-- ASHMI BIDRUPANE SURESH

-- SQL DDL FOR CREATING TABLES, CONSTRAINTS AND STORED PROCEDURES

-- CREATING TABLES & CONSTRAINTS

--INSERTION OF VALUES

CREATE TABLE seller(

Seller\_ID Decimal(10) NOT NULL,

Seller\_Name VarChar(255),

Seller\_Phone VarChar(30),

Seller\_Email Varchar(50),

Seller\_Address\_Street VarChar(255),

Seller\_Address\_Building VarChar(255),

Seller\_Address\_City VarChar(255),

Seller\_Address\_State VarChar(255),

Seller\_Address\_Zipcode Decimal(5),

PRIMARY KEY (Seller\_ID));

INSERT INTO SELLER(Seller\_ID, Seller\_Name, Seller\_Phone, Seller\_Email, Seller\_Address\_Street, Seller\_Address\_Building, Seller\_Address\_City, Seller\_Address\_State, Seller\_Address\_Zipcode)

VALUES(1,'SONY',989988678,'SONY@GMAIL.COM','ABC','ABCD','WORCESTER','MA',01606);

INSERT INTO SELLER(Seller\_ID, Seller\_Name, Seller\_Phone, Seller\_Email, Seller\_Address\_Street, Seller\_Address\_Building, Seller\_Address\_City, Seller\_Address\_State, Seller\_Address\_Zipcode)

VALUES(2,'LENOVA',989999678,'LENOVA@GMAIL.COM','BCD','BCDE','WORCESTER','MA',01603);

INSERT INTO SELLER(Seller\_ID, Seller\_Name, Seller\_Phone, Seller\_Email, Seller\_Address\_Street, Seller\_Address\_Building, Seller\_Address\_City, Seller\_Address\_State, Seller\_Address\_Zipcode)

VALUES(3,'LG',989999679,'LG@GMAIL.COM','CDE','CDEF','WORCESTER','MA',01604);

INSERT INTO SELLER(Seller\_ID, Seller\_Name, Seller\_Phone, Seller\_Email, Seller\_Address\_Street, Seller\_Address\_Building, Seller\_Address\_City, Seller\_Address\_State, Seller\_Address\_Zipcode)

VALUES(4,'NIKON',9899998787,'NIKON@GMAIL.COM','DEF','DEFG','WORCESTER','MA',01603);

INSERT INTO SELLER(Seller\_ID, Seller\_Name, Seller\_Phone, Seller\_Email, Seller\_Address\_Street, Seller\_Address\_Building, Seller\_Address\_City, Seller\_Address\_State, Seller\_Address\_Zipcode)

VALUES(5,'CANON',9896996789,'CANON@GMAIL.COM','EFG','EFGH','WORCESTER','MA',01605);

CREATE TABLE Price\_Category(

Price\_Category\_ID Decimal(10),

Category\_Price Decimal(5),

PRIMARY KEY (Price\_Category\_ID));

INSERT INTO Price\_Category (Price\_Category\_ID,Category\_Price)

VALUES(1,50);

INSERT INTO Price\_Category (Price\_Category\_ID,Category\_Price)

VALUES(2,100);

INSERT INTO Price\_Category (Price\_Category\_ID,Category\_Price)

VALUES(3,150);

INSERT INTO Price\_Category (Price\_Category\_ID,Category\_Price)

VALUES(4,200);

INSERT INTO Price\_Category (Price\_Category\_ID,Category\_Price)

VALUES(5,250);

CREATE TABLE Product\_Category(

Product\_Category\_ID Decimal(10),

Category\_Name VarChar(255),

PRIMARY KEY (Product\_Category\_ID));

INSERT INTO Product\_Category (Product\_Category\_ID, Category\_Name)

VALUES(1,'LAPTOP');

INSERT INTO Product\_Category (Product\_Category\_ID, Category\_Name)

VALUES(2,'MONITOR');

