order
	ds_description	varchar(50)	NOT NULL	Description

Shipping_Status

Stores the shipping status of the order

Shipping_Status	Attribute Name	Field Type	IF NULL?	Explanation
Primary Key & Foreign Key	purchaseID	varchar(5)	NOT NULL	Identifier of the purchase
Foreign Key	deliveryStatus	varchar(2)	NOT NULL	Identifier of the delivery status
Foreign Key	empID	varchar(5)	NOT NULL	Identifier of the employee
Foreign Key	courierID	varchar(5)	NOT NULL	identifier of the courier
	startDateofShipping	datetime	NOT NULL	Start date of shipping
	endDateofShipping	datetime		End date of shipping



Software Infrastructure

For the following database design, we have used the following software tools

MS Visio

We have used MS Visio to create the Entity Relationship Diagram (ERD), establishing relationships between the entities to understand the flow of data in the database.

MS SQL

Through MS SQL, we have created the various tables of our database, inserted sample values into these tables and created a trigger that helps making our database more efficient.

MS Access

Once the tables have been created in MS SQL, we can import the data from SQL into MS Access and after this, we have created the various views, forms and reports required to answer the major data questions of our project. Using forms and reports helps to display the data from our database in a much more readable format.

6

SQL Queries for creating tables and inserting data

CREATING TABLES

7

AddressEntity

```
create table AddressEntity (  
    addressID varchar(5) not null,  
    streetNo  varchar(10) not null,  
    streetName varchar(50) not null,  
    contactPhone varchar(15) not null,  
    city varchar(20) not null,  
    stateName varchar(20) not null,  
    country varchar(50) not null,  
    zipcode varchar(10) not null,  
  
    constraint address_pk primary key (addressID)  
);
```

Access_Type

```
create table Access_Type (  
    accessType varchar(2) not null,  
    typeName varchar(25) not null,  
    at_description varchar(100),  
  
    constraint access_type_pk primary key (accessType)  
);
```

User_Entity

```
create table UserEntity (  
    userID varchar(5) not null,  
    addressID varchar(5) not null,  
    fName varchar(15) not null,  
    mName varchar(10),  
    lName varchar(15) not null,  
    nName varchar(20) not null,  
    email varchar(30) not null,  
    hashedPassword varchar(20) not null,  
    accountCreateDate datetime not null default getdate(),  
    profilePicture varchar(60),  
    accessType varchar(2) not null default '2',  
  
    constraint user_pk primary key (userID),  
    constraint user_fk1 foreign key (addressID) references AddressEntity(addressID),  
    constraint user_fk2 foreign key (accessType) references Access_Type(accessType)  
);
```

Courier

```
create table Courier (  
    courierID varchar(5) not null,  
    addressID varchar(5) not null,  
    companyName varchar(20) not null,  
    officialWebsite varchar(100) not null,  
    contactPhone varchar(15) not null,  
  
    constraint courier_pk primary key (courierID),  
    constraint courier_fk1 foreign key (addressID) references AddressEntity(addressID)  
);
```

Employee

```
create table Employee (  
    empID varchar(5) not null,  
    addressID varchar(5) not null,  
    E_FName varchar(15) not null,  
    E_MName varchar(10),  
    E_LName varchar(15) not null,  
    E_email varchar(20) not null,  
    E_hashedPassword varchar(20) not null,  
    E_accountCreateDate datetime not null default getdate(),  
    E_profilePicture varchar(60),  
    accessType varchar(2) not null,  
  
    constraint employee_pk primary key (empID),  
    constraint employee_fk1 foreign key (addressID) references  
AddressEntity(addressID),  
    constraint employee_fk2 foreign key (accessType) references  
Access_Type(accessType)  
);
```

Post_Category

```
create table Post_Category (  
    categoryID varchar(2) not null,  
    categoryName varchar(25) not null,  
    categoryDescription varchar(150),  
  
    constraint post_category_pk primary key (categoryID)  
);
```

Status_Type

```
create table Status_Type (  
    statusID varchar(2) not null,  
    statusDescription varchar(50) not null,  
  
    constraint status_type_pk primary key (statusID)  
);
```

Content_Post

```
create table Content_Post (  
    postID int not null,  
    userID varchar(5) not null,  
    categoryID varchar(2) not null,  
    postTitle varchar(50) not null,  
    dateOfPost datetime not null default getdate(),  
    postContent varchar(300) not null,  
    picture1 varchar(50),  
    picture2 varchar(50),  
    numberOfView integer not null default 0,  
    totalRatingScore integer not null default 0,  
    ratingUserNumber integer not null default 0,  
    likes integer not null default 0,  
    dislikes integer not null default 0,  
    statusID varchar(2) not null default '1',  
  
    constraint content_post_pk primary key (postID),  
    constraint content_post_fk1 foreign key (userID) references UserEntity(userID),  
    constraint content_post_fk2 foreign key (categoryID) references  
Post_Category(categoryID),  
    constraint content_post_fk3 foreign key (statusID) references  
Status_Type(statusID)  
);
```

Post_Reply

```
create table Post_Reply (  
    replyID int not null,  
    userID varchar(5) not null,  
    postID int not null,  
    statusID varchar(2) not null default '1',  
    dateOfReply datetime not null default getdate(),  
    content varchar(120) not null,  
    pr_likes integer not null default 0,  
    pr_dislikes integer not null default 0,  
  
    constraint post_reply_pk primary key (replyID),
```

```

        constraint post_reply_fk1 foreign key (userID) references UserEntity (userID),
        constraint post_reply_fk2 foreign key (postID) references Content_Post (postID),
        constraint post_reply_fk3 foreign key (statusID) references Status_Type (statusID)
    );

```

Product_Category

```

create table Product_Category (
    productCategoryID varchar(2) not null,
    productCategory varchar(20) not null,
    pc_description varchar(50),

    constraint product_category_pk primary key (productCategoryID)
);

```

Product

```

create table Product (
    productID varchar(5) not null,
    productCategoryID varchar(2) not null,
    inventory integer not null,
    productName varchar(50) not null,
    purchasePrice numeric(8,2) not null,
    salePrice numeric(8,2) not null,
    productWeight varchar(20) not null,
    size varchar(20) not null,
    color varchar(16) not null,
    title varchar(50) not null,
    subtitle varchar(80) not null,
    p_description varchar(150) not null,
    image1 varchar(60),
    image2 varchar(60),
    keyWord1 varchar(30) not null,
    keyWord2 varchar(30) not null,

    constraint product_pk primary key (productID),
    constraint product_fk1 foreign key (productCategoryID) references
    Product_Category(productCategoryID)
);

```

Vendor

```

create table Vendor (
    vendorID varchar(5) not null,
    addressID varchar(5) not null,
    vendorName varchar(30) not null,
    officialWebsite varchar(100) not null,
    liasionContactPhone varchar(15) not null,
    V_accountCreateDate datetime not null default getdate(),

    constraint vendor_pk primary key (vendorID),
    constraint vendor_fk1 foreign key (addressID) references AddressEntity(addressID)
);

```


Product_Supply

```
create table Product_Supply (
    vendorID varchar(5) not null,
    productID varchar(5) not null,
    replenish_quantity integer not null,
    supplyStartDate datetime not null default getdate(), -- getdate() default
    supplyEndDate datetime,

    constraint product_supply_pk primary key (vendorID, productID, supplyStartDate),
    constraint product_supply_fk1 foreign key (vendorID) references Vendor(vendorID),
    constraint product_supply_fk2 foreign key (productID) references
Product(productID)
);
```

Coupon

```
create table Coupon (
    couponID varchar(5) not null,
    couponName varchar(20) not null,
    coupon_description varchar(50) not null,
    discount_rate numeric(5,2) not null,
    activeStartDate datetime not null,
    activeEndDate datetime not null,

    constraint coupon_pk primary key (couponID)
);
```

User_Own_Coupo

```
create table User_Own_Coupon (
    couponID varchar(5) not null,
    userID varchar(5) not null,
    quantity integer not null,
    ownDate datetime not null default getdate(),
    couponAvailable varchar(3) not null check(couponAvailable in ('yes','no')), --
only yes or no values hence varchar(3)

    constraint user_oc_pk primary key (couponID,userID),
    constraint user_oc_fk1 foreign key (couponID) references Coupon(couponID),
    constraint user_oc_fk2 foreign key (userID) references UserEntity(userID)
);
```

Payment_Method_Code

```
create table Payment_Method_Code (
    paymentCodeID varchar(2) not null,
    codeName varchar(20) not null,
    code_description varchar(50), --optional

    constraint payment_mc_pk primary key (paymentCodeID)
);
```

Payment_Method

```
create table Payment_Method (  
    paymentMethodID varchar(5) not null,  
    userID varchar(5) not null,  
    paymentCodeID varchar(2) not null,  
    cardNumber varchar(16) not null,  
    expiryDate VARCHAR(5) not null check(expiryDate LIKE '[0-9][0-9]/[0-9][0-9]'),  
    cardKeeper varchar(50) not null,  
    cvvNumber varchar(3) not null check(cvvNumber LIKE '[0-9][0-9][0-9]'),  
  
    constraint payment_method_pk primary key (paymentMethodID),  
    constraint payment_method_fk1 foreign key (userID) references UserEntity(userID),  
    constraint payment_method_fk2 foreign key (paymentCodeID) references  
    Payment_Method_Code(paymentCodeID)  
);
```

Purchase

```
create table Purchase (  
    purchaseID varchar(5) not null,  
    paymentMethodID varchar(5) not null,  
    userID varchar(5) not null,  
    couponID varchar(5) not null default 'CP008',  
    taxRate numeric(5,2) not null default 0.08,  
    deliveryFee numeric(6,2) not null default 0,  
    dateOfPurchase datetime not null default getdate(),  
  
    constraint purchase_pk primary key (purchaseID),  
    constraint purchase_fk1 foreign key (paymentMethodID) references  
    Payment_Method(paymentMethodID),  
    constraint purchase_fk2 foreign key(userID) references UserEntity(userID),  
    constraint purchase_fk3 foreign key(couponID, userID) references  
    User_Own_Coupon(couponID, userID)  
);
```

Purchase_Line

```
create table Purchase_Line (  
    purchaseID varchar(5) not null,  
    productID varchar(5) not null,  
    pl_quantity integer not null,  
  
    constraint purchase_line_pk primary key (productID, purchaseID),  
    constraint purchase_line_fk1 foreign key (productID) references  
    Product(productID),  
    constraint purchase_line_fk2 foreign key (purchaseID) references  
    Purchase(purchaseID)  
);
```

Delivery_Status

```

create table Delivery_Status (
    deliveryStatus varchar(2) not null,
    ds_description varchar(50) not null,

    constraint delivery_status_pk primary key (deliveryStatus)
);

```

Shipping_Status

```

create table Shipping_Status (
    purchaseID varchar(5) not null,
    deliveryStatus varchar(2) not null default '5',
    empID varchar(5) not null,
    courierID varchar(5) not null,
    startDateofShipping datetime not null default getdate(),
    endDateofShipping datetime,

    constraint shipping_status_pk primary key (purchaseID),
    constraint shipping_status_fk1 foreign key (purchaseID) references
Purchase(purchaseID),
    constraint shipping_status_fk2 foreign key (deliveryStatus) references
Delivery_Status(deliveryStatus),
    constraint shipping_status_fk3 foreign key (empID) references Employee(empID),
    constraint shipping_status_fk4 foreign key (courierID) references
Courier(courierID)
);

```

INSERTING TABLE

Address Entity

```

insert into AddressEntity
values
('A0001', '1524', 'East Genesee', '315-728-0234', 'Syracuse', 'NY', 'USA', '13210'),
('A0002', '529', 'Columbus Ave', '315-728-0255', 'Syracuse', 'NY', 'USA', '13210'),
('A0003', '613', 'East Fayette Street', '315-815-0987', 'Syracuse', 'NY', 'USA', '13210'),
('A0004', '715', 'Roosevelt Ave', '805-729-1224', 'Los Angeles', 'CA', 'USA', '12592'),
('A0005', '452', 'West Franklin Street', '707-505-8844', 'New York', 'NY', 'USA', '12334'),
('A0006', '311', 'Ackerman Ave', '404-606-7114', 'New York', 'NY', 'USA', '12501'),
('A0007', '451', 'Maryland Ave', '315-808-8765', 'Syracuse', 'NY', 'USA', '13201'),
('A0008', '1601', 'Clinton Street', '401-808-3361', 'Chicago', 'IL', 'USA', '11987'),
('A0009', '501', 'Adams Street', '391-921-8093', 'Los Angeles', 'CA', 'USA', '15561'),
('A0010', '743', 'Colin Ave', '404-851-4098', 'New York', 'NY', 'USA', '12503'),

('A0011', '451', 'Columbus Ave', '315-653-0087', 'Syracuse', 'NY', 'USA', '13210'),
('A0012', '320', 'Cherry Street', '707-087-6543', 'New York', 'NY', 'USA', '12501'),
('A0013', '400', 'Walnut Ave', '315-645-9890', 'Syracuse', 'NY', 'USA', '13210'),
('A0014', '627', 'Maple Street', '404-723-1234', 'Los Angeles', 'CA', 'USA', '15561'),

('A0015', '829', 'Irwin Ave', '401-592-0104', 'Chicago', 'IL', 'USA', '11987'),
('A0016', '2402', 'W Washington Ave', '402-743-9870', 'Chicago', 'IL', 'USA', '11310'),
('A0017', '413', 'Salt Spring Rd', '600-781-4813', 'Boston', 'MA', 'USA', '14591'),
('A0018', '1052', 'Francis Street', '718-256-3729', 'New York', 'NY', 'USA', '12432'),
('A0019', '726', 'Colin Ave', '745-894-3605', 'New York', 'NY', 'USA', '12305'),
('A0020', '828', 'Lancaster Ave', '400-201-9645', 'Los Angeles', 'CA', 'USA', '15402'),

```

```
( 'A0021', '098', 'Little Ave', '912-762-9120', 'Chicago', 'IL', 'USA', '11310'),
( 'A0022', '27', 'Spirng roll Rd', '127-122-4813', 'Boston', 'MA', 'USA', '14591'),
( 'A0023', '122', 'Frankie St.', '123-378-3729', 'New York', 'NY', 'USA', '12432'),
( 'A0024', '721', 'Strike Ave', '219-923-1231', 'New York', 'NY', 'USA', '12305'),
( 'A0025', '56', 'Matelab Ave', '293-201-1928', 'Los Angeles', 'CA', 'USA', '15402');
```

Access_Type

```
insert into Access_Type
```

```
values
```

```
( '1', 'Visitor', 'visitor does not register an official account'),
( '2', 'User', 'User who does not belong to VIP user'),
( '3', 'VIP User', Null),
( '4', 'Employee', 'Normal Employee'),
( '5', 'Sales Manager', 'Sales Manager has the access to change the sale price of product'),
( '6', 'Administrator', 'The administrator havs the access to change the data that stored in the database');
```

UserEntity

```
insert into UserEntity
```

```
values
```

```
( 'U0001', 'A0001', 'Rohit', null, 'Menon', 'Appu786', 'rmenonva@syr.edu', 'leomessi10', '01/31/2008', null, '2'),
( 'U0002', 'A0002', 'Lawton', 'Ray', 'Zavier', 'ZavierL', 'zavierLR@syr.edu', '1234567890', '12/05/2008', null, '3'),
( 'U0003', 'A0003', 'Sonamgyalpo', null, 'Sherpa', 'Sonam', 'ssherpa@syr.edu', 'atthegym247', '08/10/2012', null, '2'),
( 'U0004', 'A0004', 'Prateek', 'Prabhakar', 'Reddy', 'Pnprabha', 'preddy@syr.edu', 'password', '09/23/2016', null, '2'),
( 'U0005', 'A0005', 'Rachit', null, 'Kaushik', 'MaxImum', 'rakaushi@syr.edu', 'arsenalsucks', '11/5/2016', null, '2'),
( 'U0006', 'A0006', 'Ramiro', null, 'Ying', 'Ying_R', 'yingR@syr.edu', 'guessmypassword', '02/21/2017', null, '2'),
( 'U0007', 'A0007', 'Shaw', null, 'Wan', 'Poppy', 'swan@syr.edu', 'poppy123', '01/12/2019', null, '3'),
( 'U0008', 'A0008', 'Rousseau', null, 'Michael', 'RM001', 'Michael001@gmail.com', 'MichaelR', '02/21/2019', null, '2'),
( 'U0009', 'A0009', 'Shinnick', null, 'Ming', 'M&M', 'Ming@syr.edu', 'MingMing', '11/28/2019', null, '2'),
( 'U0010', 'A0010', 'Kennedy', null, 'Kaushik', 'Kaushik_K', 'rakaushi@syr.edu', 'abcdefg', default, null, '2'),
( 'U0011', 'A0015', 'Anakin', null, 'Skywalker', 'Skywalker_A', 'skywalker@gmail.com', 'darkside2003', '2/12/2008', null, '2'),
( 'U0012', 'A0017', 'Thomas', null, 'Shelby', 'Thomsa', 'tshelby@gmail.com', 'peakyblinder01', '3/10/2012', null, '2');
```

Courier

```
insert into Courier
```

```
values
```

```
( 'C0001', 'A0011', 'FedEx', 'www.fedex.com', '805-729-1224'),
( 'C0002', 'A0012', 'DHL', 'www.dhl.com', '707-505-8844'),
( 'C0003', 'A0013', 'UPS', 'www.ups.com', '404-606-7114'),
( 'C0004', 'A0014', 'DTDC', 'www.dtdc.com', '401-808-3361');
```

