Project Report

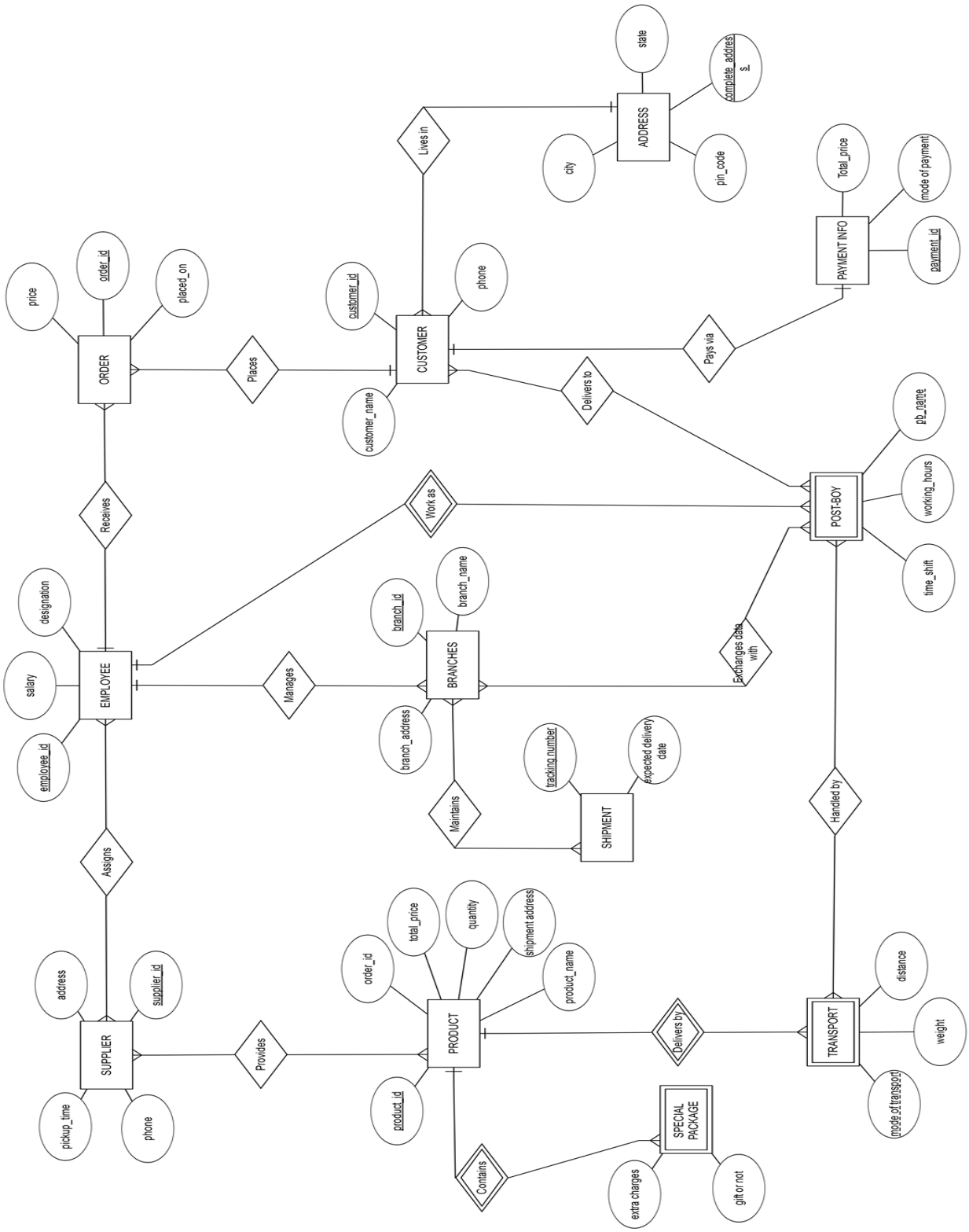
Courier Database Management System

(Team members : Rishabh Kumar Kandoi, Sudarshan Raghavan, S6 batch, B.Tech – 2nd year)

A. Problem Description

Handling logistics and queries in a courier database system is a complex task and requires undulating attention. However it is integral to any e-commerce business or mail deliveries. It all starts with the customer who is uniquely identified with an id and has other information such as name and phone number. A customer has an associated weak entity set called “address” that contains information about where the customer lives in. The customer will also have payment information uniquely identified by an id. When a customer places an order, the order query will have the id of the customer and a unique order id. This order will be received by an employee representing the concerned courier service. An employee has a unique id. An employee of the courier service can either have management duties or work as post boy. As a post boy, the employee has fixed working hours for every day with varying time shifts. As a manger, the employee may manage multiple branches of the courier service. Each branch has a unique id, name and an address. Multiple branches maintain multiple shipments each having a unique tracking number and expected delivery date of the mail.
Multiple employees each handling a single query generated by a different customer assigns these queries to the concerned supplier as stated in the order. The supplier is uniquely identified by an id. Suppliers will submit the required product package to the transport system. Each product package submitted by a supplier is uniquely identified by an id. It contains the name of the product, its price, order id, quantity and shipping address. Furthermore, a package may contain a “special package”. A mode of transport can contain a defined total weight of packages and may travel a defined maximum distance from its current location. Transport will jettison its packages to multiple post boys who will then deliver the packages to the concerned customer.