INSERT INTO Product\_Category (Product\_Category\_ID, Category\_Name)

VALUES(3,'CAMERA');

INSERT INTO Product\_Category (Product\_Category\_ID, Category\_Name)

VALUES(4,'KEYBOARD');

INSERT INTO Product\_Category (Product\_Category\_ID, Category\_Name)

VALUES(5,'PHONE');

CREATE TABLE Products(

Product\_ID Decimal(10),

Seller\_ID Decimal(10),

Product\_Name VarChar(255),

Product\_Description VarChar(255),

Product\_Price Decimal(5, 2),

Product\_Category\_ID Decimal(10),

Price\_Category\_ID Decimal(10),

PRIMARY KEY (Product\_ID),

FOREIGN KEY (Seller\_ID) REFERENCES seller,

FOREIGN KEY (Product\_Category\_ID) REFERENCES Product\_Category,

FOREIGN KEY (Price\_Category\_ID) REFERENCES Price\_Category);

INSERT INTO Products(Product\_ID,Seller\_ID, Product\_Name, Product\_Description, Product\_Price, Product\_Category\_ID, Price\_Category\_ID)

VALUES(1,2,'LAPTOP','13 INCH LAPTOP',150,3,3);

INSERT INTO Products(Product\_ID,Seller\_ID, Product\_Name, Product\_Description, Product\_Price, Product\_Category\_ID, Price\_Category\_ID)

VALUES(2,4,'CAMERA','MIRRORLESS',59,3,2);

INSERT INTO Products(Product\_ID,Seller\_ID, Product\_Name, Product\_Description, Product\_Price, Product\_Category\_ID, Price\_Category\_ID)

VALUES(3,3,'PHONE','SMARTPHONE 5G',49,5,1);

INSERT INTO Products(Product\_ID,Seller\_ID, Product\_Name, Product\_Description, Product\_Price, Product\_Category\_ID, Price\_Category\_ID)

VALUES(6,4,'PHONE','BEST CAMERA',100,5,2);

CREATE TABLE RECEIPT\_SELLER(

RECEIPT\_ID DECIMAL(10),

PRODUCT\_ID DECIMAL(10),

PRODUCT\_QTY DECIMAL(10),

SELLER\_ID DECIMAL(10),

CONDITION VARCHAR(255),

PRIMARY KEY (RECEIPT\_ID),

FOREIGN KEY (PRODUCT\_ID) REFERENCES PRODUCTS,

FOREIGN KEY (SELLER\_ID) REFERENCES SELLER);

INSERT INTO RECEIPT\_SELLER (RECEIPT\_ID,PRODUCT\_ID,PRODUCT\_QTY,SELLER\_ID, CONDITION)

VALUES(1,1,10,2,'NEW');

INSERT INTO RECEIPT\_SELLER (RECEIPT\_ID,PRODUCT\_ID,PRODUCT\_QTY,SELLER\_ID, CONDITION)

VALUES(2,2,5,4,'RENEWED');

INSERT INTO RECEIPT\_SELLER (RECEIPT\_ID,PRODUCT\_ID,PRODUCT\_QTY,SELLER\_ID, CONDITION)

VALUES(3,3,4,3,'NEW');

INSERT INTO RECEIPT\_SELLER (RECEIPT\_ID,PRODUCT\_ID,PRODUCT\_QTY,SELLER\_ID, CONDITION)

VALUES(4,4,11,1,'OLD');

INSERT INTO RECEIPT\_SELLER (RECEIPT\_ID,PRODUCT\_ID,PRODUCT\_QTY,SELLER\_ID, CONDITION)

VALUES(5,5,7,5,'NEW');

