

Name:- Aarya Varat Joshi

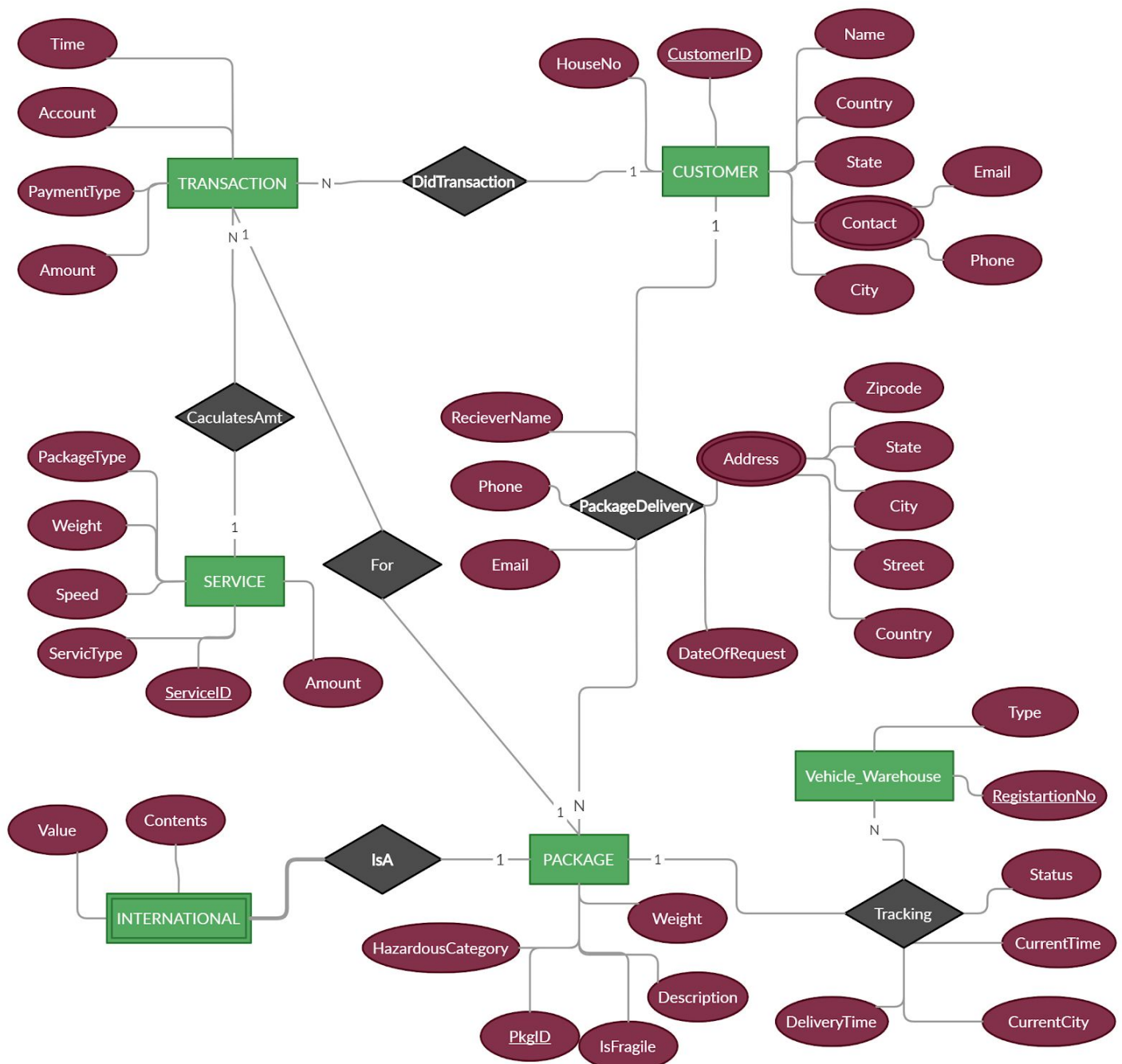
Roll No:- **1801CS01**

Date:- 30-11-2020

DBMS - CS355

## Mini Project Assignment - Package Delivery Company Database

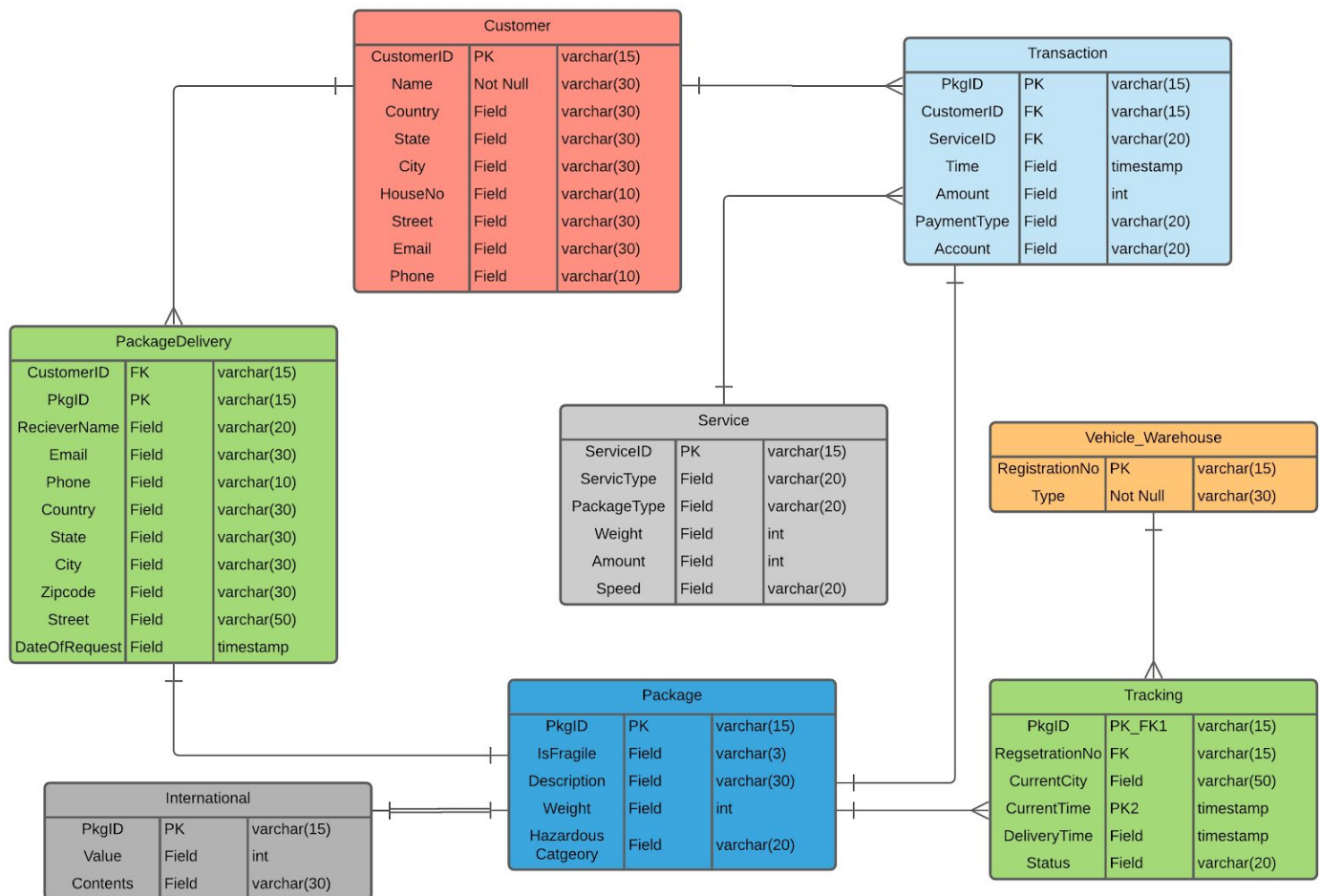
### ER(Entity Relationship) - Diagram



#### Description of ERD:

- 1-1, 1-N and N-N lines depicts One-to-One, One-to-Many and Many-to-Many respectively.
- **Green** Color Rectangles are Entities and Double Walled Rectangles are Weak Entities.
- **Grey** Color Diamonds are Rectangles.
- **Maroon** colored Circles are attributes and Double Walled Circles are Multivalued.