B. Entity – Relationship diagram



C. How to convert ER model to equivalent relational tables?

In order to convert ER diagram to tables, we have to basically follow the following rules:
1) For many-to-many relationship, the relation has its own table, with the primary keys (or 'weakly unique' attribute in case of weak relation) of the participating entities and its own descriptive attributes as the columns of the relation table, and call the primary keys of participating entities as it's 'Foreign key' then.
2) For one-to-many or many-to-one relation, we can have a separate table for relation, but it is preferred to avoid it, and rather add the primary key of 'one' side (or 'weakly unique' attribute in case of weak relation) to 'many' side and call it as 'Foreign key' then.
3) For one-to-one relation also, we have to simply add the primary key of any of the 'one' side to the other 'one' side and call it as 'Foreign key' then. If at the same time it is weak relationship, then the weak side's 'weakly unique' attribute gets added in its parent entity as 'Foreign key'.

To insert arrows in relational schema, we can simply put the head of arrow to 'one' side in case of one-to-many or many-to-one, and in case of many-to-many, put arrow head on both side away from the relation table.

Entity/Relation	Cardinality	Participating Attributes/Entities	Description	SQL Statements
Customer	Nil	Nil	This entity receives payment_id from Payment entity via “Pays via”, complete address from Address via “Lives in”. Its customer_id will transfer to Order via “Places”	CREATE TABLE CUSTOMER (customer_id VARCHAR(20) NOT NULL, customer_name VARCHAR(20) NOT NULL, phone INT(20) NOT NULL, payment_id VARCHAR(20) NOT NULL, complete_address VARCHAR(50) NOT NULL, PRIMARY KEY (customer_id), FOREIGN KEY (complete_address) REFERENCES ADDRESS_DECOMPOSED1(complete_address), FOREIGN KEY (payment_id) REFERENCES PAYMENT_INFO(payment_id));
Payment_info	Nil	Nil	This entity transfers payment_id to customer via “Pays via”.	CREATE TABLE PAYMENT_INFO (mode_of_payment VARCHAR(20) NOT NULL, Total_price INT(10) NOT NULL, payment_id VARCHAR(20) NOT NULL, PRIMARY KEY (payment_id));
Pays via	One-to-one	Customer and Payment_info	Payment_id from Payment Info	Nil

			will transfer to Customer_id acting as a foreign key in it.	
Address	Nil	Nil	This entity transfers complete_address to customer via “Lives in”	Nil (This table will be decomposed subsequently)
Lives in	Many-to-one	Customer and Address	Complete_address from Address will transfer to Customer acting as a foreign key in it.	Nil
Orders	Nil	Nil	This entity receives customer_id from Customer	CREATE TABLE ORDERS (order_id VARCHAR(20) NOT NULL, customer_id VARCHAR(20) NOT NULL, placed_on DATE NOT NULL, price INT(10), employee_id VARCHAR(20) NOT NULL, PRIMARY KEY(order_id), FOREIGN KEY (customer_id) REFERENCES CUSTOMER (customer_id),

				FOREIGN KEY (employee_id) REFERENCES EMPLOYEE (employee_id));
Places	Many-to-one	Order and Customer	Customer_id from Customer will transfer to Order acting as a foreign key in it.	Nil
Employee	Nil	Nil	For this entity, employee_id will transfer to Branches via “Manages” relation. It will also transfer to Post-Boy via “Works as”	CREATE TABLE EMPLOYEE (employee_id VARCHAR(20) NOT NULL, salary INT(6) NOT NULL, designation VARCHAR(20) NOT NULL, PRIMARY KEY (employee_id));
Address_Decomposed1	Nil	Nil	Decomposed table of Address with complete_address as primary key	CREATE TABLE ADDRESS_DECOMPOSED1 (pin_code INT(8) NOT NULL, complete_address VARCHAR(50) NOT NULL, PRIMARY KEY (complete_address));
Adress_Decomposed2	Nil	Nil	Decomposed table of	CREATE TABLE ADDRESS_DECOMPOSED2 (city VARCHAR(20) NOT NULL,