Employee

```
insert into Employee
```

```

values
('E0001','A0015','Anakin',null,'Skywalker','dvader@gmail.com','darkside2003','2/10/2008',
null,'6'),
('E0002','A0016','Mudit',null,'Bisht','mbisht@gmail.com','dehradun10','2/15/2012',null,'5'),
('E0003','A0017','Thomas',null,'Shelby','tshelby@gmail.com','peakyblinder01','3/10/2012',
null,'4'),
('E0004','A0018','Aegon','Jon','Targaeryan','atargaryan@gmail.com','winteriscoming2008','
4/16/2017',null,'4'),
('E0005','A0019','Walter',null,'White','wwhite@gmail.com','ilovejesse143','6/10/2019',nul
l,'4'),
('E0006','A0020','Tim',null,'Wykle','wykleTim@gmail.com','WYKLEtimpw','10/21/
2019',null,'4');

```

Post_Category

```
insert into Post_Category
```

```

values
('1','Product Review','Posts related with reviewing product'),
('2','Recommendations','Posts ralted with recommending product'),
('3','Product Selling','User can post their items in this section, including the detailed
information of product, such as price, picture, size and shape'),
('4','Product Buying', null);

```

Status_Type

```
insert into Status_Type
```

```

values
('1','Waiting for Approval'),
('2','Approved'),
('3','Rejected'),
('4','Reported as spam')

```

Content_Post

```
insert into Content_Post
```

```

values
(00001,'U0001','2','Lego Disney Elsa's Market Adventure', '02/03/2015', 'The Lego Disney
Elsa's Market Adventure set lets kids build a boat and market scene where Elsa and Olaf
can buy produce or a pie. Recommend it!', null, null, 102, 100, 10, 15, 2, '2'),
(00002,'U0002','2','Comment on Lego Star Wars', '07/26/2016', 'We like the Lego Star Wars
Ahch-To Island Training kit for beginners because it's a relatively fast build and
includes characters from multiple generations of movies.', null, null, 507, 3500, 376,
256, 1, '2'),
(00003,'U0001','2','Lego Education Community Minifigure Set', '11/25/2017', 'This one
offers nearly two dozen minifigures dressed for occupations such as chef, farmer, parent,
police officer, and veterinarian. ', null, null, 28, 16, 2, 0, 0, '2'),
(00004,'U0008','3','Assassin's Creed Odyssey', '02/03/2018', 'Only played once, contact
me for more details!', null, null, 28, 100, 10, 0, 0, '3'),
(00005,'U0007','2','Childhood Memory: Monopoly', '02/27/2018', 'Monopoly has launched a
new version, recommend you guys to play it!!', null, null, 552, 280, 32, 15, 2, '2'),
(00006,'U0008','1','Disney World Map 1000 Piece Jigsaw Puzzle', '05/16/2018', 'I
purchased this particular puzzle as a gift for my puzzle-loving ex-boyfriend. He loved it
- but then he routinely does puzzles with pieces in the thousands. It would have been too
challenging for me.', null, null, 184, 6702, 41, 21, 8, '2'),
(00007,'U0010','2','The Best Lego Sets for Kids', '07/13/2018', 'My daughter so far
really likes the Friends sets. I know people really gave LEGO a hard time when they came
out because they were more "gendered" if that is a thing but I like that there are sets
my daughter is interested in now.', null, null, 102, 100, 10, 15, 2, '2'),

```

```
(00008,'U0008','1','Lego Disney Elsa's Market Adventure', '08/27/2019', 'These are all super cool, I think the designers and all the people involved deserve a large round of applause. My favorite here has got to be the first room, the woodworking one.', null, null, 398, 10, 1, 1, 0, '1'),
(00009,'U0006','4','Looking for Job?', '01/18/2019', 'Still looking for jobs, come and contact me', null, null, 10, 0, 0, 0, 5, '4'),
(00010,'U0006','1','Looking for Job?', '08/03/2019', 'Still looking for jobs, come and contact me', null, null, 2, 0, 0, 0, 2, '4'),
(00011,'U0003','1','Who can recommend new video games?', '09/25/2019', 'Please recommend some new video games for me!', null, null, 572, 126, 16, 26, 0, '2'),
(00012,'U0006','3','Looking for Job?', '10/21/2019', 'Still looking for jobs, come and contact me', null, null, 0, 0, 0, 0, 0, '1');
```

Post_Reply

```
insert into Post_Reply
values
(00001,'U0002',00002,'2', '07/28/2016', 'Thank you for your recommendation', 15, 2),
(00002,'U0001',00004,'2', '02/15/2018', 'Why UBiSoft cannot spend more on debugging?', 21, 5),
(00003,'U0008',00004,'2', '02/16/2018', 'You are wrong about it, the most funny part is to find the bugs.', 12, default),
(00004,'U0004',00005,'2', '03/01/2017', 'Agree', 3, default),
(00005,'U0002',00011,'2', '09/25/2019', 'Half Life 2', 2, 1),
(00006,'U0002',00011,'2', '09/25/2019', 'Google the question then you can get lots answers', 18, 2),
(00007,'U0008',00011,'2', '09/25/2019', 'Dark Soul', 31, 3),
(00008,'U0007',00011,'2', '09/26/2019', 'Try Witcher 3, you will love it!', 280, 15),
(00009,'U0006',00011,'4', '09/30/2019', 'Still looking for jobs, come and contact me', default, 21),
(00010,'U0001',00010,'2', '10/01/2019', 'Why can t U find a new place to post this?', 12, default);
```

Product_Category

```
insert into Product_Category
values
('1','Model',null),
('2','Puzzle',null),
('3','Comic Book',null),
('4','Video Game',null),
('5','Board Game',null);
```

Product

```
insert into Product
values
('P0001','2',3,'Toy Story Puzzle',8.00,12.00,'1.37 pounds','11.7 x 9 x 1.9','Standard','Buzz Lightyear','A puzzle about Buzz','100 peices puzzle',null,null,'Toy Story','Puzzle'),
('P0002','3',10,'DC Comics',15.00,25.00,'4.8 pounds','10.2 x 1.2 x 12.1','Black','The Killing Joke','An origin story about the Joker','50 pages',null,null,'Comics','DC'),
('P0003','1',15,'Marvel', 15.00,25.00,'2.1 pounds','4 x 4 x 9.1','Red','Iron Man','A toy model of Iron Man', 'With laser light from hands and chest',null,null,'Figure','Marvel'),
('P0004','1',8,'Naruto', 10.00,30.00,'2.2 pounds','4 x 4 x 9.1','Black','Itachi Uchiha','A toy model of Itachi', 'With sharingan eyes',null,null,'Figure','Naruto');
```

```
( 'P0005', '4', 21, 'Assassins Creed', 40.00, 88.00, '3.2 ounces', '0.5 x 5.3 x 6.9', 'White', 'Assasins Creed 2', 'Origins', 'Story of an assassin named Ezio Auditore', null, null, 'Video Game', 'Ezio'),
( 'P0006', '5', 10, 'Monopoly Voice Banking Electronic Family', 12.00, 22.00, '2.05 pounds', '2.5 x 15.8 x 10.5', 'Black', 'Monopoly Voice Banking Electronic Family', 'Monopoly Voice Banking Electronic Family Board Game for Ages 8 & Up', 'The Monopoly Voice Banking game features lights and sounds, and comes with an interactive Mr. Monopoly banking unit.', null, null, 'Monopoly', 'Board Game'),
( 'P0007', '2', 6, 'Wooden Puzzles', 20.00, 38.00, '5 ounces', '7.3 x 6.4 x 0.4', 'Standard', 'Wooden Puzzles', 'Wooden Puzzles Brain Teasers Toy for Kids Adults', '16 Pcs Colorful Hexagon Fun Geometry Logic Tangrams Puzzle Table IQ Game Montessori Educational Toys for Children (Color May Vary)', null, null, 'Puzzle', 'Children');
```

Vendor

```
insert into Vendor
values
```

```
( 'V0001', 'A0021', 'Hasbro', 'www.hasbro.com', '402-4573-9821', '1/20/2019'),
( 'V0002', 'A0022', 'DC', 'www.dccomics.com', '600-214-1243', '4/14/2019'),
( 'V0003', 'A0023', 'Marvel', 'www.marvelcomics.com', '718-834-7732', '2/21/2019'),
( 'V0004', 'A0024', 'Ubisoft', 'www.ubisoft.com', '745-0921-1781', '6/5/2019'),
( 'V0005', 'A0025', 'Anime Store', 'www.animestore.com', '400-657-8873', '4/9/2019')
```

Product_Supply

```
insert into Product_Supply
values
```

```
( 'V0001', 'P0001', 20, '05/12/2019', '05/24/2019'),
( 'V0005', 'P0001', 15, '05/17/2019', '05/22/2019'),
( 'V0002', 'P0002', 8, '06/17/2019', '06/28/2019'),
( 'V0003', 'P0003', 15, '07/17/2019', '07/21/2019'),
( 'V0004', 'P0005', 18, '07/17/2019', '07/23/2019'),
( 'V0005', 'P0006', 10, '07/17/2019', '08/02/2019'),
( 'V0001', 'P0007', 12, '11/17/2019', null),
( 'V0001', 'P0001', 25, '11/17/2019', null);
```

Coupon

```
insert into Coupon
values
```

```
( 'CP001', '50OFF', '50% off, include tax and shipping', 0.50, '12/25/2017', '12/27/2017'),
( 'CP002', 'CUT35', '35% off', 0.65, '12/02/2017', '01/02/2018'),
( 'CP003', 'THANKS', '75% off, include tax and shipping', 0.25, '11/29/2018', '11/29/2018'),
( 'CP004', 'BLACKFRI', '60% off', 0.40, '11/29/2019', '11/30/2019'),
( 'CP005', 'GREYOFFER', '25% off, include tax and shipping', 0.75, '01/01/2020', '02/01/2020'),
( 'CP006', 'EMPLOYEE', '15% off, include tax and shipping', 0.85, '01/01/2008', '12/21/2028'),
( 'CP007', 'DAILY', '5% off', 0.95, '01/01/2019', '12/31/2019'),
( 'CP008', 'NOCONPON', 'No Conpon', 1.00, '01/01/2012', '12/31/2088');
```

User_Own_Coupon

```
insert into User_Own_Coupon
values
```

```
( 'CP001', 'U0001', 1, '12/25/2017', 'no'),
( 'CP001', 'U0002', 1, '12/25/2017', 'no'),
( 'CP003', 'U0003', 3, '11/29/2018', 'no'),
( 'CP002', 'U0003', 1, '12/02/2017', 'no'),
( 'CP004', 'U0008', 1, '11/29/2019', 'no'),
( 'CP004', 'U0009', 1, '11/29/2019', 'no'),
( 'CP001', 'U0011', 2, '12/25/2017', 'no');
```

```
( 'CP006' , 'U0011' , 5 , '02/12/2008' , 'yes' ),
( 'CP006' , 'U0012' , 3 , '03/10/2012' , 'yes' ),
( 'CP007' , 'U0001' , 1 , '01/01/2019' , 'yes' ),
( 'CP007' , 'U0003' , 1 , '01/01/2019' , 'yes' ),
( 'CP007' , 'U0008' , 7 , '01/01/2019' , 'yes' ),
( 'CP007' , 'U0011' , 5 , '01/01/2019' , 'yes' ),
( 'CP007' , 'U0012' , 2 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0001' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0002' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0003' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0004' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0005' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0006' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0007' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0008' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0009' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0010' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0011' , 1 , '01/01/2019' , 'yes' ),
( 'CP008' , 'U0012' , 1 , '01/01/2019' , 'yes' );
```

Payment_Method_Code

```
insert into Payment_Method_Code
values
( '1' , 'CC' , 'Credit Card' ),
( '2' , 'DC' , 'Debit Card' );
```

Payment_Method

```
insert into Payment_Method
values
( 'PM001' , 'U0001' , '1' , '5543177618965567' , '07/24' , 'Rohit Menon' , '987' ),
( 'PM002' , 'U0002' , '2' , '4432886503458765' , '06/25' , 'Lawton Ray Xavier' , '463' ),
( 'PM003' , 'U0003' , '1' , '5832817498221800' , '12/21' , 'Sonamgyalpo Sherpa' , '348' ),
( 'PM004' , 'U0004' , '1' , '1256178523483954' , '04/22' , 'Prateek Reddy' , '862' ),
( 'PM005' , 'U0005' , '2' , '2837482643785378' , '08/24' , 'Rachit Prabhakar' , '623' ),
( 'PM007' , 'U0007' , '2' , '1278235633642383' , '01/23' , 'Wan Shaw' , '359' ),
( 'PM008' , 'U0008' , '1' , '1256372363728254' , '04/22' , 'Rousseau Michael' , '782' ),
( 'PM009' , 'U0009' , '2' , '1278235633642383' , '01/23' , 'Shinnick Ming' , '206' ),
( 'PM010' , 'U0010' , '2' , '4782243745442343' , '11/28' , 'Kennedy Kaushik' , '949' ),
( 'PM011' , 'U0011' , '1' , '2348222342423483' , '07/28' , 'Anakin Anakin' , '229' ),
( 'PM012' , 'U0012' , '1' , '6824373434276783' , '08/22' , 'Thomas Shelby' , '125' );
```

Purchase

```
insert into Purchase
values
( 'PC001' , 'PM001' , 'U0001' , 'CP001' , 0.08 , 8.00 , '12/25/2017' ),
( 'PC002' , 'PM002' , 'U0002' , 'CP008' , 0.03 , 0.00 , '02/21/2018' ),
( 'PC003' , 'PM002' , 'U0002' , 'CP008' , 0.08 , 0.00 , '02/26/2018' ),
( 'PC004' , 'PM002' , 'U0002' , 'CP008' , 0.08 , 0.00 , '05/12/2018' ),
( 'PC005' , 'PM001' , 'U0001' , 'CP008' , 0.08 , 5.00 , '05/26/2018' ),
( 'PC006' , 'PM002' , 'U0002' , 'CP008' , 0.05 , 0.00 , '06/01/2018' ),
( 'PC007' , 'PM004' , 'U0004' , 'CP008' , 0.10 , 2.00 , '06/07/2018' ),
( 'PC008' , 'PM005' , 'U0005' , 'CP008' , 0.08 , 0.00 , '06/12/2018' ),
( 'PC009' , 'PM003' , 'U0003' , 'CP008' , 0.05 , 6.00 , '06/18/2018' ),
( 'PC010' , 'PM005' , 'U0005' , 'CP008' , 0.08 , 9.00 , '06/18/2018' ),
( 'PC011' , 'PM004' , 'U0004' , 'CP008' , 0.00 , 0.00 , '06/19/2018' ),
( 'PC012' , 'PM004' , 'U0004' , 'CP008' , 0.08 , 8.00 , '06/19/2018' ),
```



```
( 'PC013', 'PM004', 'U0004', 'CP008', 0.08, 5.00, '06/29/2018' ),
( 'PC014', 'PM001', 'U0001', 'CP008', 0.07, 2.00, '07/10/2018' ),
( 'PC015', 'PM005', 'U0005', 'CP008', 0.00, 5.00, '07/10/2018' ),
( 'PC016', 'PM003', 'U0003', 'CP008', 0.11, 8.00, '07/12/2018' ),
( 'PC017', 'PM002', 'U0002', 'CP008', 0.08, 0.00, '07/19/2018' ),
( 'PC018', 'PM004', 'U0004', 'CP008', 0.06, 5.00, '07/22/2018' ),
( 'PC019', 'PM003', 'U0003', 'CP008', 0.00, 0.00, '07/23/2018' ),
( 'PC020', 'PM004', 'U0004', 'CP008', 0.08, 8.00, '08/20/2018' ),
( 'PC021', 'PM002', 'U0002', 'CP008', 0.08, 0.00, '08/22/2018' ),
( 'PC022', 'PM003', 'U0003', 'CP003', 0.08, 21.00, '10/11/2018' ),
( 'PC023', 'PM004', 'U0004', 'CP008', 0.06, 10.00, '10/13/2018' ),
( 'PC024', 'PM005', 'U0005', 'CP008', 0.10, 10.00, '11/02/2018' ),
( 'PC025', 'PM008', 'U0008', 'CP008', 0.08, 9.00, '11/11/2018' ),
( 'PC026', 'PM002', 'U0002', 'CP008', 0.08, 0.00, '11/11/2018' ),
( 'PC027', 'PM003', 'U0003', 'CP008', 0.08, 2.00, '11/11/2018' ),
( 'PC028', 'PM004', 'U0004', 'CP008', 0.06, 2.00, '12/02/2018' ),
( 'PC029', 'PM004', 'U0004', 'CP008', 0.10, 0.00, '12/08/2018' ),
( 'PC030', 'PM007', 'U0007', 'CP008', 0.08, 0.00, '12/12/2018' ),
( 'PC031', 'PM002', 'U0002', 'CP008', 0.08, 0.00, '12/21/2018' ),
( 'PC032', 'PM001', 'U0001', 'CP008', 0.08, 2.00, '12/22/2018' ),
( 'PC033', 'PM001', 'U0001', 'CP008', 0.06, 9.00, '01/02/2019' ),
( 'PC034', 'PM002', 'U0002', 'CP008', 0.10, 0.00, '01/06/2019' ),
( 'PC035', 'PM003', 'U0003', 'CP008', 0.08, 9.00, '01/12/2019' ),
( 'PC036', 'PM002', 'U0002', 'CP008', 0.08, 0.00, '01/15/2019' ),
( 'PC037', 'PM001', 'U0001', 'CP008', 0.08, 2.00, '01/18/2019' ),
( 'PC038', 'PM003', 'U0003', 'CP008', 0.06, 9.00, '02/13/2019' ),
( 'PC039', 'PM003', 'U0003', 'CP008', 0.10, 0.00, '02/23/2019' ),
( 'PC040', 'PM007', 'U0007', 'CP008', 0.08, 0.00, '02/24/2019' ),
( 'PC041', 'PM008', 'U0008', 'CP008', 0.08, 8.00, '02/26/2019' ),
( 'PC042', 'PM002', 'U0002', 'CP008', 0.08, 0.00, '02/27/2019' ),
( 'PC043', 'PM007', 'U0007', 'CP008', 0.06, 0.00, '03/07/2019' ),
( 'PC044', 'PM008', 'U0008', 'CP008', 0.10, 12.00, '03/11/2019' ),
( 'PC045', 'PM009', 'U0009', 'CP008', 0.08, 9.00, '03/14/2019' ),
( 'PC046', 'PM007', 'U0007', 'CP008', 0.08, 0.00, '04/23/2019' ),
( 'PC047', 'PM002', 'U0002', 'CP008', 0.06, 0.00, '04/28/2019' ),
( 'PC048', 'PM004', 'U0004', 'CP008', 0.10, 8.00, '05/05/2019' ),
( 'PC049', 'PM009', 'U0009', 'CP008', 0.08, 9.00, '05/05/2019' ),
( 'PC050', 'PM008', 'U0008', 'CP008', 0.08, 2.00, '05/06/2019' ),
( 'PC051', 'PM005', 'U0005', 'CP008', 0.06, 6.00, '05/13/2019' ),
( 'PC052', 'PM001', 'U0001', 'CP008', 0.10, 8.00, '05/23/2019' ),
( 'PC053', 'PM012', 'U0012', 'CP008', 0.08, 9.00, '05/29/2019' ),
( 'PC054', 'PM012', 'U0012', 'CP008', 0.08, 6.00, '06/02/2019' ),
( 'PC055', 'PM011', 'U0011', 'CP008', 0.06, 9.00, '06/10/2019' ),
( 'PC056', 'PM007', 'U0007', 'CP008', 0.08, 0.00, '06/17/2019' ),
( 'PC057', 'PM004', 'U0004', 'CP008', 0.06, 9.00, '07/08/2019' ),
( 'PC058', 'PM011', 'U0011', 'CP008', 0.08, 2.00, '07/08/2019' ),
( 'PC059', 'PM007', 'U0007', 'CP008', 0.06, 0.00, '07/12/2019' ),
( 'PC060', 'PM011', 'U0011', 'CP008', 0.10, 12.00, '07/26/2019' ),
( 'PC061', 'PM005', 'U0005', 'CP008', 0.08, 9.00, '08/01/2019' ),
( 'PC062', 'PM011', 'U0011', 'CP008', 0.08, 6.00, '08/02/2019' ),
( 'PC063', 'PM008', 'U0008', 'CP008', 0.08, 9.00, '08/03/2019' );
```