# Relational Model



## Description of Relational Model:

- **Green** colored tables are Relational Tables(PackageDelivery and Tracking)
- **Customer** to **PackageDelivery** is a **One-to-Many** relationship since one Customer can ship multiple Packages.
- **Customer** to **Transaction** is a **One-to-Many** relationship since one Customer can do multiple Transactions.
- **Package** to **PackageDelivery** is a **One-to-One** relationship since one Package can have only one Receiver.
- **Package** to **Tracking** is a **One-to-Many** relationship since one Package can have multiple Locations at different times.
- **Package** to **International** is a **Weak One-to-One** relationship.
- **Service** to **Transaction** is a **One-to-Many** relationship since the same service can be used by multiple transactions.
- **Vehicle\_Warehouse** to **Tracking** is **One-to-Many** relationship since the same vehicle or warehouse can have multiple packages.
- **Package** to **Transaction** is a **One-to-One** relationship since one package can have only one transaction.

## Description of Main Tables:

- **Customer** table contains all the details like address, phoneNo, etc of each Customer.
- **Service** table contains all the details like ServiceType (like Prepaid, Postpaid, etc) , PackageType, etc.
- **Transaction** table contains details of all transactions for each PkgID.
- **Package** contains description of Package, its categories etc.

## CREATE TABLE STATEMENTS

```
CREATE TABLE Customer(  
    CustomerID varchar(20),  
    Name varchar(30) NOT NULL,  
    Country varchar(30),  
    State varchar(30),  
    City varchar(30),  
    HouseNo varchar(30),  
    Street varchar(30),  
    Email varchar(30),  
    Phone varchar(10),  
    PRIMARY KEY(CustomerID));
```

```
CREATE TABLE Service(  
    ServiceID varchar(15),  
    ServiceType varchar(20),  
    PackageType varchar(20) NOT NULL,  
    Weight int,  
    Amount int,  
    Speed varchar(20),  
    PRIMARY KEY(ServiceID));
```

```
CREATE TABLE Package(  
    PkgID varchar(15),  
    IsFragile varchar(3),  
    Description varchar(30),  
    Weight int,  
    HazardousCategory varchar(20),  
    PRIMARY KEY(PkgID));
```

```
CREATE TABLE PackageDelivery(  
    CustomerID varchar(20),  
    PkgID varchar(20),  
    RecieverName varchar(20),  
    Email varchar(30),  
    Phone varchar(10),  
    Country varchar(30),  
    State varchar(30),  
    City varchar(30),  
    Zipcode varchar(30),  
    Street varchar(30),  
    DateOfRequest timestamp,  
    PRIMARY KEY(PkgID),  
    CONSTRAINT FK_PkgDel FOREIGN KEY(PkgID) REFERENCES Package(PkgID));
```

```
CREATE TABLE Transaction(  
    PkgID varchar(15),  
    CustomerID varchar(15),  
    ServiceID varchar(20),  
    Time timestamp,  
    Amount int,
```

```

    PaymentType varchar(20),
    Account varchar(20),
    FOREIGN KEY(PkgID) REFERENCES Package(PkgID),
    CONSTRAINT FK_Transaction1 FOREIGN KEY(CustomerID) REFERENCES
Customer(CustomerID),
    CONSTRAINT FK_Transaction2 FOREIGN KEY(ServiceID) REFERENCES
Service(ServiceID));

```

```

CREATE TABLE Vehicle_Warehouse(
    RegistrationNo varchar(15),
    Type varchar(30) NOT NULL,
    PRIMARY KEY(RegistrationNo));

```

```

CREATE TABLE Tracking(
    PkgID varchar(15),
    RegistrationNo varchar(15),
    CurrentCity varchar(50),
    CurrentTime timestamp,
    DeliveryTime timestamp,
    Status varchar(20),
    PRIMARY KEY(PkgID,CurrentTime),
    CONSTRAINT FK_Tracking1 FOREIGN KEY(PkgID) REFERENCES Package(PkgID),
    CONSTRAINT FK_Tracking2 FOREIGN KEY(RegistrationNo) REFERENCES
Vehicle_Warehouse(RegistrationNo));

```

```

CREATE TABLE International(
    PkgID varchar(15),
    Value int,
    Contents varchar(30),
    PRIMARY KEY(PkgID),
    CONSTRAINT FK_International FOREIGN KEY(PkgID) REFERENCES
Package(PkgID));