			Address with pin_code as primary key	state VARCHAR(20) NOT NULL, pin_code INT(8) NOT NULL, PRIMARY KEY (pin_code));
Supplier_Decomposed1	Nil	Nil	Decomposed table of Supplier with supplier_id as primary key	CREATE TABLE SUPPLIER_DECOMPOSED1 (pickup_time VARCHAR(20) NOT NULL, supplier_id VARCHAR(20) NOT NULL, phone INT(20) NOT NULL, PRIMARY KEY (supplier_id));
Supplier_Decomposed2	Nil	Nil	Decomposed table of Supplier with phone as primary key	CREATE TABLE SUPPLIER_DECOMPOSED2 (address VARCHAR(20) NOT NULL, phone INT(20) NOT NULL, PRIMARY KEY (phone));
Branches	Nil	Nil	Receives a unique employee_id from Employee via “Manages”	Nil (This table will be decomposed subsequently)
Branches_Decomposed1	Nil	Nil	Decomposed table of Branches with branch_id as primary key	CREATE TABLE BRANCHES_DECOMPOSED1 (branch_id VARCHAR(20) NOT NULL, branch_name VARCHAR(20) NOT NULL, branch_address VARCHAR(50) NOT NULL, PRIMARY KEY (branch_id));

Branches_Decomposed2	Nil	Nil	Decomposed table of Branches with employee_id and branch_id as primary key	CREATE TABLE BRANCHES_DECOMPOSED2 (employee_id VARCHAR(20) NOT NULL, branch_id VARCHAR(20) NOT NULL, branch_address VARCHAR(50) NOT NULL, PRIMARY KEY (employee_id, branch_id));
Manages	One-to-many	Employee and Branches	One employee can manage many branches, hence a unique employee_id is transferred to an instance of Branches.	Nil
Shipment	Nil	Nil	Contains tracking number as its primary key	CREATE TABLE SHIPMENT (tracking_number VARCHAR(20) NOT NULL, expected_delivery_date DATE NOT NULL, PRIMARY KEY (tracking_number));
Maintains	Many-to-many	Shipment and Branches	Nil	CREATE TABLE Maintains (branch_id VARCHAR(20) NOT NULL, tracking_number VARCHAR(20) NOT NULL,

				PRIMARY KEY (branch_id, tracking_number), FOREIGN KEY (branch_id) REFERENCES BRANCHES_DECOMPOSED1(branch_id), FOREIGN KEY (tracking_number) REFERENCES SHIPMENT(tracking_number));
Post-Boy	Nil	Nil	Receives employee_id from Employee as an employee can work as a post-boy or not.	CREATE TABLE POST_BOY (time_shift VARCHAR(20) NOT NULL, pb_name VARCHAR(20) NOT NULL, working_hours INT(2) NOT NULL, employee_id VARCHAR(20) NOT NULL, PRIMARY KEY (pb_name, employee_id), FOREIGN KEY (employee_id) REFERENCES EMPLOYEE (employee_id));
Exchanges data with	Many-to-many	Branches and Post-boy	Nil	CREATE TABLE Exchanges_data_with (branch_id VARCHAR(20) NOT NULL, pb_name VARCHAR(20) NOT NULL, employee_id VARCHAR(20) NOT NULL, FOREIGN KEY (branch_id) REFERENCES BRANCHES_DECOMPOSED1(branch_id), FOREIGN KEY (pb_name, employee_id) REFERENCES POST_BOY(pb_name, employee_id)