Purchase_Line

insert into Purchase_Line

values

('PC001', 'P0001', 2),

('PC001', 'P0002', 1),

('PC001' , 'P0003' , 12),
('PC002' , 'P0002' , 1),
('PC003' , 'P0003' , 10),
('PC004' , 'P0003' , 2),
('PC004' , 'P0004' , 3),
('PC005' , 'P0001' , 2),
('PC006' , 'P0002' , 18),
('PC006' , 'P0006' , 1),
('PC007' , 'P0002' , 2),
('PC008' , 'P0002' , 3),
('PC008' , 'P0004' , 5),
('PC008' , 'P0005' , 2),
('PC008' , 'P0007' , 5),
('PC009' , 'P0005' , 7),
('PC010' , 'P0004' , 2),
('PC010' , 'P0006' , 8),
('PC011' , 'P0002' , 1),
('PC012' , 'P0007' , 7),
('PC013' , 'P0001' , 3),
('PC014' , 'P0005' , 24),
('PC015' , 'P0002' , 1),
('PC015' , 'P0003' , 2),
('PC015' , 'P0005' , 5),
('PC016' , 'P0004' , 2),
('PC017' , 'P0002' , 10),
('PC018' , 'P0003' , 1),
('PC018' , 'P0007' , 6),
('PC019' , 'P0004' , 8),
('PC019' , 'P0006' , 2),
('PC020' , 'P0002' , 22),
('PC021' , 'P0005' , 2),
('PC021' , 'P0006' , 2),
('PC021' , 'P0007' , 3),
('PC022' , 'P0007' , 3),
('PC023' , 'P0002' , 1),
('PC024' , 'P0004' , 5),
('PC025' , 'P0006' , 2),
('PC026' , 'P0001' , 2),
('PC027' , 'P0002' , 1),
('PC028' , 'P0003' , 12),
('PC028' , 'P0006' , 5),
('PC029' , 'P0002' , 1),
('PC030' , 'P0002' , 2),
('PC030' , 'P0006' , 7),
('PC030' , 'P0007' , 1),
('PC031' , 'P0003' , 2),
('PC032' , 'P0004' , 3),
('PC033' , 'P0001' , 2),
('PC034' , 'P0002' , 18),
('PC034' , 'P0006' , 1),
('PC035' , 'P0006' , 1),
('PC036' , 'P0002' , 2),
('PC037' , 'P0002' , 3),
('PC037' , 'P0006' , 6),
('PC038' , 'P0002' , 2),
('PC038' , 'P0005' , 5),
('PC038' , 'P0007' , 12),
('PC039' , 'P0005' , 2),

```

('PC040','P0007', 5),
('PC041','P0005', 7),
('PC042','P0002', 10),
('PC042','P0006', 12),
('PC042','P0007', 3),
('PC043','P0006', 8),
('PC044','P0002', 1),
('PC045','P0002', 12),
('PC045','P0007', 7),
('PC046','P0001', 3),
('PC047','P0005', 24),
('PC048','P0002', 1),
('PC049','P0003', 2),
('PC050','P0005', 5),
('PC051','P0005', 8),
('PC051','P0007', 2),
('PC052','P0002', 10),
('PC053','P0003', 1),
('PC054','P0007', 6),
('PC055','P0003', 12),
('PC055','P0004', 8),
('PC056','P0006', 2),
('PC057','P0001', 5),
('PC057','P0002', 6),
('PC057','P0005', 22),
('PC058','P0005', 2),
('PC059','P0002', 2),
('PC059','P0003', 8),
('PC060','P0001', 3),
('PC060','P0005', 10),
('PC061','P0007', 3),
('PC062','P0002', 1),
('PC062','P0004', 5),
('PC062','P0006', 2),
('PC063','P0006', 5),
('PC063','P0007', 10),
('PC063','P0004', 15),
('PC063','P0005', 3),
('PC063','P0003', 3);

```

Delivery_Status

```

insert into Delivery_Status
values
('1','Delivery completed'),
('2','Out of warehouse'),
('3','Damaged'),
('4','Lost'),
('5','Preparing for shipping');

```

Shipping_Status

```

insert into Shipping_Status
values
('PC001','1','E0001','C0002','12/25/2017','12/30/2017'),
('PC002','1','E0003','C0002','02/21/2018','03/08/2018'),
('PC003','4','E0002','C0004','02/26/2018', NULL),
('PC004','1','E0003','C0001','07/02/2018','07/12/2018'),
('PC005','1','E0006','C0002','05/26/2018','05/30/2018'),

```

('PC006', '1', 'E0001', 'C0003', '06/01/2018', '06/07/2018'),
('PC007', '1', 'E0001', 'C0004', '06/07/2018', '06/09/2018'),
('PC008', '1', 'E0002', 'C0001', '06/12/2018', '06/15/2018'),
('PC009', '4', 'E0002', 'C0002', '06/18/2018', NULL),
('PC010', '1', 'E0003', 'C0001', '06/18/2018', '06/22/2018'),
('PC011', '1', 'E0006', 'C0002', '06/19/2018', '06/25/2018'),
('PC012', '1', 'E0001', 'C0004', '06/19/2018', '06/26/2018'),
('PC013', '1', 'E0001', 'C0001', '06/29/2018', '06/30/2018'),
('PC014', '3', 'E0001', 'C0002', '07/10/2018', NULL),
('PC015', '1', 'E0001', 'C0001', '07/10/2018', '07/13/2018'),
('PC016', '1', 'E0001', 'C0002', '07/12/2018', '07/15/2018'),
('PC017', '1', 'E0002', 'C0002', '07/19/2018', '07/21/2018'),
('PC018', '1', 'E0003', 'C0004', '07/22/2018', '07/26/2018'),
('PC019', '1', 'E0002', 'C0001', '07/23/2018', '07/28/2018'),
('PC020', '1', 'E0001', 'C0002', '08/20/2018', '08/23/2018'),
('PC021', '1', 'E0001', 'C0002', '08/22/2018', '08/27/2018'),
('PC022', '1', 'E0001', 'C0004', '10/11/2018', '10/16/2018'),
('PC023', '1', 'E0001', 'C0001', '10/13/2018', '10/17/2018'),
('PC024', '3', 'E0004', 'C0002', '11/02/2018', NULL),
('PC025', '3', 'E0003', 'C0002', '11/11/2018', NULL),
('PC026', '3', 'E0006', 'C0004', '11/11/2018', NULL),
('PC027', '1', 'E0001', 'C0001', '11/11/2018', '11/14/2018'),
('PC028', '1', 'E0001', 'C0002', '12/02/2018', '12/12/2018'),
('PC029', '1', 'E0001', 'C0004', '12/08/2018', '12/16/2018'),
('PC030', '3', 'E0001', 'C0001', '12/12/2018', '12/16/2018'),
('PC031', '1', 'E0001', 'C0002', '12/21/2018', '12/28/2018'),
('PC032', '1', 'E0004', 'C0004', '12/22/2018', '12/26/2018'),
('PC033', '3', 'E0005', 'C0001', '01/02/2019', NULL),
('PC034', '1', 'E0002', 'C0002', '01/06/2019', '01/06/2019'),
('PC035', '1', 'E0004', 'C0003', '01/12/2019', '01/14/2019'),
('PC036', '1', 'E0005', 'C0004', '01/15/2019', '01/16/2019'),
('PC037', '1', 'E0002', 'C0002', '01/18/2019', '01/21/2019'),
('PC038', '1', 'E0001', 'C0004', '02/13/2019', '02/16/2019'),
('PC039', '1', 'E0001', 'C0001', '02/23/2019', '02/27/2019'),
('PC040', '1', 'E0004', 'C0002', '02/24/2019', '02/28/2019'),
('PC041', '4', 'E0005', 'C0004', '02/26/2019', NULL),
('PC042', '1', 'E0002', 'C0001', '02/27/2019', '02/27/2019'),
('PC043', '1', 'E0001', 'C0002', '03/07/2019', '03/14/2019'),
('PC044', '1', 'E0002', 'C0003', '03/11/2019', '03/15/2019'),
('PC045', '4', 'E0004', 'C0004', '03/14/2019', NULL),
('PC046', '1', 'E0005', 'C0001', '04/23/2019', '04/28/2019'),
('PC047', '3', 'E0002', 'C0002', '04/28/2019', NULL),
('PC048', '1', 'E0001', 'C0004', '05/05/2019', '05/11/2019'),
('PC049', '1', 'E0001', 'C0001', '05/05/2019', '05/08/2019'),
('PC050', '1', 'E0004', 'C0002', '05/06/2019', '05/09/2019'),
('PC051', '1', 'E0005', 'C0004', '05/13/2019', '05/18/2019'),
('PC052', '1', 'E0002', 'C0001', '05/23/2019', '05/26/2019'),
('PC053', '1', 'E0004', 'C0002', '05/29/2019', '05/29/2019'),
('PC054', '3', 'E0005', 'C0002', '06/02/2019', NULL),
('PC055', '1', 'E0002', 'C0002', '06/10/2019', '06/15/2019'),
('PC056', '1', 'E0006', 'C0002', '06/17/2019', '06/19/2019'),
('PC057', '1', 'E0002', 'C0004', '07/08/2019', '07/12/2019'),
('PC058', '1', 'E0001', 'C0001', '07/08/2019', '07/12/2019'),
('PC059', '1', 'E0004', 'C0002', '07/12/2019', '07/17/2019'),
('PC060', '1', 'E0005', 'C0004', '07/26/2019', '07/31/2019'),
('PC061', '1', 'E0002', 'C0001', '08/01/2019', '08/07/2019'),
('PC062', '1', 'E0006', 'C0002', '08/02/2019', '08/12/2019'),
('PC063', '2', 'E0005', 'C0003', '08/03/2019', NULL);

OUTPUT DISPLAY

AddressEntity

70 %

Results Messages

	addressID	streetNo	streetName	contactPhone	city	stateName	country	zipcode
1	A0001	1524	East Genesee	315-728-0234	Syracuse	NY	USA	13210
2	A0002	529	Columbus Ave	315-728-0255	Syracuse	NY	USA	13210
3	A0003	613	East Fayette Street	315-815-0987	Syracuse	NY	USA	13210
4	A0004	715	Roosevelt Ave	805-729-1224	Los Angeles	CA	USA	12592
5	A0005	452	West Franklin Street	707-505-8844	New York	NY	USA	12334
6	A0006	311	Ackerman Ave	404-606-7114	New York	NY	USA	12501
7	A0007	451	Maryland Ave	315-808-8765	Syracuse	NY	USA	13201
8	A0008	1601	Clinton Street	401-808-3361	Chicago	IL	USA	11987
9	A0009	501	Adams Street	391-921-8093	Los Angeles	CA	USA	15561
10	A0010	743	Colin Ave	404-851-4098	New York	NY	USA	12503
11	A0011	451	Columbus Ave	315-653-0087	Syracuse	NY	USA	13210
12	A0012	320	Cherry Street	707-087-6543	New York	NY	USA	12501
13	A0013	400	Walnut Ave	315-645-9890	Syracuse	NY	USA	13210
14	A0014	627	Maple Street	404-723-1234	Los Angeles	CA	USA	15561
15	A0015	829	Irwin Ave	401-592-0104	Chicago	IL	USA	11987
16	A0016	2402	W Washington Ave	402-743-9870	Chicago	IL	USA	11310
17	A0017	413	Salt Spring Rd	600-781-4813	Boston	MA	USA	14591
18	A0018	1052	Francis Street	718-256-3729	New York	NY	USA	12432
19	A0019	726	Colin Ave	745-894-3605	New York	NY	USA	12305
20	A0020	828	Lancaster Ave	400-201-9645	Los Angeles	CA	USA	15402
21	A0021	098	Little Ave	912-762-9120	Chicago	IL	USA	11310
22	A0022	27	Spring roll Rd	127-122-4813	Boston	MA	USA	14591
23	A0023	122	Frankie St.	123-378-3729	New York	NY	USA	12432
24	A0024	721	Strike Ave	219-923-1231	New York	NY	USA	12305
25	A0025	56	Matelab Ave	293-201-1928	Los Angeles	CA	USA	15402
26	A0026	New U...	New User	New User	New User	New User	New ...	New ...

Query executed successfully.

ist-s-students.syr.edu (12....) AD\rmenonva (102) IST659_M004_rmenonva 00:00:00 26 rows

Access_Type

Results		Messages	
	accessType	typeName	at_description
1	1	Visitor	visitor does not register an official account
2	2	User	User who does not belong to VIP user
3	3	VIP User	NULL
4	4	Employee	Normal Employee
5	5	Sales Manager	Sales Manager has the access to change the sale ...
6	6	Administrator	The administrator has the access to change the d...