```

## **FILLING PROCEDURES**

```

DROP PROCEDURE procCustDummyData;
delimiter $$
CREATE PROCEDURE procCustDummyData(IN N int)
BEGIN
    DECLARE varCustomerID varchar(15);
    DECLARE varName varchar(30);
    DECLARE varCountry varchar(30);
    DECLARE varState varchar(30);
    DECLARE varCity varchar(30);
    DECLARE varHouseNo varchar(10);
    DECLARE varStreet varchar(30);
    DECLARE varEmail varchar(30);
    DECLARE varPhone varchar(10);
    DECLARE varAnum decimal(5);
    DECLARE varBalance int;
    DECLARE flag int;
    DECLARE c int;

```

```

SET c=0;

WHILE(c<N) DO
    SET varCustomerID = concat('cust-',c);
    SET varName = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
    SET varCountry = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
    SET varCity = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
    SET varState = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
    SET varHouseNo = LPAD(FLOOR(RAND() * 10000000000), 10, '0');
    SET varStreet = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
    SET varEmail = concat(substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(10)+1)), '@',
        substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(10)+1)));
    SET varPhone = LPAD(FLOOR(RAND() * 10000000000), 10, '0');
    INSERT INTO Customer

VALUES(varCustomerID,varName,varCountry,varState,varCity,varHouseNo,varStreet,varEm
ail,varPhone);
    SET c=c+1;
END WHILE;
END$$
delimiter ;

DELETE FROM Customer;
CALL procCustDummyData(100);
SELECT * FROM Customer;

DROP PROCEDURE procServiceDummyData;
delimiter $$
CREATE PROCEDURE procServiceDummyData(IN N int)
BEGIN
    DECLARE varServiceID varchar(15);
    DECLARE varServiceType varchar(20);
    DECLARE varPackageType varchar(20);
    DECLARE varWeight int;
    DECLARE varAmount int;
    DECLARE varSpeed varchar(20);
    DECLARE c int;
    DECLARE sp int;
    DECLARE fl int;

    SET c=0;

    WHILE(c<N) DO
        SET varServiceID = concat('serv-',c);
        SET fl = FLOOR(RAND()*(3))+1;

```

```

IF fl=1 THEN
    SET varPackageType="Flat Envelope";
ELSEIF fl=2 THEN
    SET varPackageType="Small box";
ELSE
    SET varPackageType="Larger Box";
END IF;
IF fl=1 THEN
    SET varServiceType="Prepaid";
ELSEIF fl=2 THEN
    SET varServiceType="Postpaid";
ELSE
    SET varServiceType="Normal";
END IF;
SET varWeight = FLOOR(RAND()*(99999))+1;
SET varAmount = FLOOR(RAND()*(999))+1;
SET sp = LPAD(FLOOR(RAND() * 10000000000), 1, '0');
SET varSpeed = concat(sp," Day Delivery");
INSERT INTO Service

```

```

VALUES(varServiceID,varServiceType,varPackageType,varWeight,varAmount,varSpeed);

```

```

    SET c=c+1;

```

```

END WHILE;

```

```

END$$

```

```

delimiter ;

```

```

CALL procServiceDummyData(100);

```

```

SELECT * FROM Service;

```

```

DROP PROCEDURE procPackage_INTDummyData;

```

```

delimiter $$

```

```

CREATE PROCEDURE procPackage_INTDummyData(IN N int)

```

```

BEGIN

```

```

    DECLARE PkgID varchar(15);

```

```

    DECLARE IsFragile varchar(3);

```

```

    DECLARE Description varchar(30);

```

```

    DECLARE varWeight int;

```

```

    DECLARE HazardousCategory varchar(20);

```

```

    DECLARE Value int;

```

```

    DECLARE Contents varchar(30);

```

```

    DECLARE c int;

```

```

    DECLARE fl int;

```

```

    SET c=0;

```

```

    WHILE(c<N) DO

```

```

        SET PkgID = concat('pkg-',c);

```

```

        SET fl = FLOOR(RAND()*(2))+1;

```

```

        IF fl=2 THEN

```

```

            SET IsFragile = "YES";

```

```

        ELSE

```

```

            SET IsFragile = "NO";

```

```

        END IF;