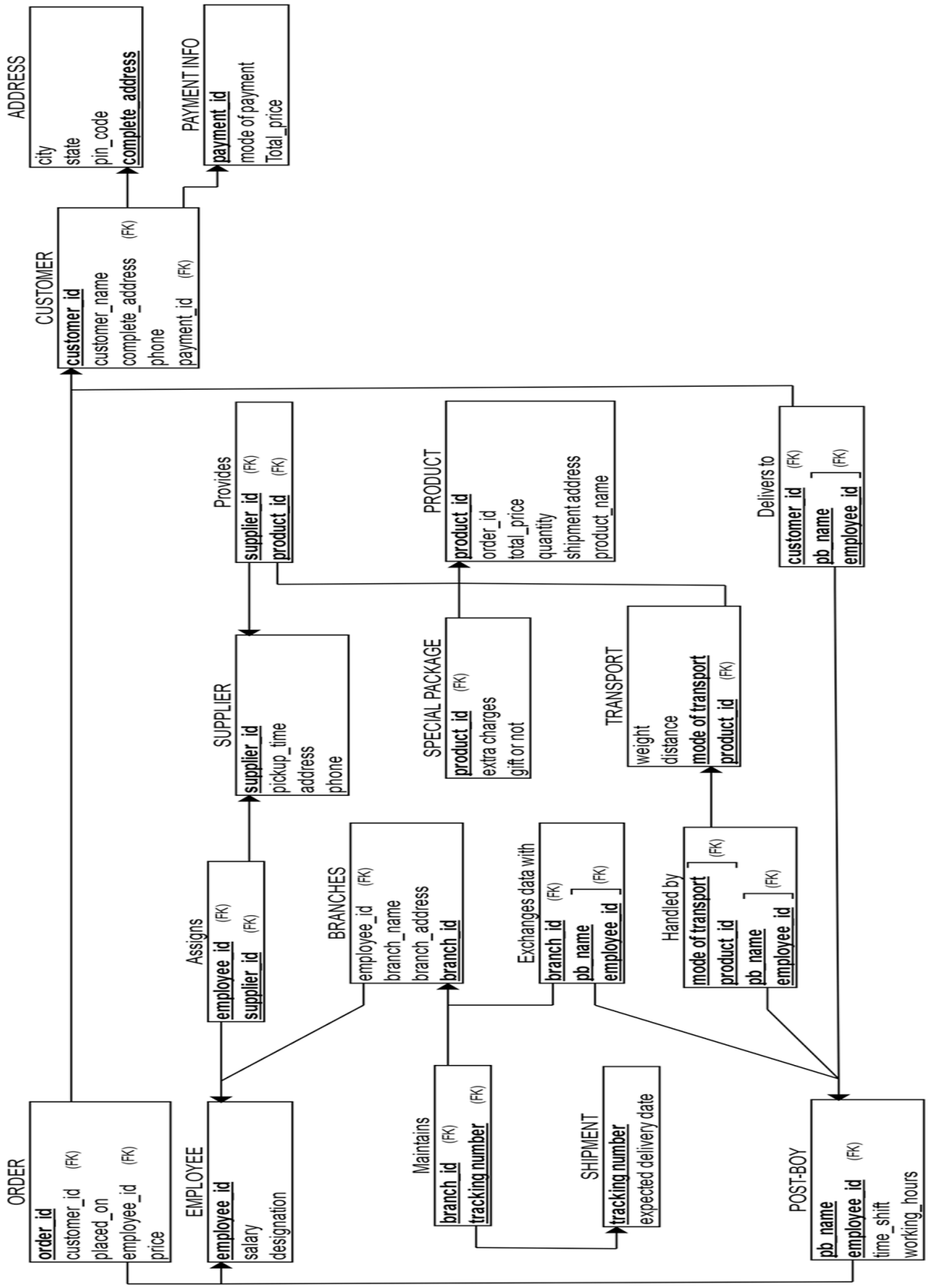
				);
Supplier	Nil	Nil	Nil	Nil (This table will be decomposed subsequently)
Assigns	Many-to-many	Supplier and Employee	Nil	CREATE TABLE Assigns (employee_id VARCHAR(20) NOT NULL, supplier_id VARCHAR(20) NOT NULL, FOREIGN KEY (employee_id) REFERENCES EMPLOYEE(employee_id), FOREIGN KEY (supplier_id) REFERENCES SUPPLIER_DECOMPOSED1(supplier_id));
Product	Nil	Nil	This entity contains order_id from Order, transfers its product_id to Transport via “Delivers by” and to Special Package via “Contains”	CREATE TABLE PRODUCT (product_id VARCHAR(20) NOT NULL, order_id VARCHAR(20) NOT NULL, total_price INT(10) NOT NULL, quantity INT(5) NOT NULL, shipment_address VARCHAR(50) NOT NULL, product_name VARCHAR(20) NOT NULL, PRIMARY KEY (product_id));
Provides	Many-to-many	Supplier and Product	Nil	CREATE TABLE Provides (supplier_id VARCHAR(20) NOT NULL, product_id VARCHAR(20) NOT NULL,

				PRIMARY KEY (supplier_id, product_id), FOREIGN KEY (supplier_id) REFERENCES SUPPLIER_DECOMPOSED1(supplier_id), FOREIGN KEY (product_id) REFERENCES PRODUCT(product_id));
Special Package	Nil	Nil	This entity receives product_id from Product via ”Contains”	CREATE TABLE SPECIAL_PACKAGE (extra_charges INT(5) NOT NULL, gift_or_not INT NOT NULL, product_id VARCHAR(20) NOT NULL, FOREIGN KEY (product_id) REFERENCES PRODUCT(product_id), CONSTRAINT chk_gift CHECK (gift_or_not>=0 AND gift_or_not<=1));
Contains	One-to-many	Product and Special Package	Here, product_id will transfer to Special Package acting as a foreign key in it.	Nil
Transport	Nil	Nil	This entity receives product_id from Product	CREATE TABLE TRANSPORT (mode_of_transport VARCHAR(20) NOT NULL, weight INT(4) NOT NULL, distance INT(5) NOT NULL,