CREATE TABLE CONSUMER (

CONSUMER\_ID DECIMAL(10),

CONSUMER\_FNAME VARCHAR(255),

CONSUMER\_LNAME VARCHAR(255),

CONSUMER\_PHONE VARCHAR(30),

CONSUMER\_EMAIL VARCHAR(50),

CONSUMER\_ADDRESS\_STREET VARCHAR(255),

CONSUMER\_ADDRESS\_BUILDING VARCHAR(255),

CONSUMER\_ADDRESS\_CITY VARCHAR(255),

CONSUMER\_ADDRESS\_STATE VARCHAR(255),

CONSUMER\_ADDRESS\_ZIPCODE DECIMAL(5),

PRIMARY KEY (CONSUMER\_ID));

INSERT INTO CONSUMER (CONSUMER\_ID,CONSUMER\_FNAME,CONSUMER\_LNAME,CONSUMER\_PHONE, CONSUMER\_EMAIL,CONSUMER\_ADDRESS\_STREET, CONSUMER\_ADDRESS\_BUILDING,CONSUMER\_ADDRESS\_CITY, CONSUMER\_ADDRESS\_STATE, CONSUMER\_ADDRESS\_ZIPCODE)

VALUES(1,'SHRESTA','MATHEW',9878766777,'SHRESTAMATHEW@GMAIL.COM','ABC','ABCD','WORCESTER','MA',01603);

INSERT INTO CONSUMER (CONSUMER\_ID,CONSUMER\_FNAME,CONSUMER\_LNAME,CONSUMER\_PHONE, CONSUMER\_EMAIL,CONSUMER\_ADDRESS\_STREET, CONSUMER\_ADDRESS\_BUILDING,CONSUMER\_ADDRESS\_CITY, CONSUMER\_ADDRESS\_STATE, CONSUMER\_ADDRESS\_ZIPCODE)

VALUES(2,'TONY','ADDAMS',9878768977,'TONYADDAMS@GMAIL.COM','BCD','BCDE','SHREWSBURY','MA',01607);

INSERT INTO CONSUMER (CONSUMER\_ID,CONSUMER\_FNAME,CONSUMER\_LNAME,CONSUMER\_PHONE, CONSUMER\_EMAIL,CONSUMER\_ADDRESS\_STREET, CONSUMER\_ADDRESS\_BUILDING,CONSUMER\_ADDRESS\_CITY, CONSUMER\_ADDRESS\_STATE, CONSUMER\_ADDRESS\_ZIPCODE)

VALUES(3,'PETER','GEORGE',9878768978,'PETERGEORGE@GMAIL.COM','CDE','CDEF','WORCESTER','MA',01607);

INSERT INTO CONSUMER (CONSUMER\_ID,CONSUMER\_FNAME,CONSUMER\_LNAME,CONSUMER\_PHONE, CONSUMER\_EMAIL,CONSUMER\_ADDRESS\_STREET, CONSUMER\_ADDRESS\_BUILDING,CONSUMER\_ADDRESS\_CITY, CONSUMER\_ADDRESS\_STATE, CONSUMER\_ADDRESS\_ZIPCODE)

VALUES(3,'GEORGE','MATHEW',9878548979,'GEORGEMATHEW@GMAIL.COM','DEF','DEFH','SHREWSBURY','MA',01607);

CREATE TABLE SELLER\_INVENTORY(

SELLER\_ID DECIMAL(10),

PRODUCT\_ID DECIMAL(10),

INVENTORY\_QTY DECIMAL(10),

FOREIGN KEY (SELLER\_ID) REFERENCES SELLER,

FOREIGN KEY (PRODUCT\_ID) REFERENCES PRODUCTS);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(1,1,20);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(2,2,11);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(3,3,13);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(4,4,10);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(5,5,6);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(1,6,16);

INSERT INTO SELLER\_INVENTORY (SELLER\_ID,PRODUCT\_ID, INVENTORY\_QTY)

VALUES(2,7,7);

CREATE TABLE SHIPPING\_CATEGORY(

SHIP\_CAT\_ID DECIMAL(10),

SHIP\_CAT\_TYPE VARCHAR(25),

PRIMARY KEY (SHIP\_CAT\_ID));