UserEntity

Results Messages											
	userID	addressID	fName	mName	lName	nName	email	hashedPassword	accountCreateDate	profilePicture	accessType
1	U0001	A0001	Rohit	NULL	Menon	Appu786	rmenonva@syrr.edu	leomessi10	2008-01-31 00:00:00.000	NULL	2
2	U0002	A0002	Lawton	NULL	Zavier	ZavierL	zavier.LR@syrr.edu	1234567890	2008-12-05 00:00:00.000	NULL	3
3	U0003	A0003	Sonamgyalpo	NULL	Sherpa	Sonam	ssherpa@syrr.edu	atthegym247	2012-08-10 00:00:00.000	NULL	2
4	U0004	A0004	Prateek	Prabhakar	Reddy	Pnprabha	preddy@syrr.edu	password	2016-09-23 00:00:00.000	NULL	2
5	U0005	A0005	Rachit	NULL	Kaushik	Maximum	rakaushi@syrr.edu	arsenalsucks	2016-11-05 00:00:00.000	NULL	2
6	U0006	A0006	Ramiro	NULL	Ying	Ying_R	yingR@syrr.edu	guessmypassword	2017-02-21 00:00:00.000	NULL	2
7	U0007	A0007	Shaw	NULL	Wan	Poppy	swan@syrr.edu	poppy123	2019-01-12 00:00:00.000	NULL	3
8	U0008	A0008	Rousseau	NULL	Michael	RM001	Michael001@gmail.com	MichaelR	2019-02-21 00:00:00.000	NULL	2
9	U0009	A0009	Shinnick	NULL	Ming	M&M	Ming@syrr.edu	MingMing	2019-11-28 00:00:00.000	NULL	2
10	U0010	A0010	Kennedy	NULL	Kaushik	Kaushik_K	rakaushi@syrr.edu	abcdefg	2019-12-01 23:09:11.853	NULL	2
11	U0011	A0015	Anakin	NULL	Skywalker	Skywalker_A	skywalker@gmail.com	darkside2003	2008-02-12 00:00:00.000	NULL	2
12	U0012	A0017	Thomas	NULL	Shelby	Thomsa	tshelby@gmail.com	peakyblinder01	2012-03-10 00:00:00.000	NULL	2
Query executed successfully. ist-s-students.syr.edu (12.... AD/rmenonva (102) IST659_M004_rmenonva 00:00:00 12 rows											

Courier

70 %					
Results Messages					
	courierID	addressID	companyName	officialWebsite	contactPhone
1	C0001	A0011	FedEx	www.fedex.com	805-729-1224
2	C0002	A0012	DHL	www.dhl.com	707-505-8844
3	C0003	A0013	UPS	www.ups.com	404-606-7114
4	C0004	A0014	DTDC	www.dtdc.com	401-808-3361

Query executed successfully. ist-s-students.syr.edu (12.... AD\rmenonva (102) IST659_M004_rmenonva 00:00:00 4 rows

Employee

Results Messages										
	empID	addressID	E_FName	E_MName	E_LName	E_email	E_hashedPassword	E_accountCreateDate	E_profilePicture	accessType
1	E0001	A0015	Anakin	NULL	Skywalker	dvader@gmail.com	darkside2003	2008-02-10 00:00:00.000	NULL	6
2	E0002	A0016	Mudit	NULL	Bisht	mbisht@gmail.com	dehradun10	2012-02-15 00:00:00.000	NULL	5
3	E0003	A0017	Thomas	NULL	Shelby	tshelby@gmail.com	peakyblinder01	2012-03-10 00:00:00.000	NULL	4
4	E0004	A0018	Aegon	Jon	Targaeryan	atargaryan@gmail.com	winteriscoming2008	2017-04-16 00:00:00.000	NULL	4
5	E0005	A0019	Walter	NULL	White	wwhite@gmail.com	ilovejesse143	2019-06-10 00:00:00.000	NULL	4
6	E0006	A0020	Tim	NULL	Wykle	wykleTim@gmail.com	WYKLEtmpsw	2019-10-21 00:00:00.000	NULL	4

Query executed successfully. ist-s-students.syr.edu (12.... AD\rmenonva (102) IST659_M004_rmenonva 00:00:00 6 rows

Post_Category

Results Messages			
	categoryID	categoryName	categoryDescription
1	1	Product Review	Posts related with reviewing product
2	2	Recommendations	Posts ralated with recommending product
3	3	Product Selling	User can post their items in this section, inclu...
4	4	Product Buying	NULL

Query executed successfully. ist-s-students.syr.edu (12.... AD\rmenonva (102) IST659_M004_rmenonva 00:00:00 4 rows

Status_Type

Results Messages	
	statusID statusDescription
1	1 Waiting for Approval
2	2 Approved
3	3 Rejected
4	4 Reported as spam

Query executed successfully. ist-s-students.syr.edu (12.... AD\rmenonva (102) IST659_M004_rmenonva 00:00:00 4 rows

Content_Post

Results

Messages

	postID	userID	categoryID	postTitle	dateOfPost	postContent	picture1	picture2	numberOfView	totalRatingScore	ratingUserNumber	likes	dislikes	statusID
1	1	U0001	2	Lego Disney Elsa'...	2015-02-03 0...	The Lego Disney ...	NULL	NULL	102	100	10	15	2	2
2	2	U0002	2	Comment on Leg...	2016-07-26 0...	We like the Lego ...	NULL	NULL	507	3500	376	256	1	2
3	3	U0001	2	Lego Education C...	2017-11-25 0...	This one offers ne...	NULL	NULL	28	16	2	0	0	2
4	4	U0008	3	Assassin's Creed ...	2018-02-03 0...	Only played once, ...	NULL	NULL	28	100	10	0	0	3
5	5	U0007	2	Childhood Memor...	2018-02-27 0...	Monopoly has lau...	NULL	NULL	552	280	32	15	2	2
6	6	U0008	1	Disney World Ma...	2018-05-16 0...	I purchased this p...	NULL	NULL	184	6702	41	21	8	2
7	7	U0010	2	The Best Lego S...	2018-07-13 0...	My daughter so fa...	NULL	NULL	102	100	10	15	2	2
8	8	U0008	1	Lego Disney Elsa'...	2019-08-27 0...	These are all supe...	NULL	NULL	398	10	1	1	0	1
9	9	U0006	4	Looking for Job?	2019-01-18 0...	Still looking for job...	NULL	NULL	10	0	0	0	5	4
10	10	U0006	1	Looking for Job?	2019-08-03 0...	Still looking for job...	NULL	NULL	2	0	0	0	2	4
11	11	U0003	1	Who can recomm...	2019-09-25 0...	Please recommen...	NULL	NULL	572	126	16	26	0	2
12	12	U0006	3	Looking for Job?	2019-10-21 0...	Still looking for job...	NULL	NULL	0	0	0	0	0	1

Query executed successfully.

ist-s-students.syr.edu (12....

AD\rmenonva (102)

IST659_M004_rmenonva

00:00:00

12 rows

Post_Reply

Results Messages									
	replyID	userID	postID	statusID	dateOfReply	content	pr_likes	pr_dislikes	
1	1	U0002	2	2	2016-07-28 00:00:00.000	Thank you for your recommendation	15	2	
2	2	U0001	4	2	2018-02-15 00:00:00.000	Why UBISoft cannot spend more on debugging?	21	5	
3	3	U0008	4	2	2018-02-16 00:00:00.000	You are wrong about it, the most funny part is to f...	12	0	
4	4	U0004	5	2	2017-03-01 00:00:00.000	Agree	3	0	
5	5	U0002	11	2	2019-09-25 00:00:00.000	Half Life 2	2	1	
6	6	U0002	11	2	2019-09-25 00:00:00.000	Google the question then you can get lots answers	18	2	
7	7	U0008	11	2	2019-09-25 00:00:00.000	Dark Soul	31	3	
8	8	U0007	11	2	2019-09-26 00:00:00.000	Try Witcher 3, you will love it!	280	15	
9	9	U0006	11	4	2019-09-30 00:00:00.000	Still looking for jobs, come and contact me	0	21	
10	10	U0001	10	2	2019-10-01 00:00:00.000	Why can t U find a new place to post this?	12	0	

Query executed successfully. ist-s-students.syr.edu (12.... AD\rmenonva (102) IST659_M004_rmenonva 00:00:00 10 rows

Product_Category

Results Messages			
	productCategoryID	productCategory	pc_description
1	1	Model	NULL
2	2	Puzzle	NULL
3	3	Comic Book	NULL
4	4	Video Game	NULL
5	5	Board Game	NULL

Query executed successfully. ist-s-students.syr.edu (12.... AD\rmenonva (102) 1

Product

Results		Messages													
	productID	productCategoryID	inventory	productName	purchasePrice	salePrice	productWeight	size	color	title	subtitle	p_description	image1	image2	keyWord1
1	P0001	2	3	Toy Story Pu...	8.00	12.00	1.37 pounds	11.7 ...	Standard	Buzz ...	A puz...	100 peices p...	NULL	NULL	Toy Story
2	P0002	3	10	DC Comics	15.00	25.00	4.8 pounds	10.2 ...	Black	The K...	An ori...	50 pages	NULL	NULL	Comics
3	P0003	1	15	Marvel	15.00	25.00	2.1 pounds	4 x 4 ...	Red	Iron ...	A toy ...	With laser lig...	NULL	NULL	Figure
4	P0004	1	8	Naruto	10.00	30.00	2.2 pounds	4 x 4 ...	Black	Itachi...	A toy ...	With sharing...	NULL	NULL	Figure
5	P0005	4	21	Assassins Cr...	40.00	88.00	3.2 ounces	0.5 x ...	White	Assa...	Origins	Story of an a...	NULL	NULL	Video Game
6	P0006	5	10	Monopoly Voi...	12.00	22.00	2.05 pounds	2.5 x ...	Black	Mono...	Mono...	The Monopol...	NULL	NULL	Monopoly
7	P0007	2	6	Wooden Puz...	20.00	38.00	5 ounces	7.3 x ...	Standard	Woo...	Wood...	16 Pcs Colorf...	NULL	NULL	Puzzle

Query executed successfully. | ist-s-students.syr.edu (12.... | AD\rmenonva (102) | IST659_M004_rmenonva | 00:00:00 | 7 rows

Vendor

Results		Messages				
	vendorID	addressID	vendorName	officialWebsite	liasionContactPhone	V_accountCreateDate
1	V0001	A0021	Hasbro	www.hasbro.com	402-4573-9821	2019-01-20 00:00:00.000
2	V0002	A0022	DC	www.dccomics.com	600-214-1243	2019-04-14 00:00:00.000
3	V0003	A0023	Marvel	www.marvelcomics.com	718-834-7732	2019-02-21 00:00:00.000
4	V0004	A0024	Ubisoft	www.ubisoft.com	745-0921-1781	2019-06-05 00:00:00.000
5	V0005	A0025	Anime Store	www.animestore.com	400-657-8873	2019-04-09 00:00:00.000

Query executed successfully. | ist-s-students.syr.edu (12.... | AD\rmenonva (102)

Product_Supply

Results		Messages			
	vendorID	productID	replenish_quantity	supplyStartDate	supplyEndDate
1	V0001	P0001	20	2019-05-12 00:00:00.000	2019-05-24 00:00:00.000
2	V0001	P0001	25	2019-11-17 00:00:00.000	NULL
3	V0001	P0007	12	2019-11-17 00:00:00.000	NULL
4	V0002	P0002	8	2019-06-17 00:00:00.000	2019-06-28 00:00:00.000
5	V0003	P0003	15	2019-07-17 00:00:00.000	2019-07-21 00:00:00.000
6	V0004	P0005	18	2019-07-17 00:00:00.000	2019-07-23 00:00:00.000
7	V0005	P0001	15	2019-05-17 00:00:00.000	2019-05-22 00:00:00.000
8	V0005	P0006	10	2019-07-17 00:00:00.000	2019-08-02 00:00:00.000

Query executed successfully. | ist-s-students.syr.edu (12.... | AD\rmenonva (102)

Coupon

Results Messages

	couponID	couponName	coupon_description	discount_rate	activeStartDate	activeEndDate
1	CP001	50OFF	50% off, include tax and shipping	0.50	2017-12-25 00:00:00.000	2017-12-27 00:00:00.000
2	CP002	CUT35	35% off	0.65	2017-12-02 00:00:00.000	2018-01-02 00:00:00.000
3	CP003	THANKS	75% off, include tax and shipping	0.25	2018-11-29 00:00:00.000	2018-11-29 00:00:00.000
4	CP004	BLACKFRI	60% off	0.40	2019-11-29 00:00:00.000	2019-11-30 00:00:00.000
5	CP005	GREYOFFER	25% off, include tax and shipping	0.75	2020-01-01 00:00:00.000	2020-02-01 00:00:00.000
6	CP006	EMPLOYEE	15% off, include tax and shipping	0.85	2008-01-01 00:00:00.000	2028-12-21 00:00:00.000
7	CP007	DAILY	5% off	0.95	2019-01-01 00:00:00.000	2019-12-31 00:00:00.000
8	CP008	NOCONPON	No Conpon	1.00	2012-01-01 00:00:00.000	2088-12-31 00:00:00.000

Query executed successfully.

ist-s-students.syr.edu (12....AD\rmenonva (102)

User_Own_Coupon

Results

Messages

	couponID	userID	quantity	ownDate	couponAvailable
1	CP001	U0001	1	2017-12-25 00:00:00.000	no
2	CP001	U0002	1	2017-12-25 00:00:00.000	no
3	CP001	U0011	2	2017-12-25 00:00:00.000	no
4	CP002	U0003	1	2017-12-02 00:00:00.000	no
5	CP003	U0003	3	2018-11-29 00:00:00.000	no
6	CP004	U0008	1	2019-11-29 00:00:00.000	no
7	CP004	U0009	1	2019-11-29 00:00:00.000	no
8	CP006	U0011	5	2008-02-12 00:00:00.000	yes
9	CP006	U0012	3	2012-03-10 00:00:00.000	yes
10	CP007	U0001	1	2019-01-01 00:00:00.000	yes
11	CP007	U0003	1	2019-01-01 00:00:00.000	yes
12	CP007	U0008	7	2019-01-01 00:00:00.000	yes
13	CP007	U0011	5	2019-01-01 00:00:00.000	yes
14	CP007	U0012	2	2019-01-01 00:00:00.000	yes
15	CP008	U0001	1	2019-01-01 00:00:00.000	yes
16	CP008	U0002	1	2019-01-01 00:00:00.000	yes
17	CP008	U0003	1	2019-01-01 00:00:00.000	yes
18	CP008	U0004	1	2019-01-01 00:00:00.000	yes
19	CP008	U0005	1	2019-01-01 00:00:00.000	yes
20	CP008	U0006	1	2019-01-01 00:00:00.000	yes
21	CP008	U0007	1	2019-01-01 00:00:00.000	yes
22	CP008	U0008	1	2019-01-01 00:00:00.000	yes
23	CP008	U0009	1	2019-01-01 00:00:00.000	yes
24	CP008	U0010	1	2019-01-01 00:00:00.000	yes
25	CP008	U0011	1	2019-01-01 00:00:00.000	yes
26	CP008	U0012	1	2019-01-01 00:00:00.000	yes

Query executed successfully.

ist-s-students.syr.edu (12.... AD\rmenonva (102)

Payment_Method_Code

Results		Messages	
	paymentCodeID	codeName	code_description
1	1	CC	Credit Card
2	2	DC	Debit Card

Query executed successfully.

ist-s-students.syr.edu (12.... AD\rmenonva (102)

Payment_Method

Results

Messages

	paymentMethodID	userID	paymentCodeID	cardNumber	expiryDate	cardKeeper	cwNumber
1	PM001	U0001	1	5543177618965567	07/24	Rohit Menon	987
2	PM002	U0002	2	4432886503458765	06/25	Lawton Ray Xavier	463
3	PM003	U0003	1	5832817498221800	12/21	Sonamgyalpo Sherpa	348
4	PM004	U0004	1	1256178523483954	04/22	Prateek Reddy	862
5	PM005	U0005	2	2837482643785378	08/24	Rachit Prabhakar	623
6	PM007	U0007	2	1278235633642383	01/23	Wan Shaw	359
7	PM008	U0008	1	1256372363728254	04/22	Rousseau Michael	782
8	PM009	U0009	2	1278235633642383	01/23	Shinnick Ming	206
9	PM010	U0010	2	4782243745442343	11/28	Kennedy Kaushik	949
10	PM011	U0011	1	2348222342423483	07/28	Anakin Anakin	229
11	PM012	U0012	1	6824373434276783	08/22	Thomas Shelby	125

Query executed successfully.

ist-s-students.syr.edu (12.... | AD\rmenonva (102)

Purchase

Results

Messages

	purchaseID	paymentMethodID	userID	couponID	taxRate	deliveryFee	dateOfPurchase
1	PC001	PM001	U0001	CP001	0.08	8.00	2017-12-25 00:00:00.000
2	PC002	PM002	U0002	CP008	0.03	0.00	2018-02-21 00:00:00.000
3	PC003	PM002	U0002	CP008	0.08	0.00	2018-02-26 00:00:00.000
4	PC004	PM002	U0002	CP008	0.08	0.00	2018-05-12 00:00:00.000
5	PC005	PM001	U0001	CP008	0.08	5.00	2018-05-26 00:00:00.000
6	PC006	PM002	U0002	CP008	0.05	0.00	2018-06-01 00:00:00.000
7	PC007	PM004	U0004	CP008	0.10	2.00	2018-06-07 00:00:00.000
8	PC008	PM005	U0005	CP008	0.08	0.00	2018-06-12 00:00:00.000
9	PC009	PM003	U0003	CP008	0.05	6.00	2018-06-18 00:00:00.000
10	PC010	PM005	U0005	CP008	0.08	9.00	2018-06-18 00:00:00.000
11	PC011	PM004	U0004	CP008	0.00	0.00	2018-06-19 00:00:00.000
12	PC012	PM004	U0004	CP008	0.08	8.00	2018-06-19 00:00:00.000
13	PC013	PM004	U0004	CP008	0.08	5.00	2018-06-29 00:00:00.000
14	PC014	PM001	U0001	CP008	0.07	2.00	2018-07-10 00:00:00.000
15	PC015	PM005	U0005	CP008	0.00	5.00	2018-07-10 00:00:00.000
16	PC016	PM003	U0003	CP008	0.11	8.00	2018-07-12 00:00:00.000
17	PC017	PM002	U0002	CP008	0.08	0.00	2018-07-19 00:00:00.000
18	PC018	PM004	U0004	CP008	0.06	5.00	2018-07-22 00:00:00.000
19	PC019	PM003	U0003	CP008	0.00	0.00	2018-07-23 00:00:00.000
20	PC020	PM004	U0004	CP008	0.08	8.00	2018-08-20 00:00:00.000
21	PC021	PM002	U0002	CP008	0.08	0.00	2018-08-22 00:00:00.000
22	PC022	PM003	U0003	CP003	0.08	21.00	2018-10-11 00:00:00.000
23	PC023	PM004	U0004	CP008	0.06	10.00	2018-10-13 00:00:00.000
24	PC024	PM005	U0005	CP008	0.10	10.00	2018-11-02 00:00:00.000
25	PC025	PM008	U0008	CP008	0.08	9.00	2018-11-11 00:00:00.000
26	PC026	PM002	U0002	CP008	0.08	0.00	2018-11-11 00:00:00.000

Query executed successfully.

ist-s-students.syr.edu (12...