			via Delivers_by	product_id VARCHAR(20) NOT NULL, PRIMARY KEY (mode_of_transport,product_id), FOREIGN KEY (product_id) REFERENCES PRODUCT (product_id));
Delivers by	One-to-many	Product and Transport	Here, product_id from Product gets transferred to Transport	Nil
Works as	One-to-many	Employee and Post-Boy	Transfers employee_id from Employee to Post-Boy	Nil
Delivers to	Many-to-many	Post-Boy and Customer	It contains pb_name referenced in Post-Boy and customer_id referenced in Customer	CREATE TABLE Delivers_to (pb_name VARCHAR(20) NOT NULL, customer_id VARCHAR(20) NOT NULL, FOREIGN KEY (pb_name) REFERENCES POST_BOY(pb_name), FOREIGN KEY (customer_id) REFERENCES CUSTOMER(customer_id));
Handled by	Many-to-many	Transport and Post-boy	It has foreign keys (mode_of_transport,	CREATE TABLE Handled_by (mode_of_transport VARCHAR(20) NOT NULL, pb_name VARCHAR(20) NOT NULL,

			product_id) referenced in Transport and (pb_name, employee_id) referenced in Post-Boy	product_id VARCHAR(20) NOT NULL, employee_id VARCHAR(20) NOT NULL, FOREIGN KEY (mode_of_transport,product_id) REFERENCES TRANSPORT(mode_of_transport,product_id), FOREIGN KEY (pb_name, employee_id) REFERENCES POST_BOY(pb_name, employee_id));

D. Table Schema for all tables



E. Functional Dependencies on Tables

1. Order:

[order_id, customer_id, placed_on, employee_id, price] --orignial table schema

{order_id} → {customer_id}, {employee_id}, {placed_on}, {price},
{customer_id, employee_id}, {customer_id, placed_on},
{customer_id, price}, {employee_id, placed_on}, {employee_id, price},
{placed_on, price}
{customer_id, employee_id, placed_on},{customer_id, employee_id, price},
{customer_id, placed_on, price}, {employee_id, placed_on, price}
{customer_id, placed_on, employee_id, price}

2. Customer

[customer_id, payment_id, customer_name, complete_address, phone] --original table schema

{customer_id} → {payment_id}, {customer_name}, {complete_address}, {phone},
{payment_id, customer_name}, {payment_id, complete_address},
{payment_id, phone}, {customer_name, complete_address},
{payment_id, phone}, {customer_name, complete_address},
{customer_name, phone}, {complete_address, phone},
{payment_id, customer_name, complete_address},
{payment_id, customer_name, phone},
{payment_id, complete_address, phone},
{customer_name, complete_address, phone}
{payment_id, customer_name, complete_address, phone}

{payment_id} → {customer_name}, {phone}, {complete_address},
{customer_name, phone}, {customer_name, complete_address},
{phone, complete_address}, {customer_name, phone, complete_address}

3. Employee:

[employee_id, salary, designation] --original table schema

{employee_id} → {salary}, {designation}, {salary, designation}

4. Address:

[complete_address , city, state, pin_code] --original table schema

{complete_address} → {city}, {state}, {pin_code},
{city, state}, {city, pin_code}, {state, pin_code},
{city, state, pin_code}

{pin_code} → {city}, {state}, {city, state}

BCNF FORM

[complete_address, pincode] --decomposed table schema

{complete_address} → {pincode}

[pin_code, city, state] --decomposed table schema

{pin_code} → {city}, {state}, {city, state}

5. Supplier:

[supplier_id, pickup_time, address, phone] --original table schema

{supplier_id} → {pickup_time}, {address}, {phone}, {pickup_time, address}.
{pickup_time, phone}, {address, phone},
{pickup_time, address, phone}

{phone} → {address}

BCNF FORM

[supplier_id, pickup_time, phone] --decomposed table schema

{supplier_id} → {pickup_time}, {phone}, {pickup_time, phone}

[phone, address] --decomposed table schema

{phone} → {address}

6. Branches:

[employee_id, branch_name, branch_id, branch_address] --original table schema

{branch_id} → {branch_name}, {branch_address}, {branch_name, branch_address}

{branch_address} → {branch_name}, {branch_id}, {branch_name, branch_id}

{employee_id, branch_id} → {branch_address}, {branch_name},
{branch_address, branch_name}

{employee_id, branch_address} → {branch_id}, {branch_name},
{branch_id, branch_name}

