/home/student/.netbeans/12.0/config/Databases/SQLCommands/SQL 1.sql

```sql
 1
 2  PRAGMA foreign_keys = OFF;
 3  BEGIN TRANSACTION;
 4
 5  -- ENTITY Table: Customer
 6  CREATE TABLE Customer (
 7      Customer_ID SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
 8      Customer_First_Name VARCHAR (50),
 9      Customer_Last_Name VARCHAR (50),
10      PRIMARY KEY (Customer_ID)
11              );
12
13  -- ENTITY Table: Order
14  CREATE TABLE Order_Des (
15      Order_ID SMALLINT UNSIGNED NOT NULL,
16      Customer_ID INTEGER NOT NULL,
17      Order_Cost INTEGER,
18      Order_Status VARCHAR (50),
19      PRIMARY KEY (Order_ID),
20      CONSTRAINT fk_order_des FOREIGN KEY (Customer_ID) REFERENCES Customer (Customer_ID)
21
22        );
23
24  --ENTITY Table: Product
25  CREATE TABLE Product (
26      Product_ID SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,
27      Order_ID INTEGER NOT NULL,
28      Department VARCHAR (50),
29      Subscription_Option BOOLEAN,
30      Product_Name VARCHAR (100),
31      Product_Description VARCHAR (255),
32      Product_Avg_Ratings INTEGER,
33      Product_Company VARCHAR (50),
34      Product_Availability BOOLEAN,
35      PRIMARY KEY (Product_ID),
36      CONSTRAINT fk_product FOREIGN KEY (Order_ID) REFERENCES Order_Des (Order_ID)
37
38              );
39
40  -- ENTITY Table: Deals
41  CREATE TABLE Deals (
42      Deal_ID SMALLINT UNSIGNED NOT NULL,
43      Product_ID INTEGER NOT NULL,
44      Product_Sell_Performance VARCHAR (255),
45      Price INTEGER,
46      Dicount_Rates INTEGER,
47      PRIMARY KEY (Deal_ID),
48      CONSTRAINT fk_deal FOREIGN KEY (Product_ID) REFERENCES Product (Product_ID)
49
50        );
51
52  -- ENTITY Table: Review
53  CREATE TABLE Review (
54      Review_ID SMALLINT UNSIGNED NOT NULL,
55      Customer_ID INTEGER NOT NULL,
56      Rating_Stars INTEGER,
57      Comment_Sentiment VARCHAR (50),
58      PRIMARY KEY (Review_ID),
59      CONSTRAINT fk_review FOREIGN KEY (Customer_ID) REFERENCES Customer (Customer_ID)
60
61        );
62
```

```
63
64
65  -- MULTIVALUED Table: Billing_Detail
66  CREATE TABLE Billing_Detail (
67      Billing_ID SMALLINT UNSIGNED NOT NULL,
68      Customer_ID INTEGER NOT NULL,
69      Street_Num INTEGER,
70      Street_Name VARCHAR (50),
71      Postal_Code VARCHAR (50),
72      City VARCHAR (50),
73      Province VARCHAR (50),
74      Country VARCHAR (50),
75      PRIMARY KEY (Billing_ID),
76      CONSTRAINT fk_bill FOREIGN KEY (Customer_ID) REFERENCES Customer (Customer_ID)
77      );
78
79
80  -- MULTIVALUED Table: Product_Description
81  CREATE TABLE Product_Description (
82      Description_ID SMALLINT UNSIGNED NOT NULL,
83      Product_ID INTEGER NOT NULL,
84      Product_Description VARCHAR (255),
85      PRIMARY KEY (Description_ID),
86      CONSTRAINT fk_Product_ID FOREIGN KEY (Product_ID) REFERENCES Product (Product_ID)
87      );
88
89
90  -- RELATIONSHIP Table: Places
91  CREATE TABLE Places (
92      Customer_ID INTEGER REFERENCES Customer (Customer_ID) ON UPDATE RESTRICT NOT NULL,
93      Order_ID INTEGER REFERENCES Orders_Des (Order_ID) ON UPDATE RESTRICT NOT NULL,
94      Num_Of_Orders INTeger UNIQUE NOT NULL,
95      Order_date INTeger UNIQUE NOT NULL
96      );
97
98  -- RELATIONSHIP Table: Contains
99  CREATE TABLE Contains_ (
100     Order_ID INTEGER REFERENCES Orders_Des (Order_ID) ON UPDATE RESTRICT NOT NULL,
101     Product_ID INTEGER REFERENCES Product (Product_ID) ON UPDATE RESTRICT NOT NULL
102     );
103
104 -- RELATIONSHIP Table: Display
105 CREATE TABLE Display_ (
106     Deal_ID INTEGER REFERENCES Deals (Deal_ID) ON UPDATE RESTRICT NOT NULL,
107     Product_ID INTEGER REFERENCES Product (Product_ID) ON UPDATE RESTRICT NOT NULL
108     );
109
110 -- RELATIONSHIP Table: Has
111 CREATE TABLE Has_ (
112     Product_ID INTEGER REFERENCES Product (Product_ID) ON UPDATE RESTRICT NOT NULL,
113     Review _ID INTEGER REFERENCES Review (Review_ID) ON UPDATE RESTRICT NOT NULL
114     );
115
116 -- RELATIONSHIP Table: Write
117 CREATE TABLE Write_ (
118     Review _ID INTEGER REFERENCES Review (Review_ID) ON UPDATE RESTRICT NOT NULL,
119     Customer_ID INTEGER REFERENCES Customer (Customer_ID) ON UPDATE RESTRICT NOT NULL
120 );
121
122
123 COMMIT TRANSACTION;
124 PRAGMA foreign_keys = off;
125
126
```