AD\rmenonva (102)

70 %

Results Messages

	purchaseID	paymentMethodID	userID	couponID	taxRate	deliveryFee	dateOfPurchase
27	PC027	PM003	U0003	CP008	0.08	2.00	2018-11-11 00:00:00.000
28	PC028	PM004	U0004	CP008	0.06	2.00	2018-12-02 00:00:00.000
29	PC029	PM004	U0004	CP008	0.10	0.00	2018-12-08 00:00:00.000
30	PC030	PM007	U0007	CP008	0.08	0.00	2018-12-12 00:00:00.000
31	PC031	PM002	U0002	CP008	0.08	0.00	2018-12-21 00:00:00.000
32	PC032	PM001	U0001	CP008	0.08	2.00	2018-12-22 00:00:00.000
33	PC033	PM001	U0001	CP008	0.06	9.00	2019-01-02 00:00:00.000
34	PC034	PM002	U0002	CP008	0.10	0.00	2019-01-06 00:00:00.000
35	PC035	PM003	U0003	CP008	0.08	9.00	2019-01-12 00:00:00.000
36	PC036	PM002	U0002	CP008	0.08	0.00	2019-01-15 00:00:00.000
37	PC037	PM001	U0001	CP008	0.08	2.00	2019-01-18 00:00:00.000
38	PC038	PM003	U0003	CP008	0.06	9.00	2019-02-13 00:00:00.000
39	PC039	PM003	U0003	CP008	0.10	0.00	2019-02-23 00:00:00.000
40	PC040	PM007	U0007	CP008	0.08	0.00	2019-02-24 00:00:00.000
41	PC041	PM008	U0008	CP008	0.08	8.00	2019-02-26 00:00:00.000
42	PC042	PM002	U0002	CP008	0.08	0.00	2019-02-27 00:00:00.000
43	PC043	PM007	U0007	CP008	0.06	0.00	2019-03-07 00:00:00.000
44	PC044	PM008	U0008	CP008	0.10	12.00	2019-03-11 00:00:00.000
45	PC045	PM009	U0009	CP008	0.08	9.00	2019-03-14 00:00:00.000
46	PC046	PM007	U0007	CP008	0.08	0.00	2019-04-23 00:00:00.000
47	PC047	PM002	U0002	CP008	0.06	0.00	2019-04-28 00:00:00.000
48	PC048	PM004	U0004	CP008	0.10	8.00	2019-05-05 00:00:00.000
49	PC049	PM009	U0009	CP008	0.08	9.00	2019-05-05 00:00:00.000
50	PC050	PM008	U0008	CP008	0.08	2.00	2019-05-06 00:00:00.000
51	PC051	PM005	U0005	CP008	0.06	6.00	2019-05-13 00:00:00.000
52	PC052	PM001	U0001	CP008	0.10	8.00	2019-05-23 00:00:00.000

Query executed successfully.

ist-s-students.syr.edu (12.... AD\rmenonva (102)

Results Messages

	purchaseID	paymentMethodID	userID	couponID	taxRate	deliveryFee	dateOfPurchase
39	PC039	PM003	U0003	CP008	0.10	0.00	2019-02-23 00:00:00.000
40	PC040	PM007	U0007	CP008	0.08	0.00	2019-02-24 00:00:00.000
41	PC041	PM008	U0008	CP008	0.08	8.00	2019-02-26 00:00:00.000
42	PC042	PM002	U0002	CP008	0.08	0.00	2019-02-27 00:00:00.000
43	PC043	PM007	U0007	CP008	0.06	0.00	2019-03-07 00:00:00.000
44	PC044	PM008	U0008	CP008	0.10	12.00	2019-03-11 00:00:00.000
45	PC045	PM009	U0009	CP008	0.08	9.00	2019-03-14 00:00:00.000
46	PC046	PM007	U0007	CP008	0.08	0.00	2019-04-23 00:00:00.000
47	PC047	PM002	U0002	CP008	0.06	0.00	2019-04-28 00:00:00.000
48	PC048	PM004	U0004	CP008	0.10	8.00	2019-05-05 00:00:00.000
49	PC049	PM009	U0009	CP008	0.08	9.00	2019-05-05 00:00:00.000
50	PC050	PM008	U0008	CP008	0.08	2.00	2019-05-06 00:00:00.000
51	PC051	PM005	U0005	CP008	0.06	6.00	2019-05-13 00:00:00.000
52	PC052	PM001	U0001	CP008	0.10	8.00	2019-05-23 00:00:00.000
53	PC053	PM012	U0012	CP008	0.08	9.00	2019-05-29 00:00:00.000
54	PC054	PM012	U0012	CP008	0.08	6.00	2019-06-02 00:00:00.000
55	PC055	PM011	U0011	CP008	0.06	9.00	2019-06-10 00:00:00.000
56	PC056	PM007	U0007	CP008	0.08	0.00	2019-06-17 00:00:00.000
57	PC057	PM004	U0004	CP008	0.06	9.00	2019-07-08 00:00:00.000
58	PC058	PM011	U0011	CP008	0.08	2.00	2019-07-08 00:00:00.000
59	PC059	PM007	U0007	CP008	0.06	0.00	2019-07-12 00:00:00.000
60	PC060	PM011	U0011	CP008	0.10	12.00	2019-07-26 00:00:00.000
61	PC061	PM005	U0005	CP008	0.08	9.00	2019-08-01 00:00:00.000
62	PC062	PM011	U0011	CP008	0.08	6.00	2019-08-02 00:00:00.000
63	PC063	PM008	U0008	CP008	0.08	9.00	2019-08-03 00:00:00.000

Query executed successfully.

ist-s-students.syr.edu (12.... AD\rmenonva (102)

Purchase_Line

	purchaseID	productID	pl_quantity
1	PC001	P0001	2
2	PC005	P0001	2
3	PC013	P0001	3
4	PC026	P0001	2
5	PC033	P0001	2
6	PC046	P0001	3
7	PC057	P0001	5
8	PC060	P0001	3
9	PC001	P0002	1
10	PC002	P0002	1
11	PC006	P0002	18
12	PC007	P0002	2
13	PC008	P0002	3
14	PC011	P0002	1
15	PC015	P0002	1
16	PC017	P0002	10
17	PC020	P0002	22
18	PC023	P0002	1
19	PC027	P0002	1
20	PC029	P0002	1
21	PC030	P0002	2
22	PC034	P0002	18
23	PC036	P0002	2
24	PC037	P0002	3
25	PC038	P0002	2
26	PC042	P0002	10

Query executed successfully.

	purchaseID	productID	pl_quantity
26	PC042	P0002	10
27	PC044	P0002	1
28	PC045	P0002	12
29	PC048	P0002	1
30	PC052	P0002	10
31	PC057	P0002	6
32	PC059	P0002	2
33	PC062	P0002	1
34	PC001	P0003	12
35	PC003	P0003	10
36	PC004	P0003	2
37	PC015	P0003	2
38	PC018	P0003	1
39	PC028	P0003	12
40	PC031	P0003	2
41	PC049	P0003	2
42	PC053	P0003	1
43	PC055	P0003	12
44	PC059	P0003	8
45	PC063	P0003	3
46	PC004	P0004	3
47	PC008	P0004	5
48	PC010	P0004	2
49	PC016	P0004	2
50	PC019	P0004	8
51	PC024	P0004	5

Query executed successfully.

	purchaseID	productID	pl_quantity
51	PC024	P0004	5
52	PC032	P0004	3
53	PC055	P0004	8
54	PC062	P0004	5
55	PC063	P0004	15
56	PC008	P0005	2
57	PC009	P0005	7
58	PC014	P0005	24
59	PC015	P0005	5
60	PC021	P0005	2
61	PC038	P0005	5
62	PC039	P0005	2
63	PC041	P0005	7
64	PC047	P0005	24
65	PC050	P0005	5
66	PC051	P0005	8
67	PC057	P0005	22
68	PC058	P0005	2
69	PC060	P0005	10
70	PC063	P0005	3
71	PC006	P0006	1
72	PC010	P0006	8
73	PC019	P0006	2
74	PC021	P0006	2
75	PC025	P0006	2
76	PC028	P0006	5

Query executed successfully.

	purchaseID	productID	pl_quantity
75	PC025	P0006	2
76	PC028	P0006	5
77	PC030	P0006	7
78	PC034	P0006	1
79	PC035	P0006	1
80	PC037	P0006	6
81	PC042	P0006	12
82	PC043	P0006	8
83	PC056	P0006	2
84	PC062	P0006	2
85	PC063	P0006	5
86	PC008	P0007	5
87	PC012	P0007	7
88	PC018	P0007	6
89	PC021	P0007	3
90	PC022	P0007	3
91	PC030	P0007	1
92	PC038	P0007	12
93	PC040	P0007	5
94	PC042	P0007	3
95	PC045	P0007	7
96	PC051	P0007	2
97	PC054	P0007	6
98	PC061	P0007	3
99	PC063	P0007	10

Query executed successfully.

Delivery_Status

	deliveryStatus	ds_description
1	1	Delivery completed
2	2	Out of warehouse
3	3	Damaged
4	4	Lost
5	5	Preparing for shipping

Query executed successfully.

Shipping_Status

70 %						
Results Messages						
	purchaseID	deliveryStatus	empID	courierID	startDateofShipping	endDateofShipping
1	PC001	1	E0001	C0002	2017-12-25 00:00:00.000	2017-12-30 00:00:00.000
2	PC002	1	E0003	C0002	2018-02-21 00:00:00.000	2018-03-08 00:00:00.000
3	PC003	4	E0002	C0004	2018-02-26 00:00:00.000	NULL
4	PC004	1	E0003	C0001	2018-07-02 00:00:00.000	2018-07-12 00:00:00.000
5	PC005	1	E0006	C0002	2018-05-26 00:00:00.000	2018-05-30 00:00:00.000
6	PC006	1	E0001	C0003	2018-06-01 00:00:00.000	2018-06-07 00:00:00.000
7	PC007	1	E0001	C0004	2018-06-07 00:00:00.000	2018-06-09 00:00:00.000
8	PC008	1	E0002	C0001	2018-06-12 00:00:00.000	2018-06-15 00:00:00.000
9	PC009	4	E0002	C0002	2018-06-18 00:00:00.000	NULL
10	PC010	1	E0003	C0001	2018-06-18 00:00:00.000	2018-06-22 00:00:00.000
11	PC011	1	E0006	C0002	2018-06-19 00:00:00.000	2018-06-25 00:00:00.000
12	PC012	1	E0001	C0004	2018-06-19 00:00:00.000	2018-06-26 00:00:00.000
13	PC013	1	E0001	C0001	2018-06-29 00:00:00.000	2018-06-30 00:00:00.000
14	PC014	3	E0001	C0002	2018-07-10 00:00:00.000	NULL
15	PC015	1	E0001	C0001	2018-07-10 00:00:00.000	2018-07-13 00:00:00.000
16	PC016	1	E0001	C0002	2018-07-12 00:00:00.000	2018-07-15 00:00:00.000
17	PC017	1	E0002	C0002	2018-07-19 00:00:00.000	2018-07-21 00:00:00.000
18	PC018	1	E0003	C0004	2018-07-22 00:00:00.000	2018-07-26 00:00:00.000
19	PC019	1	E0002	C0001	2018-07-23 00:00:00.000	2018-07-28 00:00:00.000
20	PC020	1	E0001	C0002	2018-08-20 00:00:00.000	2018-08-23 00:00:00.000
21	PC021	1	E0001	C0002	2018-08-22 00:00:00.000	2018-08-27 00:00:00.000
22	PC022	1	E0001	C0004	2018-10-11 00:00:00.000	2018-10-16 00:00:00.000
23	PC023	1	E0001	C0001	2018-10-13 00:00:00.000	2018-10-17 00:00:00.000
24	PC024	3	E0004	C0002	2018-11-02 00:00:00.000	NULL
25	PC025	3	E0003	C0002	2018-11-11 00:00:00.000	NULL
26	PC026	3	E0006	C0004	2018-11-11 00:00:00.000	NULL

Query executed successfully.

ist-s-student

Results		Messages				
	purchaseID	deliveryStatus	empID	courierID	startDateofShipping	endDateofShipping
26	PC026	3	E0006	C0004	2018-11-11 00:00:00.000	NULL
27	PC027	1	E0001	C0001	2018-11-11 00:00:00.000	2018-11-14 00:00:00.000
28	PC028	1	E0001	C0002	2018-12-02 00:00:00.000	2018-12-12 00:00:00.000
29	PC029	1	E0001	C0004	2018-12-08 00:00:00.000	2018-12-16 00:00:00.000
30	PC030	3	E0001	C0001	2018-12-12 00:00:00.000	2018-12-16 00:00:00.000
31	PC031	1	E0001	C0002	2018-12-21 00:00:00.000	2018-12-28 00:00:00.000
32	PC032	1	E0004	C0004	2018-12-22 00:00:00.000	2018-12-26 00:00:00.000
33	PC033	3	E0005	C0001	2019-01-02 00:00:00.000	NULL
34	PC034	1	E0002	C0002	2019-01-06 00:00:00.000	2019-01-06 00:00:00.000
35	PC035	1	E0004	C0003	2019-01-12 00:00:00.000	2019-01-14 00:00:00.000
36	PC036	1	E0005	C0004	2019-01-15 00:00:00.000	2019-01-16 00:00:00.000
37	PC037	1	E0002	C0002	2019-01-18 00:00:00.000	2019-01-21 00:00:00.000
38	PC038	1	E0001	C0004	2019-02-13 00:00:00.000	2019-02-16 00:00:00.000
39	PC039	1	E0001	C0001	2019-02-23 00:00:00.000	2019-02-27 00:00:00.000
40	PC040	1	E0004	C0002	2019-02-24 00:00:00.000	2019-02-28 00:00:00.000
41	PC041	4	E0005	C0004	2019-02-26 00:00:00.000	NULL
42	PC042	1	E0002	C0001	2019-02-27 00:00:00.000	2019-02-27 00:00:00.000
43	PC043	1	E0001	C0002	2019-03-07 00:00:00.000	2019-03-14 00:00:00.000
44	PC044	1	E0002	C0003	2019-03-11 00:00:00.000	2019-03-15 00:00:00.000
45	PC045	4	E0004	C0004	2019-03-14 00:00:00.000	NULL
46	PC046	1	E0005	C0001	2019-04-23 00:00:00.000	2019-04-28 00:00:00.000
47	PC047	3	E0002	C0002	2019-04-28 00:00:00.000	NULL
48	PC048	1	E0001	C0004	2019-05-05 00:00:00.000	2019-05-11 00:00:00.000
49	PC049	1	E0001	C0001	2019-05-05 00:00:00.000	2019-05-08 00:00:00.000
50	PC050	1	E0004	C0002	2019-05-06 00:00:00.000	2019-05-09 00:00:00.000
51	PC051	1	E0005	C0004	2019-05-13 00:00:00.000	2019-05-18 00:00:00.000
Query executed successfully.						ist-s-students

70 %						
Results Messages						
	purchaseID	deliveryStatus	empID	courierID	startDateofShipping	endDateofShipping
39	PC039	1	E0001	C0001	2019-02-23 00:00:00.000	2019-02-27 00:00:00.000
40	PC040	1	E0004	C0002	2019-02-24 00:00:00.000	2019-02-28 00:00:00.000
41	PC041	4	E0005	C0004	2019-02-26 00:00:00.000	NULL
42	PC042	1	E0002	C0001	2019-02-27 00:00:00.000	2019-02-27 00:00:00.000
43	PC043	1	E0001	C0002	2019-03-07 00:00:00.000	2019-03-14 00:00:00.000
44	PC044	1	E0002	C0003	2019-03-11 00:00:00.000	2019-03-15 00:00:00.000
45	PC045	4	E0004	C0004	2019-03-14 00:00:00.000	NULL
46	PC046	1	E0005	C0001	2019-04-23 00:00:00.000	2019-04-28 00:00:00.000
47	PC047	3	E0002	C0002	2019-04-28 00:00:00.000	NULL
48	PC048	1	E0001	C0004	2019-05-05 00:00:00.000	2019-05-11 00:00:00.000
49	PC049	1	E0001	C0001	2019-05-05 00:00:00.000	2019-05-08 00:00:00.000
50	PC050	1	E0004	C0002	2019-05-06 00:00:00.000	2019-05-09 00:00:00.000
51	PC051	1	E0005	C0004	2019-05-13 00:00:00.000	2019-05-18 00:00:00.000
52	PC052	1	E0002	C0001	2019-05-23 00:00:00.000	2019-05-26 00:00:00.000
53	PC053	1	E0004	C0002	2019-05-29 00:00:00.000	2019-05-29 00:00:00.000
54	PC054	3	E0005	C0002	2019-06-02 00:00:00.000	NULL
55	PC055	1	E0002	C0002	2019-06-10 00:00:00.000	2019-06-15 00:00:00.000
56	PC056	1	E0006	C0002	2019-06-17 00:00:00.000	2019-06-19 00:00:00.000
57	PC057	1	E0002	C0004	2019-07-08 00:00:00.000	2019-07-12 00:00:00.000
58	PC058	1	E0001	C0001	2019-07-08 00:00:00.000	2019-07-12 00:00:00.000
59	PC059	1	E0004	C0002	2019-07-12 00:00:00.000	2019-07-17 00:00:00.000
60	PC060	1	E0005	C0004	2019-07-26 00:00:00.000	2019-07-31 00:00:00.000
61	PC061	1	E0002	C0001	2019-08-01 00:00:00.000	2019-08-07 00:00:00.000
62	PC062	1	E0006	C0002	2019-08-02 00:00:00.000	2019-08-12 00:00:00.000
63	PC063	2	E0005	C0003	2019-08-03 00:00:00.000	NULL

Query executed successfully.

ist-s-student



Major Data Questions

The following data questions have been answered by using SQL Server and Access Report Function. Answering these six major questions can help manager to deeply understand different aspects of current situation, ranging from user information investigation to benefit management.