BCNF FORM

[branch_id, branch_name, branch_address] --decomposed table schema

{branch_id} → {branch_address}, {branc_name}, {branch_address, branch_name}

[employee_id, branch_id, branch_address] --decomposed table schema

{employee_id, branch_id} → {branch_address}

7. Payment_info:

[payment_id, mode_of_payment, total_price] --original table schema

{payment_id} → {mode_of_payment}, {total_price}, {mode_of_payment, total_price}

8. Shipment:

[tracking_number, expected_delivery_date] --original table schema

{tracking_number} → {expected_delivery_date}

9. Special_package:

[product_id, extra_charges, gift_or_not] --original table schema

{product_id} → {extra_charges}, {gift_or_not}, {extra_charges, gift_or_not}

10. Product:

[product_id, order_id, total_price, quantity, shipment_address, product_name] --original table schema

{product_id} → {order_id}, {total_price}, {quantity}, {shipment_address}, {product_name}
{order_id, total_price}, {order_id, quantity}, {order_id, product_name}, {order_id, shipment_address}, {total_price, quantity}, {total_price, shipment_address}, {total_price, product_name}, {quantity, shipment_address}, {quantity, product_name}, {shipment_address, product_name}, {order_id, total_price, quantity}, {order_id, total_price, product_name}, {order_id, total_price, shipment_address}, {order_id, quantity, shipment_address}, {order_id, quantity, product_name}, {order_id, shipment_address, product_name}, {total_price, quantity, shipment_address}, {total_price, quantity, product_name}, {quantity, shipment_address, product_name}, {order_id, total_price, quantity, shipment_address}, {order_id, total_price, quantity, product_name}, {order_id, quantity, shipment_address, product_name}, {order_id, total_price, shipment_address, product_name}, {total_price, quantity, shipment_address, product_name}, {order_id, total_price, quantity, shipment_address, product_name}

11. Post-Boy:

[pb_name, employee_id, time_shift, working_hours] --original table schema

{employee_id} → {pb_name}, {time_shift}, {working_hours}, {pb_name, time_shift}, {pb_name, working_hours}, {time_shift, working_hours}, {pb_name, time_shift, working_hourse}

12. Transport:

[weight, distance, mode_of_transport, product_id]

{product_id, mode_of_transport} → {weight}, {distance}, {weight, distance}

F. Normalization Process

All the Functional Dependencies defined above have been converted to **BCNF** form. We know that in BCNF, the left side of the FD must always be a key, so in the case of “Branches”, “Address”, and “Supplier” it was imperative to decompose the original table schema into subsequent tables whilst preserving FD and now all those table’s FDs are in BCNF form (since all FDs were based on key only).

For example,
1) In Address,

The FD, {pin code} → {city}, {state}, {city, state} violates BCNF, since key is actually {complete_address}.

On decomposing, we get,
[complete_address, pin_code] & [pin_code, city, state]. In doing so, we have preserved the originally defined FDs and this final form is now in BCNF.

2) Similarly in Supplier,

The FD, {phone} → {address} violates BCNF, since key is actually {supplier_id}.

On decomposing, we get,
[supplier_id, pickup_time, phone] & [phone, address]. In doing so, we have preserved the originally defined FDs and this final form is now in BCNF.

3) Also in Branch,

The FD, branch_address → {branch_name}, {branch_id}, {branch_name, branch_id} violates BCNF, since key is actually {branch_id}.

On decomposing, we get,

[branch_id, branch_address, branch_name] & [employee_id, branch_id, branch_address].]. In doing so, we have preserved the originally defined FDs and this final form is now in BCNF.

G. Sample Output (screenshot)

```
CREATE TABLE PAYMENT_INFO
(
  mode_of_payment VARCHAR(20) NOT NULL,
  Total_price INT(10) NOT NULL,
  payment_id VARCHAR(20) NOT NULL,
  PRIMARY KEY (payment_id)
);

insert into PAYMENT_INFO values("COD", 1500, "COD_100511");
insert into PAYMENT_INFO values("VISA", 250, "HDFC_VISA_120317");
insert into PAYMENT_INFO values("PAYTM", 300, "PAYTM_300916");
insert into PAYMENT_INFO values("VISA", 150, "ICICI_VISA_100213");
insert into PAYMENT_INFO values("MOBIKWIK", 500, "MOBI_210817");
insert into PAYMENT_INFO values("PAYPAL", 4500, "PAYPAL_VISA_211015");
insert into PAYMENT_INFO values("COD", 90, "COD_040413");
insert into PAYMENT_INFO values("PAYTM", 2700, "PAYTM_060112");
insert into PAYMENT_INFO values("VISA", 6500, "YES_VISA_311214");
insert into PAYMENT_INFO values("COD", 400, "COD_161111");
```