```

```

        SET Description = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(26)+1));
        SET varWeight = FLOOR(RAND()*(99999))+1;
        SET fl = FLOOR(RAND()*(2))+1;
        IF fl=2 THEN
            SET HazardousCategory = "NORMAL";
        ELSE
            SET HazardousCategory = "RADIOACTIVE";
        END IF;
        INSERT INTO Package
            VALUES(PkgID,IsFragile,Description,varWeight,HazardousCategory);
        SET fl = FLOOR(RAND()*(25))+1;
        IF fl=12 THEN
            SET Contents = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(26)+1));
            SET Value=FLOOR(RAND()*(99999))+1;
            INSERT INTO International
                VALUES(PkgID,Value,Contents);
        END IF;
        SET c=c+1;
    END WHILE;
END$$
delimiter ;
CALL procPackage_INTDummyData(100);
SELECT * FROM Package;
SELECT * FROM International;

```

```

DROP PROCEDURE procPackageDeliveryDummyData;
delimiter $$
CREATE PROCEDURE procPackageDeliveryDummyData(IN N int)
BEGIN
    DECLARE CustomerID varchar(15);
    DECLARE PkgID varchar(15);
    DECLARE varRName varchar(30);
    DECLARE varEmail varchar(30);
    DECLARE varPhone varchar(10);
    DECLARE varCountry varchar(30);
    DECLARE varState varchar(30);
    DECLARE varCity varchar(30);
    DECLARE varZipcode varchar(10);
    DECLARE varStreet varchar(30);
    DECLARE DateOfRequest timestamp;
    DECLARE c int;
    DECLARE x int;
    DECLARE sp int;

    SET c=0;

    WHILE(c<N) DO
        SET x=FLOOR(RAND()*(25))+1;
        SET CustomerID = concat('cust-',x);
        SET PkgID = concat('pkg-',c);
    
```

```

        SET varRName = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        SET varCountry = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        SET varCity = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        SET varState = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        SET varStreet = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        SET varEmail = concat(substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(10)+1)), '@',
        substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(10)+1)));
        SET varPhone = LPAD(FLOOR(RAND() * 10000000000), 10, '0');
        SET varZipcode = LPAD(FLOOR(RAND() * 10000000000), 6, '0');
        SELECT FROM_UNIXTIME(UNIX_TIMESTAMP('2019-01-01 14:53:27') +
FLOOR(0 + (RAND() * 63072000))) INTO DateOfRequest;
        INSERT INTO PackageDelivery

VALUES(CustomerID,PkgID,varRName,varEmail,varPhone,varCountry,varState,varCity,varZ
ipcode,varStreet,DateOfRequest);
        SET c=c+1;
    END WHILE;
END$$
delimiter ;
DELETE FROM PackageDelivery;
CALL procPackageDeliveryDummyData(100);
SELECT * FROM PackageDelivery;

DROP PROCEDURE procTransactionDummyData;
delimiter $$
CREATE PROCEDURE procTransactionDummyData(IN N int)
BEGIN
    DECLARE PkgID varchar(15);
    DECLARE CustomerID varchar(15);
    DECLARE ServiceID varchar(15);
    DECLARE time timestamp;
    DECLARE Amount int;
    DECLARE PaymentType varchar(20);
    DECLARE Account int;
    DECLARE c int;
    DECLARE sp int;
    DECLARE x int;

    SET c=0;

    WHILE(c<N) DO
        SET x = FLOOR(RAND()*(99))+1;
        SET CustomerID = concat('cust-',x);
        SET PkgID = concat('pkg-',c);
        SET ServiceID = concat('serv-',c);

```



```

        SELECT FROM_UNIXTIME(UNIX_TIMESTAMP('2019-01-01 14:53:27') +
FLOOR(0 + (RAND() * 63072000))) INTO time;
        SET Amount = FLOOR(RAND()*(999))+1;
        SET Account = LPAD(FLOOR(RAND() * 10000000000), 8, '0');
        SET PaymentType = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        INSERT INTO Transaction

```

```

VALUES(PkgID, CustomerID, ServiceID, Time, Amount, PaymentType, Account);
        SET c=c+1;
    END WHILE;

```

```

END$$

```

```

delimiter ;

```

```

DELETE FROM Transaction;

```

```

CALL procTransactionDummyData(100);