1. User Information Investigation: Who are the users of the forum? What their personal information?

Access Query

```
SELECT
u.userID,
u.fName & ' ' & u.mName & ' ' & u.lName as "User Name",
u.nName as "Nick Name",
a.streetNo & ' ' & a.streetName & ' ' & a.city & ' ' & a.stateName & ' ' & a.country & ' (' &
a.zipcode & ')' as "Address",
t.typeName AS "User Type",
u.email AS "Email Address",
Format (u.accountCreateDate, "yyyy/mm/dd") AS "Account Create Date"
FROM dbo_UserEntity u, dbo_AddressEntity a, dbo_Access_Type t
WHERE u.accessType = t.accessType and u.addressID = a.addressID
ORDER BY u.userID, u.accountCreateDate;
```

Output of SQL Server

99 %

Results

Messages

	userID	User Name	Nick Name	Address	User Type	Email Address	Account Create Date
1	U0001	Rohit Menon	Appu786	1524, East Genessee, Syracuse, NY, USA, (13210)	User	menonva@syr.edu	2008-01-31
2	U0002	Lawton Ray Xavier	ZavierL	529, Columbus Ave, Syracuse, NY, USA, (13210)	VIP User	xavierLR@syr.edu	2008-12-05
3	U0003	Sonamgyalpo Sherpa	Sonam	613, East Fayette Street, Syracuse, NY, USA, (132...	User	ssherpa@syr.edu	2012-08-10
4	U0004	Prateek Prabhakar Reddy	Pnprabha	715, Roosevelt Ave, Los Angeles, CA, USA, (12592)	User	preddy@syr.edu	2016-09-23
5	U0005	Rachit Kaushik	Maximum	452, West Franklin Street, New York, NY, USA, (1...	User	rakaushi@syr.edu	2016-11-05
6	U0006	Ramiro Ying	Ying_R	311, Ackeman Ave, New York, NY, USA, (12501)	User	yingR@syr.edu	2017-02-21
7	U0007	Shaw Wan	Poppy	451, Maryland Ave, Syracuse, NY, USA, (13201)	VIP User	swan@syr.edu	2019-01-12
8	U0008	Rousseau Michael	RM001	1601, Clinton Street, Chicago, IL, USA, (11987)	User	Michael001@gmail.com	2019-02-21
9	U0009	Shinnick Ming	M&M	501, Adams Street, Los Angeles, CA, USA, (15561)	User	Ming@syr.edu	2019-11-28
10	U0010	Kennedy Kaushik	Kaushik_K	743, Colin Ave, New York, NY, USA, (12503)	User	rakaushi@syr.edu	2019-11-30
11	U0011	Anakin Skywalker	Skywalker_A	829, Irwin Ave, Chicago, IL, USA, (11987)	User	skywalker@gmail.com	2008-02-12
12	U0012	Thomas Shelby	Thomsa	413, Salt Spring Rd, Boston, MA, USA, (14591)	User	tshelby@gmail.com	2012-03-10

Query executed successfully.

ist-s-students.syr.edu (12...

AD\xwanyue (80)

IST659

Access Report Output

User Information Report

UserID	UserName	NickName	UserType	AccountCreateDate
U0001	Rohit Menon	Appu786	User	2008/01/31
EmailAddress			Address	
rmenonva@syr.edu			1524, East Genesee, Syracuse, NY, USA, (13210)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0002	Lawton Ray Zavi	ZavierL	VIP User	2008/12/05
EmailAddress			Address	
zavierLR@syr.edu			529, Columbus Ave, Syracuse, NY, USA, (13210)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0003	Sonamgyalpo Sh	Sonam	User	2012/08/10
EmailAddress			Address	
ssherpa@syr.edu			613, East Fayette Street, Syracuse, NY, USA, (13210)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0004	Prateek Prabhak	Pnprabha	User	2016/09/23
EmailAddress			Address	
preddy@syr.edu			715, Roosevelt Ave, Los Angeles, CA, USA, (12592)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0005	Rachit Kaushik	Maxlimum	User	2016/11/05
EmailAddress			Address	
rakaushi@syr.edu			452, West Franklin Street, New York, NY, USA, (12334)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0006	Ramiro Ying	Ying_R	User	2017/02/21
EmailAddress			Address	
yingR@syr.edu			311, Ackerman Ave, New York, NY, USA, (12501)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0007	Shaw Wan	Poppy	VIP User	2019/01/12
EmailAddress			Address	
swan@syr.edu			451, Maryland Ave, Syracuse, NY, USA, (13201)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0008	Rousseau Micha	RM001	User	2019/02/21
EmailAddress			Address	
Michael001@gmail.com			1601, Clinton Street, Chicago, IL, USA, (11987)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0009	Shinnick Ming	M&M	User	2019/11/28
EmailAddress			Address	
Ming@syr.edu			501, Adams Street, Los Angeles, CA, USA, (15561)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0010	Kennedy Kaushi	Kaushik_K	User	2019/11/30
EmailAddress			Address	
rakaushi@syr.edu			743, Colin Ave, New York, NY, USA, (12503)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0011	Anakin Skywalker	Skywalker_A	User	2008/02/12
EmailAddress			Address	
skywalker@gmail.com			829, Irwin Ave, Chicago, IL, USA, (11987)	

UserID	UserName	NickName	UserType	AccountCreateDate
U0012	Thomas Shelby	Thomsa	User	2012/03/10
EmailAddress			Address	
tshelby@gmail.com			413, Salt Spring Rd, Boston, MA, USA, (14591)	

2. User Sales Performance: Which users have purchased products from the forum previously?
The result should also include the total expenditure generated by each user.

Access Query

```
SELECT u.userID,
u.fName & ' ' & u.mName & ' ' & u.lName as "User Name",
u.nName AS "Nick Name",
t.typeName AS "User Type",
Format (re.Total_Expenditure, "Currency") as "Expenditure"
FROM dbo_UserEntity u, dbo_Access_Type t, (SELECT u.userID, SUM((pl.pl_quantity *
p.salePrice * (1 + pc.taxRate) * cp.discount_rate) + pc.deliveryFee) AS Total_Expenditure
FROM dbo_Product p, dbo_Purchase pc, dbo_Purchase_Line pl, dbo_UserEntity u, dbo_Coupon
cp
WHERE pc.purchaseID = pl.purchaseID and p.productID = pl.productID and u.userID =
pc.userID and pc.couponID = cp.couponID
GROUP BY u.userID) re
WHERE u.accessType = t.accessType and u.userID = re.userID
```

Output of SQL Server

99 %						
		Results	Messages			
	userID	User Name	Nick Name	User Type	Total Expenditure	
1	U0001	Rohit Menon	Appu786	User	\$3,149.42	
2	U0002	Lawton Ray Xavier	ZavierL	VIP User	\$5,143.35	
3	U0003	Sonamgyalpo Sherpa	Sonam	User	\$2,348.30	
4	U0004	Prateek Prabhakar Reddy	Pnprabha	User	\$4,141.20	
5	U0005	Rachit Kaushik	Maxlmum	User	\$2,587.08	
6	U0007	Shaw Wan	Poppy	VIP User	\$1,274.52	
7	U0008	Rousseau Michael	RM001	User	\$2,668.38	
8	U0009	Shinnick Ming	M&M	User	\$692.28	
9	U0011	Anakin Skywalker	Skywalker_A	User	\$2,068.60	
10	U0012	Thomas Shelby	Thomsa	User	\$288.24	

Query executed successfully.

Access Report

User Expenditure Report

UserID	UserName	NickName	User Type	Expenditure
U0001	Rohit Menon	Appu786	User	\$3,149.42
U0002	Lawton Ray Xavier	ZavierL	VIP User	\$5,143.35
U0003	Sonamgyalpo Sherpa	Sonam	User	\$2,348.30
U0004	Prateek Prabhakar Reddy	Pnprabha	User	\$4,141.20
U0005	Rachit Kaushik	Maximum	User	\$2,587.08
U0007	Shaw Wan	Poppy	VIP User	\$1,274.52
U0008	Rousseau Michael	RM001	User	\$2,668.38
U0009	Shinnick Ming	M&M	User	\$692.28
U0011	Anakin Skywalker	Skywalker_A	User	\$2,068.60
U0012	Thomas Shelby	Thomsa	User	\$288.24

Saturday, November 30, 2019

Page 1 of 1

3. Shopping List: What products have purchased by the user> The result should summarize the product quantity and the total expenditure for each purchase line?

Access Query

```

SELECT
pc.purchaseID,
u.fName & ' ' & u.mName & ' ' & u.lName as "User Name",
p.productName & ' - ' & p.title AS "Product Name",
pl.pl_quantity,
FORMAT(p.salePrice,"Currency") as "Price for Each Product",
FORMAT(((pl.pl_quantity * p.salePrice * (1 + pc.taxRate) * cp.discount_rate) + pc.deliveryFee),
"Currency") AS "Total Expenditure for Purchase Line",
Format(pc.dateOfPurchase, "yyyy/mm/dd") as "Purchase Date"
FROM dbo_Product p, dbo_Purchase pc, dbo_Purchase_Line pl, dbo_UserEntity u, dbo_Coupon
cp
WHERE pc.purchaseID = pl.purchaseID and p.productID = pl.productID and u.userID =
pc.userID and pc.couponID = cp.couponID
ORDER BY u.userID, pc.purchaseID;

```

Output of SQL Server

99 %

ResultsMessages

	purchaseID	User Name	Product Name	pl_quantity	Price for Each Product	Total Expenditure for Purchase Line	Purchase Date
1	PC001	Rohit Menon	Toy Story Puzzle - Buzz Lightyear	2	\$12.00	\$20.96	2017-12-25
2	PC001	Rohit Menon	DC Comics - The Killing Joke	1	\$25.00	\$21.50	2017-12-25
3	PC001	Rohit Menon	Marvel - Iron Man	12	\$25.00	\$170.00	2017-12-25
4	PC005	Rohit Menon	Toy Story Puzzle - Buzz Lightyear	2	\$12.00	\$30.92	2018-05-26
5	PC014	Rohit Menon	Assassins Creed - Assassins Creed 2	24	\$88.00	\$2,261.84	2018-07-10
6	PC032	Rohit Menon	Naruto - Itachi Uchiha	3	\$30.00	\$99.20	2018-12-22
7	PC033	Rohit Menon	Toy Story Puzzle - Buzz Lightyear	2	\$12.00	\$34.44	2019-01-02
8	PC037	Rohit Menon	DC Comics - The Killing Joke	3	\$25.00	\$83.00	2019-01-18
9	PC037	Rohit Menon	Monopoly Voice Banking Electronic Family - Monop...	6	\$22.00	\$144.56	2019-01-18
10	PC052	Rohit Menon	DC Comics - The Killing Joke	10	\$25.00	\$283.00	2019-05-23
11	PC002	Lawton Ray Xavier	DC Comics - The Killing Joke	1	\$25.00	\$25.75	2018-02-21
12	PC003	Lawton Ray Xavier	Marvel - Iron Man	10	\$25.00	\$270.00	2018-02-26
13	PC004	Lawton Ray Xavier	Marvel - Iron Man	2	\$25.00	\$54.00	2018-05-12
14	PC004	Lawton Ray Xavier	Naruto - Itachi Uchiha	3	\$30.00	\$97.20	2018-05-12
15	PC006	Lawton Ray Xavier	Monopoly Voice Banking Electronic Family - Monop...	1	\$22.00	\$23.10	2018-06-01
16	PC006	Lawton Ray Xavier	DC Comics - The Killing Joke	18	\$25.00	\$472.50	2018-06-01
17	PC017	Lawton Ray Xavier	DC Comics - The Killing Joke	10	\$25.00	\$270.00	2018-07-19

Query executed successfully.

ist-s-students.syr.edu (12...AD\lwanyue (80)IST659_M004_xwanyue00:00:00100 rows

Access Report

User Shopping List						
UserName	PurchaseID	ProductName	Quantity	PricePerProduct	Expenditure	PurchaseDate
Anakin Skywalker						
	PC055	Naruto - Itachi Uchiha	8	\$30.00	\$263.40	2019/06/10
	PC055	Marvel - Iron Man	12	\$25.00	\$327.00	2019/06/10
	PC058	Assassins Creed - Assassins Cree	2	\$88.00	\$192.08	2019/07/08
	PC060	Assassins Creed - Assassins Cree	10	\$88.00	\$980.00	2019/07/26
	PC060	Toy Story Puzzle - Buzz Lightye	3	\$12.00	\$51.60	2019/07/26
	PC062	DC Comics - The Killing Joke	1	\$25.00	\$33.00	2019/08/02
	PC062	Naruto - Itachi Uchiha	5	\$30.00	\$168.00	2019/08/02
	PC062	Monopoly Voice Banking Elect	2	\$22.00	\$53.52	2019/08/02
Lawton Ray Xavier						
	PC002	DC Comics - The Killing Joke	1	\$25.00	\$25.75	2018/02/21
	PC003	Marvel - Iron Man	10	\$25.00	\$270.00	2018/02/26
	PC004	Naruto - Itachi Uchiha	3	\$30.00	\$97.20	2018/05/12
	PC004	Marvel - Iron Man	2	\$25.00	\$54.00	2018/05/12
	PC006	DC Comics - The Killing Joke	18	\$25.00	\$472.50	2018/06/01
	PC006	Monopoly Voice Banking Elect	1	\$22.00	\$23.10	2018/06/01
	PC017	DC Comics - The Killing Joke	10	\$25.00	\$270.00	2018/07/19
	PC021	Monopoly Voice Banking Elect	2	\$22.00	\$47.52	2018/08/22
	PC021	Wooden Puzzles - Wooden Puz	3	\$38.00	\$123.12	2018/08/22
	PC021	Assassins Creed - Assassins Cree	2	\$88.00	\$190.08	2018/08/22
	PC026	Toy Story Puzzle - Buzz Lightye	2	\$12.00	\$25.92	2018/11/11
	PC031	Marvel - Iron Man	2	\$25.00	\$54.00	2018/12/21
	PC034	Monopoly Voice Banking Elect	1	\$22.00	\$24.20	2019/01/06

Saturday, November 30, 2019 Page 1 of 5

4. **Product Sales Investigation:** What are the different products that are available in the forum database, including product category, the current inventory level, total number of products sold previously, and the existing vendors who supplied that product?

Access Query

```
SELECT
DISTINCT p.productID,
p.inventory AS "Current Inventory",
pc.productCategory AS "Product Category",
p.productName & ' - ' & p.title AS "Product Name",
pr.Previous_Sold_Quantity AS "Previous Sold Quantity",
v.vendorName AS "Vendor"
FROM dbo_Product p, dbo_Product_Category pc, (SELECT pl.productID, SUM(pl.pl_quantity)
AS Previous_Sold_Quantity FROM dbo_Product p, dbo_Purchase_Line pl WHERE p.productID
= pl.productID GROUP BY pl.productID) pr, dbo_Vendor v, dbo_Product_Supply ps
WHERE p.productCategoryID = pc.productCategoryID and p.productID = pr.productID and
p.productID = ps.productID and ps.vendorID = v.vendorID
ORDER BY p.productID;
```

Output of SQL Server

99 %						
Results Messages						
	productID	Current Inventory	Product Category	Product Name	Previous Sold Quantity	Vendor
1	P0001	3	Puzzle	Toy Story Puzzle - Buzz Lightyear	22	Anime Store
2	P0001	3	Puzzle	Toy Story Puzzle - Buzz Lightyear	22	Hasbro
3	P0002	10	Comic Book	DC Comics - The Killing Joke	132	DC
4	P0003	15	Model	Marvel - Iron Man	77	Marvel
5	P0005	21	Video Game	Assassins Creed - Assasins Creed 2	128	Ubisoft
6	P0006	10	Board Game	Monopoly Voice Banking Electronic Family - Monop...	64	Anime Store
7	P0007	6	Puzzle	Wooden Puzzles - Wooden Puzzles	73	Hasbro

Query executed successfully. | ist-s-students.syr.edu (12.... | Al

Access Report

Inventory Management Report					
ProductCategory	productID	CurrentInventory	ProductName	PreviousSold	Vendor
Board Game	P0006	10	Monopoly Voice Banking Electronic Fi	64	Anime Store
	P0002	10	DC Comics - The Killing Joke	132	DC
Model	P0003	15	Marvel - Iron Man	77	Marvel
	P0001	3	Toy Story Puzzle - Buzz Lightyear	22	Hasbro
Puzzle	P0001	3	Toy Story Puzzle - Buzz Lightyear	22	Anime Store
	P0007	6	Wooden Puzzles - Wooden Puzzles	73	Hasbro
Video Game	P0005	21	Assassins Creed - Assasins Creed 2	128	Ubisoft
	P0001	3	Toy Story Puzzle - Buzz Lightyear	22	Anime Store
Saturday, November 30, 2019					
Page 1 of 1					

5. **Best Selling Product in certain period:** What is the top selling product in the second half of 2018 along with the quantity sold of that product?

Access Query

```

SELECT
p.productName AS "Product Name",
pr.Total_Sold_Quantity as "Total Sold Quantity"
FROM dbo_product p, (SELECT TOP 1 pl.productID, SUM(pl.pl_quantity) AS
Total_Sold_Quantity FROM dbo_purchase_line pl
WHERE pl.purchaseID in (SELECT purchaseID FROM dbo_purchase WHERE
FORMAT(dateOfPurchase, 'yyyy') = 2018 and FORMAT(dateOfPurchase, 'MM') in
('07','08','09','10','11','12')) GROUP BY pl.productID ORDER BY SUM(pl.pl_quantity) DESC) pr
WHERE p.productID = pr.productID;

```

Output of SQL Server

99 %	Results	Messages
1	Product Name	Total Sold Quantity
1	DC Comics	38
Query executed successfully.		

Access Report

Best Selling Product Report

ProductName	TotalSoldQuantity
DC Comics	38

Saturday, November 30, 2019

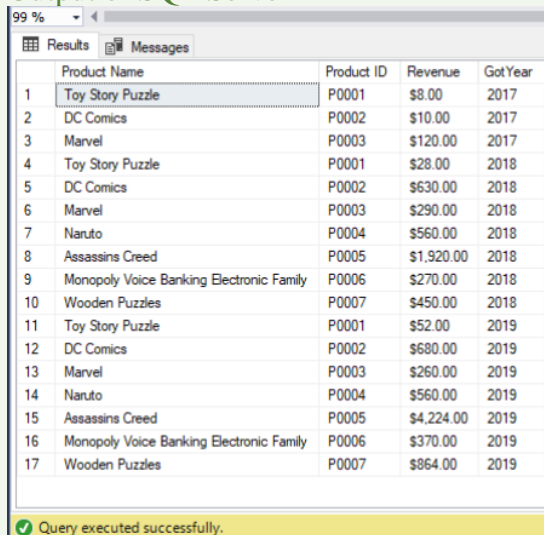
Page 1 of 1

6. **Benefit Management:** What is the total revenue of each product for each year?