mode_of_payment	Total_price	payment_id
COD	90	COD_040413
COD	1500	COD_100511
COD	400	COD_161111
VISA	250	HDFC_VISA_120317
VISA	150	ICICI_VISA_100213
MOBIKWIK	500	MOBI_210817
PAYPAL	4500	PAYPAL_VISA_211015
PAYTM	2700	PAYTM_060112
PAYTM	300	PAYTM_300916
VISA	6500	YES_VISA_311214
NULL	NULL	NULL

```sql
CREATE TABLE SUPPLIER_DECOMPOSED1
(
  pickup_time VARCHAR(20) NOT NULL,
  supplier_id VARCHAR(20) NOT NULL,
  phone INT(20) NOT NULL,
  PRIMARY KEY (supplier_id)
);


CREATE TABLE SUPPLIER_DECOMPOSED2
(
  address VARCHAR(20) NOT NULL,
  phone INT(20) NOT NULL,
  PRIMARY KEY (phone)
);

insert into SUPPLIER_DECOMPOSED1 values("09:30", "INFRACARE",782940593);
insert into SUPPLIER_DECOMPOSED1 values("10:00", "SERVICER",938572057);
insert into SUPPLIER_DECOMPOSED1 values("08:30", "CUREWO",782940593);
insert into SUPPLIER_DECOMPOSED1 values("00:30", "BASE12", 819403856);
insert into SUPPLIER_DECOMPOSED1 values("21:00", "OPERATE", 901836927);
insert into SUPPLIER_DECOMPOSED1 values("09:45", "MACHINE", 782945938);
insert into SUPPLIER_DECOMPOSED1 values("09:30", "COMPUTERS", 93572057);
insert into SUPPLIER_DECOMPOSED1 values("00:30", "ALLDEVICES", 994038576);
insert into SUPPLIER_DECOMPOSED1 values("08:30", "REPAIRER", 789405938);
insert into SUPPLIER_DECOMPOSED1 values("20:15", "LEATHERCARE", 910193847);

insert into SUPPLIER_DECOMPOSED2 values("TILAK_NAGAR, JAIPUR", 782940938);
insert into SUPPLIER_DECOMPOSED2 values("TONK_ROAD, DELHI", 938572051);
insert into SUPPLIER_DECOMPOSED2 values("MALL_ROAD, HYDERABAD", 819038576);
insert into SUPPLIER_DECOMPOSED2 values("AJMER_ROAD, JAIPUR", 901863927);
insert into SUPPLIER_DECOMPOSED2 values("TONK_ROAD, DELHI", 901013847);
```

pickup_time	supplier_id	phone
00:30	ALLDEVICES	994038576
00:30	BASE12	819403856
09:30	COMPUTERS	93572057
08:30	CUREWO	782940593
09:30	INFRACARE	782940593
20:15	LEATHERCARE	910193847
09:45	MACHINE	782945938
21:00	OPERATE	901836927
08:30	REPAIRER	789405938
10:00	SERVICER	938572057
NULL	NULL	NULL

address	phone
TILAK_NAGAR, JAIPUR	782940938
MALL_ROAD, HYDERABAD	819038576
TONK_ROAD, DELHI	901013847
AJMER_ROAD, JAIPUR	901863927
TONK_ROAD, DELHI	938572051
NULL	NULL

```sql
CREATE TABLE ADDRESS_DECOMPOSED1
(
  pin_code INT(8) NOT NULL,
  complete_address VARCHAR(50) NOT NULL,
  PRIMARY KEY (complete_address)
);

CREATE TABLE ADDRESS_DECOMPOSED2
(
  city VARCHAR(20) NOT NULL,
  state VARCHAR(20) NOT NULL,
  pin_code INT(8) NOT NULL,
  PRIMARY KEY (pin_code)
);
insert into ADDRESS_DECOMPOSED1 values(302010, "JAY, A-21, NIRMAN NAGAR, JAIPUR");
insert into ADDRESS_DECOMPOSED1 values(31928, "MAYANK, B-4, VAISHALI, KOTA");
insert into ADDRESS_DECOMPOSED1 values(302012, "NINA, A-56, TAGORE NAGAR, JAIPUR");
insert into ADDRESS_DECOMPOSED1 values(24789, "JAY, C-32, TILAK NAGAR, DELHI");
insert into ADDRESS_DECOMPOSED1 values(31928, "KAMAL, C-1, VAISHALI, KOTA");
insert into ADDRESS_DECOMPOSED1 values(92847, "MAYANK, B-3, MALL ROAD, HYDERABAD");
insert into ADDRESS_DECOMPOSED1 values(302003, "HEMA, A-2, OLD CITY, JAIPUR");
insert into ADDRESS_DECOMPOSED1 values(98427, "BOBBY, D-32, KAYAK MARG, MUMBAI");
insert into ADDRESS_DECOMPOSED1 values(24765, "KAMAL, D-10, JAIL ROAD, DELHI");
insert into ADDRESS_DECOMPOSED1 values(82376, "JAY, A-31, MANA MARG, LUCKNOW");

insert into ADDRESS_DECOMPOSED2 values("JAIPUR", "RAJASTHAN", 302010);
insert into ADDRESS_DECOMPOSED2 values("KOTA", "RAJASTHAN", 31928);
insert into ADDRESS_DECOMPOSED2 values("JAIPUR", "RAJASTHAN", 302012);
insert into ADDRESS_DECOMPOSED2 values("DELHI", "DELHI", 24789);
insert into ADDRESS_DECOMPOSED2 values("HYDERABAD", "AP", 92847);
insert into ADDRESS_DECOMPOSED2 values("JAIPUR", "RAJASTHAN", 302003);
insert into ADDRESS_DECOMPOSED2 values("MUMBAI", "MAHARASHTRA", 98427);
insert into ADDRESS_DECOMPOSED2 values("DELHI", "DELHI", 24765);
insert into ADDRESS_DECOMPOSED2 values("LUCKNOW", "UP", 82376);
```