INSERT INTO SHIPPING\_CATEGORY (SHIP\_CAT\_ID, SHIP\_CAT\_TYPE)

VALUES(1,'SUPER-SAVING SHIPPING');

INSERT INTO SHIPPING\_CATEGORY (SHIP\_CAT\_ID, SHIP\_CAT\_TYPE)

VALUES(2,'STANDARD SHIPPING');

INSERT INTO SHIPPING\_CATEGORY (SHIP\_CAT\_ID, SHIP\_CAT\_TYPE)

VALUES(3,'TWO DAY SHIPPING');

INSERT INTO SHIPPING\_CATEGORY (SHIP\_CAT\_ID, SHIP\_CAT\_TYPE)

VALUES(4,'ONE DAY SHIPPING');

CREATE TABLE ORDERS(

ORDER\_ID DECIMAL(10),

PRODUCT\_ID DECIMAL(10),

SELLER\_ID DECIMAL(10),

ORDER\_QTY DECIMAL(10),

CONSUMER\_ID DECIMAL(10),

SHIP\_CAT\_ID DECIMAL(10),

PRIMARY KEY (ORDER\_ID),

FOREIGN KEY (PRODUCT\_ID) REFERENCES PRODUCTS,

FOREIGN KEY (SELLER\_ID) REFERENCES SELLER,

FOREIGN KEY (CONSUMER\_ID) REFERENCES CONSUMER,

FOREIGN KEY (SHIP\_CAT\_ID) REFERENCES SHIPPING\_CATEGORY);

INSERT INTO ORDERS (ORDER\_ID, PRODUCT\_ID, SELLER\_ID, ORDER\_QTY, CONSUMER\_ID, SHIP\_CAT\_ID)

VALUES(1,3,1,2,2,1);

INSERT INTO ORDERS (ORDER\_ID, PRODUCT\_ID, SELLER\_ID, ORDER\_QTY, CONSUMER\_ID, SHIP\_CAT\_ID)

VALUES(2,2,2,1,1,2);

INSERT INTO ORDERS (ORDER\_ID, PRODUCT\_ID, SELLER\_ID, ORDER\_QTY, CONSUMER\_ID, SHIP\_CAT\_ID)

VALUES(3,4,4,4,4,3);

INSERT INTO ORDERS (ORDER\_ID, PRODUCT\_ID, SELLER\_ID, ORDER\_QTY, CONSUMER\_ID, SHIP\_CAT\_ID)

VALUES(4,6,3,4,5,4);

CREATE TABLE SHIPMENT(

TRACKING\_ID DECIMAL(10),

ORDER\_ID DECIMAL(10),

CONSUMER\_ID DECIMAL(10),

PRODUCT\_ID DECIMAL(10),

PRIMARY KEY (TRACKING\_ID),

FOREIGN KEY (ORDER\_ID) REFERENCES ORDERS,

FOREIGN KEY (CONSUMER\_ID) REFERENCES CONSUMER,

FOREIGN KEY (PRODUCT\_ID) REFERENCES PRODUCTS);

INSERT INTO SHIPMENT (TRACKING\_ID,ORDER\_ID,CONSUMER\_ID,PRODUCT\_ID)

VALUES(1,1,2,3);

INSERT INTO SHIPMENT (TRACKING\_ID,ORDER\_ID,CONSUMER\_ID,PRODUCT\_ID)

VALUES(2,2,1,2);

INSERT INTO SHIPMENT (TRACKING\_ID,ORDER\_ID,CONSUMER\_ID,PRODUCT\_ID)

VALUES(3,3,4,4);

INSERT INTO SHIPMENT (TRACKING\_ID,ORDER\_ID,CONSUMER\_ID,PRODUCT\_ID)

VALUES(4,4,5,6);