```

```

SELECT * FROM Transaction;

```

```

DROP PROCEDURE procVehicle_WarehouseDummyData;

```

```

delimiter $$

```

```

CREATE PROCEDURE procVehicle_WarehouseDummyData(IN N int)

```

```

BEGIN

```

```

    DECLARE RegistrationNo varchar(15);

```

```

    DECLARE Type varchar(30);

```

```

    DECLARE VWCondition varchar(20);

```

```

    DECLARE c int;

```

```

    DECLARE fl int;

```

```

    SET c=0;

```

```

    WHILE(c<N) DO

```

```

        SET RegistrationNo = concat('UK07-',c);

```

```

        SET fl = FLOOR(RAND()*(3))+1;

```

```

        IF fl=2 THEN

```

```

            SET Type = "Truck";

```

```

        ELSEIF fl=1 THEN

```

```

            SET Type = "Plane";

```

```

        ELSE

```

```

            SET Type = "Warehouse";

```

```

        END IF;

```

```

        INSERT INTO Vehicle_Warehouse

```

```

            VALUES(RegistrationNo,Type);

```

```

        SET c=c+1;

```

```

    END WHILE;

```

```

END$$

```

```

delimiter ;

```

```

DELETE FROM Vehicle_Warehouse;

```

```

CALL procVehicle_WarehouseDummyData(100);

```

```

SELECT * FROM Vehicle_Warehouse;

```

```

DROP PROCEDURE procTrackingDummyData;

```

```

delimiter $$

```

```

CREATE PROCEDURE procTrackingDummyData(IN N int)

```

```

BEGIN

```

```

DECLARE varPkgID varchar(15);
DECLARE RegistrationNo varchar(15);
DECLARE CurrentCity varchar(50);
DECLARE DeliveryTime timestamp;
DECLARE Status varchar(20);
DECLARE varStrtTime timestamp;
DECLARE varCurrentTime timestamp;
DECLARE c int;
DECLARE x int;
DECLARE fl int;
DECLARE cnt int;
DECLARE sn int;

SET c=0;
SET x=0;

WHILE(c<N*4) DO
    SET varPkgID = concat('pkg-',x);
    SET RegistrationNo = concat('UK07-',x);
    SET DeliveryTime=NULL;
    SET Status="Out For Delivery";
    SELECT DateOfRequest FROM PackageDelivery
    WHERE PkgID=varPkgID INTO varStrtTime;
    SET cnt=0;
    SET sn=FLOOR(RAND()*(5))+1;
    WHILE(cnt<sn) DO
        SET fl = FLOOR(RAND()*(50))+1;
        SET CurrentCity = substring('ABCDEFGHIJKLMNOPQRSTUVWXYZ',
FLOOR(RAND()*(26)+1), FLOOR(RAND()*(20)+1));
        SET RegistrationNo = concat('UK07-',fl);
        SELECT FROM_UNIXTIME(UNIX_TIMESTAMP(varStrtTime) + FLOOR(0
+ (RAND() * 172800))) INTO varCurrentTime;
        SET varStrtTime=varCurrentTime;
        SET DeliveryTime=FROM_UNIXTIME(UNIX_TIMESTAMP(varStrtTime) +
FLOOR(0 + (RAND() * 172800)));
        IF cnt=sn-1 THEN
            SET Status="Delivered";
            SET DeliveryTime=varStrtTime;
        END IF;
        INSERT INTO Tracking

VALUES(varPkgID,RegistrationNo,CurrentCity,varStrtTime,DeliveryTime,Status);
        SET cnt=cnt+1;
    END WHILE;
    SET x=x+1;
    SET c=c+4;
END WHILE;
END$$
delimiter ;
DELETE FROM Tracking;
CALL procTrackingDummyData(100);
SELECT * FROM Tracking;

```

## EXPORTING DATA INTO .CSV FILES

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/Customer.csv'  
fields terminated by ','  
lines terminated by '\n' FROM Customer;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/Service.csv'  
fields terminated by ','  
lines terminated by '\n' FROM Service;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/Package.csv'  
fields terminated by ','  
lines terminated by '\n' FROM Package;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server  
8.0/Uploads/PackageDelivery.csv'  
fields terminated by ','  
lines terminated by '\n' FROM PackageDelivery;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/Transaction.csv'  
fields terminated by ','  
lines terminated by '\n' FROM Transaction;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server  
8.0/Uploads/Vehicle_Warehouse.csv'  
fields terminated by ','  
lines terminated by '\n' FROM Vehicle_Warehouse;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server 8.0/Uploads/Tracking.csv'  
fields terminated by ','  
lines terminated by '\n' FROM Tracking;
```