Access Query

```
SELECT pr.productID,  
pr.productName,  
FORMAT((pq.Total_Sold_Quantity * pr.Revenue_of_Individual_Item), "Currency") AS "Total  
Revenue of Each Product"  
FROM  
(SELECT productID, productName, (salePrice - purchasePrice) AS Revenue_of_Individual_Item  
FROM dbo_Product) pr,  
(SELECT pl.productID, SUM(pl.pl_quantity) AS Total_Sold_Quantity FROM dbo_Purchase pc,  
dbo_Product p, dbo_Purchase_Line pl  
WHERE p.productID = pl.productID and pc.purchaseID = pl.purchaseID and  
FORMAT(pc.dateOfPurchase, 'yyyy') = 2019  
GROUP BY pl.productID) pq  
WHERE pr.productID = pq.productID;
```

Output of SQL Server



	Product Name	Product ID	Revenue	GotYear
1	Toy Story Puzzle	P0001	\$8.00	2017
2	DC Comics	P0002	\$10.00	2017
3	Marvel	P0003	\$120.00	2017
4	Toy Story Puzzle	P0001	\$28.00	2018
5	DC Comics	P0002	\$630.00	2018
6	Marvel	P0003	\$290.00	2018
7	Naruto	P0004	\$560.00	2018
8	Assassins Creed	P0005	\$1,920.00	2018
9	Monopoly Voice Banking Electronic Family	P0006	\$270.00	2018
10	Wooden Puzzles	P0007	\$450.00	2018
11	Toy Story Puzzle	P0001	\$52.00	2019
12	DC Comics	P0002	\$680.00	2019
13	Marvel	P0003	\$260.00	2019
14	Naruto	P0004	\$560.00	2019
15	Assassins Creed	P0005	\$4,224.00	2019
16	Monopoly Voice Banking Electronic Family	P0006	\$370.00	2019
17	Wooden Puzzles	P0007	\$864.00	2019

Query executed successfully.

Total Revenue of Product For Each Year

Year	ProductID	Product Name	Revenue
2017			
	P0001	Toy Story Puzzle	\$8.00
	P0002	DC Comics	\$10.00
	P0003	Marvel	\$120.00
2018			
	P0001	Toy Story Puzzle	\$28.00
	P0002	DC Comics	\$630.00
	P0003	Marvel	\$290.00
	P0004	Naruto	\$560.00
	P0005	Assassins Creed	\$1,920.00
	P0006	Monopoly Voice Banking Electronic Family	\$270.00
	P0007	Wooden Puzzles	\$450.00
2019			
	P0001	Toy Story Puzzle	\$52.00
	P0002	DC Comics	\$680.00
	P0003	Marvel	\$260.00
	P0004	Naruto	\$560.00
	P0005	Assassins Creed	\$4,224.00
	P0006	Monopoly Voice Banking Electronic Family	\$370.00
	P0007	Wooden Puzzles	\$864.00
Monday, December 2, 2019			
Page 1 of 1			

Interface Display

As mentioned above, there are two type of users which are the consumer group and administration group. To fulfill their disparate requirements, the user interface should also be separated into two parts with corresponding functionalities.

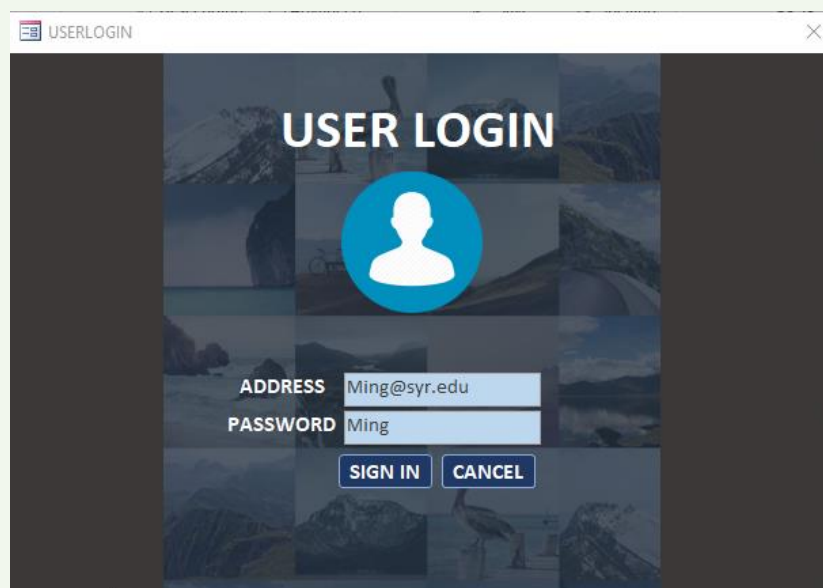
It is worthwhile to mention that the prototype shown in the paper can only being used to display the all-important functions. Hence, some functions, such as payment method verification, will be neglected.

¹⁰ For Consumer Group, the interfaces should include:

- **Sign Up** which allows new user to register an official account
- **User Information Management** which enables user to alter their private information, including their Name, Nick Name, Phone Number, Address.
- **Content Posting and Post Replying** empower user to create a post according to their needs and to comment on other users' posts. This interface also includes the rating function which encourages user to give each post or each reply a ranking score. Since that Toypedia is a web-based form which will use PHP and HTML5 to construct the forum, the rating functions will not be implemented in this prototype.
- **Shopping Cart** enables user to check the availability of product, to select their unique payment method and then make a purchase order.
- **Coupon Management** only provides an information page which displays all the coupons, including both the valid and invalid ones, that belong to the user.
- **Purchase History**, similarly, allows user to go through a list of purchase orders they made previously.

Login Page

Before official access the interfaces, each user needs to use the USER LOGIN page to validate their account authentication. The ADDRESS and PASSWORD entered by user will be used to compare with records that stored in UserEntity table. After successfully log into the system, the filter mechanism will display different navigation page to user according to their access right.



Sign Up Page

If the person does not have a resisted account, he or she needs to open SIGN UP page and fill the sign-up form. If the user misses some columns, the system will generate an error. Besides, if the system detects that the entered USERID has already existed in the UserEntity table, it will also throw out an error.

Before creating account, the user should create new address record as first. By clicking at the AddressID column, and then a window will pop up with the latest record of Address ID, such as A0025, stored in AddressEntity table. The next step is to enter an AddressID, such as A0026, to the subform and then complete click the submit button. Now the new address record has been created. Next, the user can start to create the account now. By clicking as the USER ID column, the user can get the latest USER ID. After putting the UserID, the AddressID which just created, and the rest columns, a user account can be created.

SIGN UP

USERID

FIRST NAME

MIDDLE

ADDRESSID

Microsoft Access

The Latest Address ID is A0026

OK

Set the Address First

addressID	streetNo	streetName	contactPh
A0026	New User	New User	New User
*	New User	New User	New User

Record: 1 of 1

No Filter

Search

SUBMIT CANCEL

In order to give user a concise interface, the system will automatically save the access right of the new user as “Normal User”. Similarly, the rest attributes that are relevant with AddressID will be automatically insert with “NEW USER”. The user can use the User Information Management Interface to change the values of Address.

SIGN UP

USERID: U0013

FIRST NAME: Test

MIDDLE: Test

LAST NAME: Test

NICK NAME: Test

EMAIL: Test

PASSWORD: Test

ADDRESSID: A0026

Set the Address First

addressID	streetNo	streetName	contactPh
*	New User	New User	New User

Record: 1 of 1 | No Filter | Search

SUBMIT **CANCEL**

Here is the result of UserEntity and AddressEntity. It is easy to find that account “U0013” and address “A0026” has been successfully recorded in the database.

dbo_UserEntity										
userID	addressID	fName	mName	lName	nName	email	hashedPassword	accountCreateDate	profilePicture	accessType
U0013	A0026	Test	Test	Test	Test	Test	Test	12/3/2019		2
*										

dbo_UserEntity								dbo_AddressEntity		
addressID	streetNo	streetName	contactPhon	city	stateName	country	zipcode			
A0026	New User	New User	New User	New User	New User	New User	New User			
*										

User Information Management

The User Information Management Interface enables user to change their personal information. Firstly, the user needs to use the filter to find the account. The next steps to alter the values that displayed in the rectangular boxes, such as the Name, Nick Name, and Address.

User Navigation Form

SIGN UP

USER INFORMATION

CREATE POST

CREATE REPLY

SHOPPING CART

COUPON MANAGEMENT

PURCHASE HISTORY

User Information Management

Personal Information

User ID

U0001

Account

Type Name

User

Address

StreetNum

1524

StateName

NY

USA

13210

Find and Replace

Find

Replace

Find What

U0013

Find Next

Cancel

Look In

Current field

Match

Whole Field

Search

All

Match Case

Search Fields As Formatted

User Navigation Form

SIGN UP

USER INFORMATION

CREATE POST

CREATE REPLY

SHOPPING CART

COUPON MANAGEMENT

PURCHASE HISTORY

User Information Management

Personal Information

User ID

First Name

Middle Name

Last Name

Nick Name

ContactPhone

U0013

Test

Test

Test

New User

Account Information

Type Name

Email Address

Account Password

User

Test@syr.edu

TestTest

Address Information

StreetNumber

StreetName

City

New User

New User

New User

StateName

Country

Zipcode

New User

New User


New User

Here is the result of alter the data.

userID	addressID	fName	mName	lName	nName	email	hashedPassword	accountCreateDate	profilePictui	accessType
U0012	A0017	Thomas		Shelby	Thomsa	tshelby@gmail	peakyblinder01	3/10/2012		2
U0013	A0026	It	is	Cold	Outside	snowing@syr.e	ChillyOutside	12/2/2019		2

Create Post


One of the elementary functionalities that a forum should possess is content posting and replying. Similar with the mechanism mentioned above, the user can click the PostID columns, and then the result will display automatically. The rest tasks are category selection, UserID entering, post title and post content entering. Once the user clicks the submit bottom, the post content will be updated to the Content_Post Entity.



User Navigation Form

[SIGN UP](#)
[USER INFORMATION](#)
[CREATE POST](#)
[CREATE REPLY](#)
[SHOPPING CART](#)
[COUPON MANAGEMENT](#)
[PURCHASE HISTORY](#)

Create Thread



Post Information

PostID	CategoryID	Check For Product Category
13		12

UserID	DateOfPost
U0013	12/2/2019

PostTitle

PostContent

1	Product Review
2	Recommendations
3	Product Selling
4	Product Buying

Here is the result of creating a post.

dbo_UserEntity	UserNavi	dbo_Content_Post								
postID	userID	categoryID	postTitle	dateOfPost	postContent	picture1	picture2	numberOfV	totalRatingS	ratingUserN
12	U0001	2	Lego Disney Els	2/3/2015	The Lego Disne			102	100	10
12	U0006	3	Looking for Job	10/21/2019	Still looking for			0	0	0
13	U0013	2	Testing	12/2/2019	Function Testir			0	0	

Post Reply

For content replying. A similar pattern will be executed. The use should use the filter button to locate a post by entering the PostID. Then the next step is to go through the detailed information of that post, including the title, content, the number of people who had clicked this post, the number of people who like or dislike the post. It is necessary to remind that the rating system is not included in this prototype even though the attributes that are used to store the records exists.

User Navigation Form

[SIGN UP](#)
[USER INFORMATION](#)
[CREATE POST](#)
[CONTENT REPLY](#)
[SHOPPING CART](#)
[COUPON MANAGEMENT](#)
[PURCHASE HISTORY](#)

Post Reply

Post Information

PostID	UserID	DateOfPost	NumberOfView	Likes	Dislikes
1					

Title
Lego Disney

Content
The Lego Dis scene where

Reply In

postID	ReplyID	UserID	ReplyContent	DateOfReply
* 1				12/2/2019

Find and Replace

Find Replace

Find What: 13

Find Next

Cancel

Look In: Current field

Match: Whole Field

Search: All

☐ Match Case
 ☒ Search Fields As Formatted

Here, the user should input his or her userID, and the content of reply. The value of PostID, ReplyID, and Date Of Reply will be fulfilled with appropriate vaues.

User Navigation Form

[SIGN UP](#)
[USER INFORMATION](#)
[CREATE POST](#)
[CONTENT REPLY](#)
[SHOPPING CART](#)
[COUPON MANAGEMENT](#)
[PURCHASE HISTORY](#)

Post Reply

Post Information

PostID	UserID	DateOfPost	NumberOfView	Likes	Dislikes
13	U0013	12/2/2019	0	0	0

Title
Testing

Content
Function Testing

Reply Information

postID	ReplyID	UserID	ReplyContent	DateOfReply
13	11	U0013	Test for replying	12/2/2019
* 13				12/2/2019

Here is the result of replying a post.

replyID	userID	postID	statusID	dateOfReply	content	pr_likes	pr_dislikes
10	U0001	10	2	10/1/2019	Why can t U find a new pl	12	0
11	U0013	13	1	12/2/2019	Test for replying	0	0

Shopping Cart

The product purchasing function is essential for the updated database. It is being expected to bring profits to the form administration group. The benefits will be used in system daily maintenance and employees' payroll. To create a purchase order, the user can click the purchasID, and then the system will give the latest Purchase ID record that stored in the database. Next step is to enter the UserID and select corresponding paymentID. Then the information that relates to that paymentID will be shown.

In the real scenario, more steps will be implemented in this function. For example, the user can enter a couponID to get a promotion. What's more, the verification of payment method will also be being required.

User Navigation Form

Shopping Cart

Purchase Details

PurchaseID: UserID: DateOfPurchase: 12/2/2019

Payment Information

PaymentID: CardNumber: ExpiryDate: CardKeeper:

Available Product

Shopping Cart

	purchaseID	productID	pl_quantity
*			

Record: 1 of 1 | No Filter | Search

Microsoft Access

The Latest Purchase ID is PC063

OK

User can click the drop-down button to check the available products.



User Navigation Form

Shopping Cart



Purchase Details

PurchaseID UserID DateOfPurchase

PC064 U0003 12/2/2019

Payment Information

PaymentID CardNumber ExpiryDate CardKeeper

PM003 5832817498221800 12/21 Sonamgyalpo Sherpa

Available Product

P0002				
P0001	Toy Story Puzzle	12	Buzz Lightyear	Standard
P0002	DC Comics	25	The Killing Joke	Black
P0003	Marvel	25	Iron Man	Red
P0004	Naruto	30	Itachi Uchiha	Black
P0005	Assasins Creed	88	Assasins Creed 2	White
P0006	Monopoly Voice Ba	22	Monopoly Voice Ba	Black
P0007	Wooden Puzzles	38	Wooden Puzzles	Standard

Record: 1 of 1 No Filter Search

In the Subform region, user should input the productID and the number of that product. According to the business rules, one purchase can contain several purchase lines orders. Each purchase line order can only contain one specific product with the quantity of that product.



User Navigation Form

Shopping Cart



Purchase Details

PurchaseID UserID DateOfPurchase

PC064 U0003 12/2/2019

Payment Information

PaymentID CardNumber ExpiryDate CardKeeper

PM003 5832817498221800 12/21 Sonamgyalpo Sherpa

Available Product

P0002

Shopping Cart

purchasel	productID	pl_quantity
PC064	P0002	3
PC064	P0003	1
* PC064		

Record: 2 of 2 No Filter Search


The two graphs below show the result of Purchas Entity and Purchase_Line Entity respectively.

purchaseID	paymentMe	userID	couponID	taxRate	deliveryFee	dateOfPurch
PC062	PM011	U0011	CP008	0.08	6	8/2/2019
PC063	PM008	U0008	CP008	0.08	9	8/3/2019
PC064	PM003	U0003	CP008	0.08	10	12/2/2019
*						

purchaseID	productID	pl_quantity
PC063	P0007	10
PC064	P0003	1
PC064	P0002	3
*		

Coupon Management


The Coupon Management Interface works as a coupon display report. It shows all the coupon that user possessed as well as the corresponding details, containing the coupon name, the discount rate, and coupon availability etc.



User Navigation Form

[SIGN UP](#)
[USER INFORMATION](#)
[CREATE POST](#)
[CONTENT REPLY](#)
[SHOPPING CART](#)
[COUPON MANAGEMENT](#)
[PURCHASE HISTORY](#)

Coupon Management



User Information

userID	UserName
U0012	yes
U0019	yes
U0019	yes

Find and Replace

Find

Replace

Find What: U0012

Find Next

Cancel

Look In: Current field


Match: Whole Field

Search: All

☐ Match Case
 ☒ Search Fields As Formatted

SIGN UP
USER INFORMATION
CREATE POST
CONTENT REPLY
SHOPPING CART
COUPON MANAGEMENT
PURCHASE HISTORY

Coupon Management




User Information

userID

U0012

UserName

Thomas Shelby




Coupon Information

CouponID	CouponName	DiscountRate	Quantity	OwnDate	CouponAvailability
CP006	EMPLOYEE	0.85	3	3/10/2012	yes
CP007	DAILY	0.95	2	1/1/2019	yes
CP008	NOCONPON	1	1	1/1/2019	yes
*					


Purchase History

The last interface is the Purchase History which shows all the previous purchase orders that made by the user. Also, the total number of expenditure that user spent on purchasing these orders can also be seen from the User Information section. Besides, the user can use the subform to check the delivery status of product. If the user find that the status is being marked as “Damaged” or “Lost”, he or she might need to contact the sales manager.


