Group 20 Project

Bhavajna Kallakuri (202003046**)**

Divyansh Jain (202003047**)**

Lab 7 (Date: 08-10-2021**)**

Bus Management System

Normalization and Schema Refinement

Relation - 1 : Bus

Attributes present in Bus relation are BusID, Bus type, No. Of seats available, Model, Total no. of seats, RouteID, Bus number

FDs in Bus are,

1. BusID → Bus type
2. BusID → No. of seats available
3. BusID → Model
4. BusID → Total No. of seats
5. BusID → Bus Number
6. BusID → RouteID
7. Bus number → BusID
8. Bus number → Bus type
9. Bus number → No. of seats available
10. Bus number → Total No. of seats
11. Bus number → Model
12. Bus number → RouteID

Here, Candidate keys are **BusID** and **Bus number**. The above relation is in BCNF because the attribute which is present in the left side of each FD is a super key(Candidate Key).

Relation - 2 : User_Login

Attributes present in User relation are UserID, Username, Password, MobileNo., Email, LName, FName, Doornum, Landmark, District, State_name, Pincode, Category, AadharNo.

FDs in the User_Login are,

1. UserID → Username
2. UserID → Username
3. UserID → Password
4. UserID → MobileNo.
5. UserID → Email
6. UserID → LName,
7. UserID → FName,
8. UserID → AadharNo.
9. UserID → Doornum
10. UserID → Landmark
11. UserID → District
12. UserID → State_name
13. UserID → Pincode
14. UserID → Category

- Here, the candidate key is **UserID**. All the attributes present in the relation User_login are included in the above FDs. The attribute which is present in the left side of each FD is UserID which is a super key.
- So, we can say that the relation **User_login** relation is present in BCNF.

Relation - 3 : Employee

Attributes present in Employee relation are EmployeeID, UserID, Role, Gender, DOB, BusID.

FDs in the Employee relation are,

1. Employee ID → UserID
2. Employee ID → Role
3. Employee ID → Gender
4. Employee ID → DOB
5. Employee ID → BusID.

- Here, the candidate key is **EmployeeID**. All the attributes which present in the above relation are included in the above FDs. EmployeeID(a super key) is present in the left side of every FD.
- So, we can say that **Employee** relation is present in BCNF.

Relation - 4 : Customer

Attributes present in Customer relation are CustomerID, BookingID, PaymentID, UserID, Gender, DOB.

FDs in the Customer relation are,

1. CustomerID → BookingID
2. CustomerID → PaymentID
3. CustomerID → UserID
4. CustomerID → Gender
5. CustomerID → DOB.

- Here, the candidate key is **CustomerID**. All the attributes which present in the above relation are included in the above FDs. CustomerID(super key) is present in the left side of every FD.
- So, we can say that **Customer** relation is present in BCNF.

Relation - 5 : Booking

Attributes present in Booking relation are BookingID, ParcelID, Source, Destination, Date of booking, Status, Total amount.

FDs in the Booking relation are,

1. BookingID → ParcelID
2. BookingID → Source
3. BookingID → Destination
4. BookingID → Date of booking
5. BookingID → Status
6. BookingID → Total amount

- Here, the candidate key is **BookingID**. All the attributes which present in the relation are included in the above FDs. BookingID(super key) is present in the left side of every FD.
- So, we can say that **Booking** relation is present in BCNF.

Relation - 6 : Route

Attributes present in Route relation are RouteID, Source, Destination, Arrival Time, Departure time, Destination, Source, Distance, Scheduled Date.

FDs in Route relation are,

1. RouteID → Source
2. RouteID → Destination
3. RouteID → Arrival Time
4. RouteID → Departure time
5. RouteID → Distance
6. RouteID → Scheduled Date

- The candidate key for the above relation is **RouteID**. The attribute which is present in the left side of each FD is RouteID(super key). It is satisfying the conditions of BCNF
- Hence, we can say that **Route** relation is present in BCNF.

Relation - 7 : Station

Attributes present in Station relation are StationID, Name, District, State_name.

FDs present in Station relation are,

1. StationID → Name
2. StationID → District
3. StationID → State_name
4. {Name, District, State_name} → StationID

- Here, the candidate keys are {StationID ; (Name, District, State_Name)}. This is also the set of super keys. The attributes which are present in the left side of each FD is a super key.
- So, we can say that **Station** relation is in BCNF.

Relation - 8 : Connects

Attributes present in Connects relation are RouteID, StationID. Both the attributes are foreign keys and we cannot list find any FDs and we cannot check the conditions for any FD. So, we can say that **Connects** relation is in BCNF.

Relation - 9 : Payment

Attributes present in Payment relation are PaymentID, BookingID, Date of Payment, Amount paid, Mode, Payment gateway

FDs present in the Payment relation are,

1. PaymentID → BookingID
2. PaymentID → Date of Payment

3. PaymentID → Amount paid
4. PaymentID → Mode
5. PaymentID → Payment gateway

Here, the candidate key is **PaymentID**. The attribute which is present in the left side is PaymentID which is a super key. So, it is satisfying the conditions of BCNF. So, we can say that **Payment** relation is in BCNF.