CREATE TABLE ACCOUNT(

ACCOUNT\_ID DECIMAL(50),

ACCOUNT\_NAME VARCHAR(60),

ORDER\_ID DECIMAL(10),

CONSUMER\_ID DECIMAL(10),

CONSUMER\_PHONE VARCHAR(30),

CONSUMER\_EMAIL VARCHAR(50),

PRIMARY KEY (ACCOUNT\_ID),

FOREIGN KEY (ORDER\_ID) REFERENCES ORDERS,

FOREIGN KEY (CONSUMER\_ID) REFERENCES CONSUMER);

CREATE TABLE PAYEMENT(

PAYEMENT\_ID DECIMAL(50),

PAYEMENT\_MODE VARCHAR(60),

ORDER\_ID DECIMAL(10),

ACCOUNT\_ID DECIMAL(50),

PRIMARY KEY (PAYEMENT\_ID),

FOREIGN KEY (ORDER\_ID) REFERENCES ORDERS,

FOREIGN KEY (ACCOUNT\_ID) REFERENCES ACCOUNT);

--CREATING STORED PROCEDURES

CREATE OR REPLACE PROCEDURE ADD\_PRODUCTS(

Product\_ID\_arg IN DECIMAL,

Seller\_ID\_arg IN DECIMAL,

Product\_Name\_arg IN VARCHAR,

Product\_Description\_arg IN VARCHAR,

Product\_Price\_arg IN DECIMAL,

Product\_Category\_ID\_arg IN DECIMAL,

Price\_Category\_ID\_arg IN DECIMAL)

IS

BEGIN

INSERT INTO Products

(Product\_ID,Seller\_ID,Product\_Name,Product\_Description,Product\_Price,Product\_Category\_ID,Price\_Category\_ID)

VALUES(Product\_ID\_arg,Seller\_ID\_arg,Product\_Name\_arg,Product\_Description\_arg,Product\_Price\_arg,Product\_Category\_ID\_arg,Price\_Category\_ID\_arg);

END;

/

CREATE OR REPLACE PROCEDURE ADD\_RECEIPT\_SELLER(

RECEIPT\_ID\_arg IN DECIMAL,

PRODUCT\_ID\_arg IN DECIMAL,

PRODUCT\_QTY\_arg IN DECIMAL,

SELLER\_ID\_arg IN DECIMAL,

CONDITION\_arg IN VARCHAR)

IS

BEGIN

INSERT INTO RECEIPT\_SELLER

(RECEIPT\_ID,PRODUCT\_ID,PRODUCT\_QTY,SELLER\_ID,CONDITION)

VALUES(RECEIPT\_ID\_arg,PRODUCT\_ID\_arg,PRODUCT\_QTY\_arg,SELLER\_ID\_arg,CONDITION\_arg);

END;

/

CREATE OR REPLACE PROCEDURE ADD\_CONSUMER(

CONSUMER\_ID\_arg IN DECIMAL,

CONSUMER\_FNAME\_arg IN VARCHAR,

CONSUMER\_LNAME\_arg IN VARCHAR,

CONSUMER\_PHONE\_arg IN VARCHAR,

CONSUMER\_EMAIL\_arg IN VARCHAR,

CONSUMER\_ADDRESS\_STREET\_arg IN VARCHAR,

CONSUMER\_ADDRESS\_BUILDING\_arg IN VARCHAR,

CONSUMER\_ADDRESS\_CITY\_arg IN VARCHAR,

CONSUMER\_ADDRESS\_STATE\_arg IN VARCHAR,

CONSUMER\_ADDRESS\_ZIPCODE\_arg IN VARCHAR)

IS

BEGIN

INSERT INTO CONSUMER

(CONSUMER\_ID,CONSUMER\_FNAME,CONSUMER\_LNAME,CONSUMER\_PHONE,CONSUMER\_EMAIL,CONSUMER\_ADDRESS\_STREET,CONSUMER\_ADDRESS\_BUILDING,CONSUMER\_ADDRESS\_CITY,CONSUMER\_ADDRESS\_STATE,CONSUMER\_ADDRESS\_ZIPCODE)