```
SELECT * INTO outfile 'C:/ProgramData/MySQL/MySQL Server  
8.0/Uploads/International.csv'  
fields terminated by ','  
lines terminated by '\n' FROM International;
```

## QUERIES

**#A)**

**#1**

```
SELECT Name FROM Customer  
WHERE Customer.CustomerID IN(  
    SELECT CustomerID FROM PackageDelivery  
    WHERE PkgID IN(  
        SELECT DISTINCT PkgID  
        FROM Tracking  
        WHERE RegistrationNo="UK07-13"  
        AND Status = "Out For Delivery"  
    ));
```

**#2**

```
SELECT RecieverName FROM PackageDelivery
```

```
WHERE PkgID IN(  
    SELECT DISTINCT PkgID FROM Tracking  
    WHERE RegistrationNo="UK07-13"  
    AND Status = "Out For Delivery");
```

**#3**

```
SELECT PkgID FROM Tracking  
WHERE DeliveryTime=(SELECT MAX(DeliveryTime) FROM Tracking  
    WHERE RegistrationNo="UK07-13"  
    AND Status = "Delivered");
```

**#B)**

```
SELECT CustomerID FROM packagedelivery  
WHERE year(DateOfRequest)=2019 GROUP BY CustomerID ORDER BY count(*) DESC  
LIMIT 1;
```

**#C)**

```
SELECT CustomerID from Transaction  
WHERE year(time)=2019  
GROUP BY CustomerID  
ORDER BY sum(amount) desc LIMIT 1;
```

**#D)**

```
SELECT Street FROM Customer GROUP BY Street ORDER BY count(*) DESC LIMIT 1;
```

**#E)**

```
SELECT DISTINCT Transaction.PkgID  
FROM Transaction,Tracking,PackageDelivery,Service  
WHERE Transaction.PkgID=Tracking.PkgID  
AND Transaction.ServiceID=Service.ServiceID  
AND PackageDelivery.PkgID=Transaction.PkgID  
AND DATE_ADD(PackageDelivery.DateOfRequest,  
    interval substring(Service.speed,1,1) day)<Tracking.CurrentTime;
```

**#F)**

```
SELECT Customer.CustomerID, Customer.Name, Transaction.Amount, Customer.HouseNo,  
Customer.Street, Customer.City, Customer.State, Customer.Country, Customer.Email,  
Customer.Phone  
FROM Customer, Transaction  
WHERE Customer.CustomerID = Transaction.CustomerID AND Transaction.ServiceID IN(  
    SELECT ServiceID  
    FROM Service  
    WHERE ServiceType = 'Postpaid')  
UNION  
SELECT Customer.CustomerID, Customer.Name, 0, Customer.HouseNo, Customer.Street,  
Customer.City, Customer.State, Customer.Country, Customer.Email, Customer.Phone  
FROM Customer WHERE Customer.CustomerID not IN (SELECT CustomerID from  
Transaction)  
UNION  
SELECT Customer.CustomerID, Customer.Name, 0, Customer.HouseNo, Customer.Street,  
Customer.City, Customer.State, Customer.Country, Customer.Email, Customer.Phone  
FROM Customer, Transaction
```

```
WHERE Customer.CustomerID = Transaction.CustomerID AND Transaction.ServiceID IN(  
    SELECT ServiceID  
    FROM Service  
    WHERE ServiceType!= 'Postpaid');
```

**#G)**

```
SELECT * FROM Transaction;
```

## **EXTRA QUERIES**

**#1) Find the Transaction with the largest amount.**

```
SELECT * FROM Transaction  
WHERE Amount IN (SELECT max(Amount) FROM Transaction);
```

**#2) Find the name of all customers who have shipped international packages.**

```
SELECT Name From Customer  
WHERE CustomerID IN(  
    SELECT CustomerID FROM PackageDelivery WHERE  
        PkgID IN (SELECT PkgID FROM International));
```