User Navigation Form

SIGN UP
USER INFORMATION
CREATE POST
CONTENT REPLY
SHOPPING CART
COUPON MANAGEMENT
PURCHASE HISTORY

Purchase History



User Information

UserID

U0011

User Name

Anakin Skywalker_A

Nick Name

Skywalker_A

User Type

User

Total Expenditure

\$2,068.60

Purchase History

purchasell	Name	Quantity	Price	Expenditur	PurchaseDa	DeliveryStatuse
PC055	Marvel - Iron Man	12	\$25.00	\$327.00	2019/06/10	Delivery completed
PC055	Naruto - Itachi Uchiha	8	\$30.00	\$263.40	2019/06/10	Delivery completed
PC058	Assassins Creed - Assasins C	2	\$88.00	\$192.08	2019/07/08	Delivery completed
PC060	Assassins Creed - Assasins C	10	\$88.00	\$980.00	2019/07/26	Delivery completed
PC060	Toy Story Puzzle - Buzz Ligh	3	\$12.00	\$51.60	2019/07/26	Delivery completed
PC062	DC Comics - The Killing Joke	1	\$25.00	\$33.00	2019/08/02	Delivery completed
PC062	Naruto - Itachi Uchiha	5	\$30.00	\$168.00	2019/08/02	Delivery completed
PC062	Monopoly Voice Banking El	2	\$22.00	\$53.52	2019/08/02	Delivery completed
*						

For Administration Group, the relevant interfaces are

- 14 **User Information Management** which enable employees to alter the personal information of the user.

- **Purchasing List** is similar with the Purchase History which provides purchase order made by each user. It focuses on the aspect of sales management.
- **Post Management interface** enables employee to search, view, alter, and delete the content posted by user.
- **Product Replenishment Management Interface** shows the information of product. It focusses on aspect of the inventory management.
- **Key Performance Indicator Interface** shows the total number of packages that handled by each employee
- **Shipping Management Interface** displays the delivery status of package, the employee who takes in charge of the package shipping, and courier who is responsible for package shipping
- **Revenue Management Interface** derives from the major question. It shows the total revenue of product that earned in different year

User Information Management

Unlike the user information management interface that includes in the Consumer Group, this interface enables employee to change all the elements that are relevant with user, even including the access type.

The screenshot displays the 'Admin Navigation Form' with a sidebar on the left containing icons for user management, a key, and address management. The main content area is divided into two sections: 'Personal Information' and 'Address Management'.

Personal Information section includes a table with columns: User ID, First Name, Middle Name, and Last Name.

A 'Find and Replace' dialog box is open, showing the 'Find' tab. It contains the following fields and options:

- Find What: U0013
- Look In: Current field
- Match: Whole Field
- Search: All
- ☐ Match Case
- ☒ Search Fields As Formatted

Address Management section includes a table with columns: Address ID, Current Address ID, Contact Phone, Street Number, Street Name, City, State Name, Country, and Zipcode.

Address ID	Current Address ID	Contact Phone	Street Number	Street Name	City	State Name	Country	Zipcode
A0026	A0026	123-4567-890	123	Snowing Street	Syracuse	NY	USA	13210

All the information is displayed in rectangular boxes, which means that all these elements are editable.

Admin Navigation Form

Personal Information

User ID	First Name	Middle Name	Last Name
U0013	It	is	Cold

Account Information

Email Address	Nick Name	Account Password	Account Create Dat
snowing@syr.edu	Outside	ChillyOutside	12/2/2019

Access Management

Access Type	Type Name	Description
2	User	User who does not belong to VIP U

Select Access Type
2

Function Buttons
Save Change

Address Management

Address ID	Current Address ID	Contact Phone
A0026	A0026	123-4567-890

Street Number	Street Name	City	State Name	Country	Zipcode
123	Snowing Street	Syracuse	NY	USA	13210

Here the example is to change the access type of User “U0013” from Normal User to VIP User.

Admin Navigation Form

Personal Information

User ID	First Name	Middle Name	Last Name
U0013	It	is	Cold

Account Information

Email Address	Nick Name	Account Password	Account Create Dat
snowing@syr.edu	Outside	ChillyOutside	12/2/2019

Access Management

Access Type	Type Name	Description
2	User	User who does not belong to VIP U

Select Access Type
2

Function Buttons
Save Change

Address Management

Address ID	Current Address ID	Contact Phone
A0026	A0026	123-4567-890

Street Number	Street Name	City	State Name	Country	Zipcode
123	Snowing Street	Syracuse	NY	USA	13210

This is the result of changing the access type.

dbo_UserEntity

AdminNavi

userID

addressID

fName

mName

lName

nName

email

hashedPassword

accountCreateDate

profilePictui

accessType

U0012	A0017	Thomas		Shelby	Thomsa	tshelby@gmail	peakyblinder01	3/10/2012	2
U0013	A0026	It	is	Cold	Outside	snowing@syr.	ChillyOutside	12/2/2019	3
*									

Purchasing List


The purpose of creating this interface is to help sales manager to understand the loyalty level and preferences of each user. By doing that, the sales manager can make a targeted customer list and then, is possible, give appropriate coupon to those customers.

Therefore, it is easy to find that the user personal information and the purchase information from this form. A lists of purchase orders are shown in the subform. The total expenditure shown below is the result contains tax and delivery fee.


Admin Navigation Form



[User Information Management](#)
[Use Shopping List](#)
[Post Management](#)
[Product Replenishment](#)
[KPI Index](#)
[Shipping Management](#)
[Benefit Management](#)


User Purchase List



UserID	UserName	Email
U0012	Thomas Shelby	tshelby@gmail.com
Nick Name	Type Name	Contact Phone
Thomsa	User	600-781-4813
		accountCreateDate
		3/10/2012



AddressID	Street Number	Street Name	City	Last Record	Next Record
A0017	413	Salt Spring Rd	Boston		
State Name	Country	Zipcode			
MA	USA	14591			



Shopping List

PurchaseID	ProductName	Quantity	Price	TotalExpenditure	Purch
PC053	Marvel - Iron Man	1	\$25.00	\$36.00	2019/05/29
PC054	Wooden Puzzles - Wooden Puzzles	6	\$38.00	\$252.24	2019/06/02

Content Post Management

Even though the administration team encourages user to publish post as many as possible, in order to maintain the forum discussion environment, it is still necessary to check the quality and the content of those posts. In this interface, the administrator can carefully exam the content of the posts and then decide the status level each post belongs to. For example, if the manager finds that user “U0010” just post a topic which recommends some products to other users, he then can change the status category from “Waiting for Approval” to “Approved”. If the manager reckons this post might cause some issues, then the manager can use the delete button to delete that record.

Admin Navigation Form

User Information Management
Use Shopping List
Post Management
Product Replenishment
KPI Index
Shipping Management
Benefit Management

Post Management

Author Information

UserID

U0010

UserName

Kennedy Kaushik

NickName

Kaushik_K

Post Information

✓

🗑️

📄

Po	PostDat	Ci	categoryNa	Title	content	S	statusDescri	Number	
7	7/13/2018	2	Recommendations	The Best Lego Sets for Kids	My daughter so far really likes the Friends	2	Approved	102	15
<div>⋮</div>									

Admin Navigation Form

User Information Management
Use Shopping List
Post Management
Product Replenishment
KPI Index
Shipping Management
Benefit Management

Post Management

Author Information

UserID

U0010

UserName

Kennedy Kaushik

NickName

Kaushik_K

Post Information

✓

🗑️

📄

Po	PostDat	Ci	categoryNa	Title	content	S	statusDescri	Number	Li
⋮									

Microsoft Access

⚠️

You are about to delete 1 record(s).

If you click Yes, you won't be able to undo this Delete operation. Are you sure you want to delete these records?

Yes

No

Product Replenishment Management

The product replenishment management interface, unlike others, contains two subforms. The upper one records inventory information of product, including the current inventory level, the number of products sold previously, and the vendor who supplied the product. The below one shows the transactions history about the products that appear in the upper section. The records are grouped by product category. Then the manager can click the button and go through all supply relationship between products the vendors as well as the trading records.

Admin Navigation Form

User Information Management Use Shopping List Post Management Product Replenishment KPI Index Shipping Management Benefit Management

Product Replenishment Management

Product Category



ProductCategoryID CategoryName Description
2 Puzzle



Product Information



productID	Inventor	Name	PreviousS	Vendor
P0001	3	Toy Story Puzzle - Buzz Lightyea	21	Anime Store
P0001	3	Toy Story Puzzle - Buzz Lightyea	21	Hasbro
P0007	6	Wooden Puzzles - Wooden Puzz	73	Hasbro
*				

Record: 1 of 3 No Filter Search

Replenishment Information



Vendor	ProductID	ProductName	ReplenishQu	SupplyStartDa
Hasbro	P0001	Toy Story Puzzle - Buzz Lightyea	20	5/12/201
Anime Store	P0001	Toy Story Puzzle - Buzz Lightyea	20	5/12/201
Hasbro	P0001	Toy Story Puzzle - Buzz Lightyea	15	5/17/201

Admin Navigation Form

User Information Management Use Shopping List Post Management Product Replenishment KPI Index Shipping Management Benefit Management

Product Replenishment Management

Product Information



productID	Inventor	Name	PreviousS	Vendor
P0001	3	Toy Story Puzzle - Buzz Lightyea	21	Anime Store
P0001	3	Toy Story Puzzle - Buzz Lightyea	21	Hasbro
P0007	6	Wooden Puzzles - Wooden Puzz	73	Hasbro
*				

Record: 1 of 3 No Filter Search

Replenishment Information



Vendor	ProductID	ProductName	ReplenishC	SupplyStartDa	SupplyEndDat
Hasbro	P0007	Wooden Puzzles -	12	11/17/2019	
Hasbro	P0001	Toy Story Puzzle -	25	11/17/2019	
Anime Store	P0001	Toy Story Puzzle -	25	11/17/2019	
Hasbro	P0001	Toy Story Puzzle -	15	5/17/2019	5/22/2019
Anime Store	P0001	Toy Story Puzzle -	15	5/17/2019	5/22/2019
Hasbro	P0001	Toy Story Puzzle -	20	5/17/2019	5/22/2019

Record: 1 of 7 No Filter Search

Key Performance Indicator

By selecting the EmployeeID, the name of employee will be displayed below. Correspondingly, the total number of packages that handled by that employee will be calculated and displayed. Therefore, this indicator could be identified as a key element in the employee performance management.


Here at the right corner, there are two buttons which are “Data Refresh” and “Form Printing”. If the manager wishes to print this form, he or she only needs to find the employee and then print the output by clicking the printing button.

Admin Navigation Form

User Information Management
Use Shopping List
Post Management
Product Replenishment
KPI Index
Shipping Management
Benefit Management

16

Key Performance Indicator for Employee



Select EmployeeID

E0001

E0002

E0003

E0004

E0005

E0006


Employee ID Employee Name
E0001 Anakin Skywalker
E0003 Thomas Shelby

Number of purchase order that handled by employee
count
23

Admin Navigation Form

User Information Management
Use Shopping List
Post Management
Product Replenishment
KPI Index
Shipping Management
Benefit Management

Key Performance Indicator for Employee



Select EmployeeID

E0003

Employee ID Employee Name
E0003 Thomas Shelby

Number of purchase order that handled by employee
count
5

Package Shipping Management

Once the purchase order was confirmed by the system, the employee should view the order list and then check the availability of product, pick up that product from warehouse, wrap the product, contact with courier, and then prepare to ship the product. All these processes will be recorded and updated to the database. If the manager or employee wish to check the shipping status of each purchase order, he or she can select the drop-down button, find the purchaseID, and check the delivery status. Besides, the information of courier who is responsible for shipping, the start date of shipping, and the end date which indicates the date on which user receives the package.

Therefore, this interface not only shows the delivery status of each package, but also reflects the performance of the courier in term of efficiency.

User Information Management

Use Shopping List

Post Management

Product Replenishment

KPI

Shipping Investigation

Purchase ID

PC001

PC002

PC003

PC004

PC005

PC006

PC007

PC008

PC009

PC010

PC011

PC012

PC013

PC014

PC015

PC016

Who takes incharge of this Purchase Order

Enter

5

eted

Admin Navigation Form

User Information Management Use Shopping List Post Management Product Replenishment KPI

Shipping Investigation



Purchase ID

PC003

Employee who takes incharge of this Purchase Order

Mudit Bisht

Courier

DTDC

StartDate

2018-02-26

EndDate

DeliveryStatus

Lost



Admin Navigation Form

User Information Management Use Shopping List Post Management Product Replenishment KPI

Shipping Investigation



Purchase ID

PC006

Employee who takes incharge of this Purchase Order

Anakin Skywalker

Courier

UPS

StartDate

2018-06-01

EndDate

2018-06-07

DeliveryStatus

Delivery completed




Benefit Management

The last interface in the administration group is benefit management interface which shows the revenue earned by each product in different year. Even though it derives from the sixth major data question, it is still necessary to be added into the administration interface group. This form is concise and informative. It shows the detailed information of each product and the profits brought by selling this product. If the managers can combine the result of this interface with that of product replenishment interface, they can set marketing strategy and manipulate the inventory level.

Admin Navigation Form



User Information Management Use Shopping List Post Management Product Replenishment KPI Index Shipping Management Benefit Management



Revenue Investigation For Each Year

Product Information

Name	Title	Details
Toy Story Puzzle	Buzz Lightyear	A puzzle about Buzz
Color	Weight	Size
Standard	1.37 pounds	11.7 x 9 x 1.9




Benefit For Each Year

GotYear	productID	Revenue
2017	P0001	\$8.00
2018	P0001	\$28.00
2019	P0001	\$52.00
*	P0001	

Record: 1 of 3 No Filter Search

Admin Navigation Form



User Information Management Use Shopping List Post Management Product Replenishment KPI Index Shipping Management Benefit Management



Revenue Investigation For Each Year

Product Information

Name	Title	Details
Assassins Creed	Assasins Creed 2	Origins
Color	Weight	Size
White	3.2 ounces	0.5 x 5.3 x 6.9



Benefit For Each Year

GotYear	productID	Revenue
2018	P0005	\$1,920.00
2019	P0005	\$4,224.00
*	P0005	

Record: 1 of 2 No Filter Search

Trigger

- 17 For every new user, the user will get a 5% discount coupon and for this trigger have been implemented. With this trigger, the administrator only needs to create a user. Then the trigger will automatically add a couponID and UserID in the User_Own_Coupon table. The trigger is executed when an insert command is done on the UserEntity table.

SQL Code

```
create trigger InstUser
on UserEntity
after insert as
IF @@ROWCOUNT = 1
begin
    declare @UID varchar(5)
    declare @userDate datetime

    select @UID = userID, @userDate = accountCreateDate from inserted

    insert into User_Own_Coupon
    values( 'CP007', @UID, 1, @userDate, 'yes' )
end
go
```

Execution

Before entering a new user, we can see that there are only 26 records in the User_Own_Coupon table.

Results		Messages	
		(No column name)	
1	26		

SQL query for inserting a user

```
insert into UserEntity
values( 'U0013', 'A0026', 'Hemant', 'Kamal', 'Sharma', 'HemSha', 'hsharma@syr.edu', 'asdxzczv', default, NULL, '2' )

select count(*) from User_Own_Coupon
```

Output

Results		Messages	
		(No column name)	
1	27		

Verification of the User_Own_Coupon Table

Results Messages					
	couponID	userID	quantity	ownDate	couponAvailable
8	CP006	U0011	5	2008-02-12 00:00:00.000	yes
9	CP006	U0012	3	2012-03-10 00:00:00.000	yes
10	CP007	U0001	1	2019-01-01 00:00:00.000	yes
11	CP007	U0003	1	2019-01-01 00:00:00.000	yes
12	CP007	U0008	7	2019-01-01 00:00:00.000	yes
13	CP007	U0011	5	2019-01-01 00:00:00.000	yes
14	CP007	U0012	2	2019-01-01 00:00:00.000	yes
15	CP007	U0013	1	2019-12-03 12:32:46.730	yes
16	CP008	U0001	1	2019-01-01 00:00:00.000	yes
17	CP008	U0002	1	2019-01-01 00:00:00.000	yes
18	CP008	U0003	1	2019-01-01 00:00:00.000	yes
19	CP008	U0004	1	2019-01-01 00:00:00.000	yes

Query executed successfully.

ist-s-students.syr.edu (12.... | AD\rmenonva (135)

Accordingly, it can be seen after the insertion of the new user with userID U0013, there has also been an insertion in the User_Own_Coupon table for the couponID CP007 (which is the coupon that gives a 5% discount) with userID U0013. Hence, the trigger was successfully executed.

Comment Summary

Page 1

1. I think you have a typo.....isn't it Toypedia?
i. -.25

2. Where are your names? That was an explicit requirement for the title page.

Page 8

3. I am a little confused about product supply, does this represent a purchase from the vendor? Do they have agreements with the vendor?

Page 11

4. Amazing job with your data dictionary.
I am not sure if i saw any domain constraints.

Page 12

5. Great job with your ERD, although it is a little difficult to see what is required and what is not required.

Page 13

6. This is fine.
7. At this point they are no longer entities.....at this point it they are tables...

Page 37

8. Great job with your create and insert statements. No comments really.
I am very impressed.

Page 46

9. I asked for an explanation of why each of these data questions is important.

-.25

Page 47

10. Great layout of activity views and user flow.

Page 49

11. Thanks for the proof here, it is good to know that the functionality is working.

Page 51

12. Great use of a drop down.
13. I like your navigation incorporated at the top.

Page 56

14. A matrix might have beneficial to represent this in one table for both user and admins.

Page 60

15. Great job with the forms guys.
I can tell the great amount of effort that you guys put into this.

Page 62

16. Great idea incorporating KPIs for employees.

Page 66

17. Great implementation of a trigger,
18. great job with the trigger implementation.