Relation - 10 : Parcels

Attributes present in Parcel relation are ParcelID, Product type, Weight, BusID

FDs present in the Parcel relation are,

1. ParcelID → Product type
2. ParcelID → Weight
3. ParcelID → BusID

Here, the candidate key is **ParcelID**. ParcelID is present in the left side of above 3 FDs which is a super key. So, the condition for BCNF is satisfied and we can say that **Parcel** relation is in BCNF.

Relation - 11 : Seat

Attributes which are present in Seat relation are SeatID, BusID, Status, BookingID.

FDs present in the Seat relation are,

1. {SeatID, BusID} → Status
2. {SeatID, BusID} → BookingID

Here, the candidate key is the union of SeatID and BusID i.e., {**SeatID, BusID**}. In the 2 FDs, the attribute which is present on the left side is a candidate key (also a super key). It is satisfying the condition of BCNF and hence we can say that **Seat** relation is present in BCNF.

All the relations in the given schema are in BCNF. So, we can say that the schema is in BCNF and is normalised.

REFERENCING TABLES:

Parcels, booking, Bus, connects, Seat, Payment, Customer, Employee

REFERENCE TABLE:

Route, Station, User_login, Parcels, booking, Bus, Payment, Customer

NOTE: Here, referencing tables are those in which foreign key is present and takes reference from the primary key of the reference table.

1. Parcels:

   Insert: May cause violation

   Delete: May cause violation

   Update: May cause violation

2. Booking:

   Insert: May cause violation

   Delete: May cause violation

   Update: May cause violation

3. Bus:

   Insert: May cause violation

   Delete: May cause violation

   Update: May cause violation

4. Connects:

   Insert: May cause violation

   Delete: No violation

   Update: May cause violation

5. Seat:

   Insert: May cause violation

Delete: No violation

Update: May cause violation

6. Payment:

   Insert: May cause violation

   Delete: May cause violation

   Update: May cause violation

7. Customer:

   Insert: May cause violation

   Delete: May cause violation

   Update: May cause violation

8. Employee:

   Insert: May cause violation

   Delete: No violation

   Update: May cause violation

9. Route:

   Insert: No violation

   Delete: May cause violation

   Update: May cause violation

10. Station:

    Insert: No violation

    Delete: May cause violation

    Update: May cause violation

11. User_login:

    Insert: No violation

    Delete: May cause violation

    Update: May cause violation

Solutions to the violations:

1. On delete cascade.

2. On delete set NULL.

3. On delete no action.

SQL DDL Code:

```
—DROP SCHEMA bus_database CASCADE;


CREATE SCHEMA bus_database;

SET SEARCH_PATH TO bus_database;


CREATE DOMAIN NAME_DOMAIN as VARCHAR(30);

CREATE DOMAIN DOMAIN_2 as VARCHAR(255); --To store the
variables whose size is unknown(can be very large or small)

CREATE DOMAIN ID_DOMAIN as DECIMAL(7,0);

CREATE DOMAIN USER_DOMAIN as DECIMAL(9,0);

CREATE DOMAIN MONEY_DOMAIN as DECIMAL(5,4);


CREATE TABLE Route

(

     RouteID ID_DOMAIN NOT NULL,

     Source_name NAME_DOMAIN NOT NULL,

     Departure_time CHAR(4) NOT NULL,

     Arrival_time CHAR(4) NOT NULL,

     Destination NAME_DOMAIN NOT NULL,
```

```
    Distance DECIMAL(5,3) NOT NULL,
    Scheduled_Date CHAR(10) ,
    PRIMARY KEY (RouteID)
);

CREATE TABLE Bus
(
    Type_of_bus VARCHAR(20) NOT NULL,
    No_of_seats_available INT NOT NULL,
    Model VARCHAR(20) NOT NULL,
    AC_or_not SMALLINT NOT NULL CHECK (AC_or_not = 1 or
AC_or_not = 2),
    Available_Number_of_seats INT NOT NULL,
    Total_number_of_seats INT NOT NULL,
    Bus_number VARCHAR(20) ,
    BusID ID_DOMAIN ,
    RouteID ID_DOMAIN,
    PRIMARY KEY (BusID),
    FOREIGN KEY (RouteID) REFERENCES Route (RouteID) ON
DELETE SET DEFAULT ON UPDATE CASCADE
);

CREATE TABLE Parcels
(
    ParcelID ID_DOMAIN ,
    Product_Type SMALLINT ,
    Weight INT NOT NULL,
```

```
    BusID ID_DOMAIN,
    PRIMARY KEY (ParcelID),
    FOREIGN KEY (BusID) REFERENCES Bus(BusID) ON UPDATE
CASCADE ON DELETE RESTRICT,
    CHECK(Product_Type = 1 or Product_Type = 2)
);


CREATE TABLE Booking
(
    BookingID ID_DOMAIN ,
    BusID ID_DOMAIN ,
    PaymentID ID_DOMAIN NOT NULL,
    Status BOOLEAN NOT NULL,
    Source_name NAME_DOMAIN NOT NULL,
    Destination NAME_DOMAIN NOT NULL,
    CustomerID USER_DOMAIN,
    ParcelID ID_DOMAIN ,
    Date_of_booking TIMESTAMP NOT NULL,
    Total_amount MONEY_DOMAIN NOT NULL,
    SeatID DECIMAL(10,0) NOT NULL,
    PRIMARY KEY (BookingID),
    FOREIGN KEY (ParcelID) REFERENCES Parcels(ParcelID) ON
DELETE CASCADE
);


CREATE TABLE Seat
(
```

```
    SeatID VARCHAR(5) ,
    BusID ID_DOMAIN NOT NULL,
    Status BOOLEAN DEFAULT '0',
    BookingID ID_DOMAIN,
    PRIMARY KEY (SeatID, BusID),
    FOREIGN KEY (BusID) REFERENCES Bus(BusID) ON UPDATE
CASCADE,
    FOREIGN KEY (BookingID) REFERENCES Booking(BookingID) ON
DELETE SET DEFAULT
);


CREATE TABLE User_Login
(
    UserID USER_DOMAIN ,
    Username VARCHAR(15) NOT NULL,
    Password_user VARCHAR(15) NOT NULL,
    Category SMALLINT NOT NULL,
    MobileNo VARCHAR(15) NOT NULL UNIQUE,
    Email DOMAIN_2 NOT NULL,
    Doornum NAME_DOMAIN NOT NULL,
    Landmark DOMAIN_2,
    District NAME_DOMAIN,
    State_name NAME_DOMAIN NOT NULL,
    Pincode CHAR(6) NOT NULL,
    FName NAME_DOMAIN NOT NULL,
    LName NAME_DOMAIN NOT NULL,
    AadharNo CHAR(12) NOT NULL,
```

```
    UNIQUE (FName,LName),
    PRIMARY KEY (UserID)
);

CREATE TABLE Employee
(
    EmployeeID USER_DOMAIN ,
    UserID USER_DOMAIN NOT NULL,
    BusID ID_DOMAIN ,
    Person_Role SMALLINT NOT NULL,
    Gender CHAR(1) NOT NULL CHECK(Gender = 'M' or Gender =
'F' or Gender = 'O'),
    DOB DATE ,
    PRIMARY KEY (EmployeeID),
    FOREIGN KEY (UserID) REFERENCES User_Login(UserID) ON
UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (BusID) REFERENCES Bus(BusID) ON UPDATE
CASCADE ON DELETE RESTRICT,
    CHECK (Person_Role = 1 or Person_Role = 2 or Person_Role
= 3)
);

CREATE TABLE Station
(
    StationID ID_DOMAIN ,
    Station_name NAME_DOMAIN NOT NULL,
    District NAME_DOMAIN NOT NULL,
    State_name NAME_DOMAIN  NOT NULL,
```

```
    PRIMARY KEY (StationID),
    UNIQUE (Station_name, District, State_name)
);

CREATE TABLE Payment
(
    PaymentID ID_DOMAIN ,
    Payment_mode SMALLINT NOT NULL,
    Date_of_payment TIMESTAMP ,
    Amount_Paid MONEY_DOMAIN NOT NULL,
    Payment_Gateway VARCHAR(30) NOT NULL,
    BookingID ID_DOMAIN NOT NULL,
    PRIMARY KEY (PaymentID),
    FOREIGN KEY (BookingID) REFERENCES Booking(BookingID) ON
DELETE RESTRICT
);

CREATE TABLE Customer
(
    CustomerID USER_DOMAIN ,
    UserID USER_DOMAIN NOT NULL,
    Gender CHAR(1) NOT NULL CHECK(Gender = 'M' or Gender =
'F' or Gender = 'O'),
    DOB DATE NOT NULL,
    PaymentID ID_DOMAIN NOT NULL,
    BookingID ID_DOMAIN NOT NULL,
    PRIMARY KEY (CustomerID),
```

```
    FOREIGN KEY (PaymentID) REFERENCES Payment(PaymentID) ON
UPDATE CASCADE ON DELETE RESTRICT,

    FOREIGN KEY (UserID) REFERENCES User_Login(UserID) ON
UPDATE CASCADE ON DELETE CASCADE,

    FOREIGN KEY (BookingID) REFERENCES Booking(BookingID) ON
UPDATE CASCADE ON DELETE RESTRICT

);


CREATE TABLE Connects

(

    RouteID ID_DOMAIN NOT NULL,

    StationID ID_DOMAIN NOT NULL,

    PRIMARY KEY (RouteID, StationID),

    FOREIGN KEY (RouteID) REFERENCES Route(RouteID) ON UPDATE
CASCADE ON DELETE CASCADE,

    FOREIGN KEY (StationID) REFERENCES Station(StationID) ON
UPDATE CASCADE ON DELETE CASCADE

);
```