VALUES(CONSUMER\_ID\_arg,CONSUMER\_FNAME\_arg,CONSUMER\_LNAME\_arg,CONSUMER\_PHONE\_arg,CONSUMER\_EMAIL\_arg,CONSUMER\_ADDRESS\_STREET\_arg,CONSUMER\_ADDRESS\_BUILDING\_arg,CONSUMER\_ADDRESS\_CITY\_arg,CONSUMER\_ADDRESS\_STATE\_arg,CONSUMER\_ADDRESS\_ZIPCODE\_arg);

END;

/

CREATE OR REPLACE PROCEDURE ADD\_ORDERS(

ORDER\_ID\_arg IN DECIMAL,

PRODUCT\_ID\_arg IN DECIMAL,

SELLER\_ID\_arg IN DECIMAL,

ORDER\_QTY\_arg IN DECIMAL,

CONSUMER\_ID\_arg IN DECIMAL,

SHIP\_CAT\_ID\_arg IN DECIMAL)

IS

BEGIN

INSERT INTO ORDERS

(ORDER\_ID,PRODUCT\_ID,SELLER\_ID,ORDER\_QTY,CONSUMER\_ID,SHIP\_CAT\_ID)

VALUES

(ORDER\_ID\_arg,PRODUCT\_ID\_arg,SELLER\_ID\_arg,ORDER\_QTY\_arg,CONSUMER\_ID\_arg,SHIP\_CAT\_ID\_arg);

END;

/

CREATE OR REPLACE PROCEDURE ADD\_SHIPMENT(

TRACKING\_ID\_arg IN DECIMAL,

ORDER\_ID\_arg IN DECIMAL,

CONSUMER\_ID\_arg IN DECIMAL,

PRODUCT\_ID\_arg in DECIMAL)

IS

BEGIN

INSERT INTO SHIPMENT

(TRACKING\_ID,ORDER\_ID,CONSUMER\_ID,PRODUCT\_ID)

VALUES

(TRACKING\_ID\_arg,ORDER\_ID\_arg,CONSUMER\_ID\_arg,PRODUCT\_ID\_arg);

END;

/

-- CREATED TRIGGER TO INSERT THE DATA IN SELLER\_INVENTORY WHENEVER THE DATA ADDS IN RECEIPT\_SELLER:-

CREATE OR REPLACE TRIGGER Seller\_Inventory\_Add\_Trg1

BEFORE INSERT ON RECEIPT\_SELLER

FOR EACH ROW

BEGIN

IF :OLD.PRODUCT\_QTY IS NULL THEN

INSERT INTO Seller\_Inventory (Seller\_ID, Product\_ID, Inventory\_Qty)

VALUES (:NEW.Seller\_ID, :NEW.Product\_ID, :NEW.PRODUCT\_QTY);

ELSE

INSERT INTO Seller\_Inventory (Seller\_ID, Product\_ID, Inventory\_Qty)

VALUES (:Old.Seller\_ID, :OLD.Product\_ID, :NEW.PRODUCT\_QTY);

END IF;

END;

-- CREATED TRIGGER TO UPDATE (DEDUCT) THE INVENTORY QTY IN THE SELLER\_INVENTORY WHENEVER THE PURCHASE HAPPENS:-

CREATE OR REPLACE TRIGGER Seller\_Inventory\_Orders\_Update\_Trg

BEFORE UPDATE OR INSERT ON ORDERS

FOR EACH ROW

BEGIN

UPDATE Seller\_Inventory

SET Inventory\_Qty = Inventory\_Qty - :NEW.Order\_Qty

WHERE Product\_ID = :NEW.Product\_ID and Seller\_ID = :NEW.Seller\_ID;

END;