pin_code	complete_address
98427	BOBBY, D-32, KAYAK MARG, MUMBAI
302003	HEMA, A-2, OLD CITY, JAIPUR
302010	JAY, A-21, NIRMAN NAGAR, JAIPUR
82376	JAY, A-31, MANA MARG, LUCKNOW
24789	JAY, C-32, TILAK NAGAR, DELHI
31928	KAMAL, C-1, VAISHALI, KOTA
24765	KAMAL, D-10, JAIL ROAD, DELHI
92847	MAYANK, B-3, MALL ROAD, HYDERABAD
31928	MAYANK, B-4, VAISHALI, KOTA
302012	NINA, A-56, TAGORE NAGAR, JAIPUR
NULL	NULL

city	state	pin_code
DELHI	DELHI	24765
DELHI	DELHI	24789
KOTA	RAJASTHAN	31928
LUCKNOW	UP	82376
HYDERABAD	AP	92847
MUMBAI	MAHARASHTRA	98427
JAIPUR	RAJASTHAN	302003
JAIPUR	RAJASTHAN	302010
JAIPUR	RAJASTHAN	302012
NULL	NULL	NULL

```
CREATE TABLE CUSTOMER
(
  customer_id VARCHAR(20) NOT NULL,
  customer_name VARCHAR(20) NOT NULL,
  phone INT(20) NOT NULL,
  payment_id VARCHAR(20) NOT NULL,
  complete_address VARCHAR(50) NOT NULL,
  PRIMARY KEY (customer_id),
  FOREIGN KEY (complete_address) REFERENCES ADDRESS_DECOMPOSED1(complete_address),
  FOREIGN KEY (payment_id) REFERENCES PAYMENT_INFO(payment_id)
);

insert into CUSTOMER values("JAY12", "JAY MALHOTRA", 918236454, "PAYPAL_VISA_211015", "JAY, A-21, NIRMAN NAGAR, JAIPUR");
insert into CUSTOMER values("MANK1", "MAYANK KUMAR", 890706152, "YES_VISA_311214", "MAYANK, B-4, VAISHALI, KOTA");
insert into CUSTOMER values("NIN098", "NINA MALIK", 900131414, "COD_100511", "NINA, A-56, TAGORE NAGAR, JAIPUR");
insert into CUSTOMER values("JAMAL", "JAY MALHOTRA", 910292150, "MOBI_210817", "JAY, C-32, TILAK NAGAR, DELHI");
insert into CUSTOMER values("KAM43", "KAMAL JAIN", 992031154, "COD_040413", "KAMAL, C-1, VAISHALI, KOTA");
insert into CUSTOMER values("MAYANK67", "MAYANK KUMAR", 929011441, "PAYTM_060112", "MAYANK, B-3, MALL ROAD, HYDERABAD");
insert into CUSTOMER values("HEMA56", "HEMA JAIN", 888223110, "PAYTM_300916", "HEMA, A-2, OLD CITY, JAIPUR");
insert into CUSTOMER values("BOB007", "BOBBY DUDE", 818016883, "HDFC_VISA_120317", "BOBBY, D-32, KAYAK MARG, MUMBAI");
insert into CUSTOMER values("KAMAL", "KAMAL JAIN", 918276454, "ICICI_VISA_100213", "KAMAL, D-10, JAIL ROAD, DELHI");
insert into CUSTOMER values("JAY109", "JAY MALHOTRA", 982031154, "COD_161111", "JAY, A-31, MANA MARG, LUCKNOW");
```

customer_id	customer_name	phone	payment_id	complete_address
BOB007	BOBBY DUDE	818016883	HDFC_VISA_120317	BOBBY, D-32, KAYAK MARG, MUMBAI
HEMA56	HEMA JAIN	888223110	PAYTM_300916	HEMA, A-2, OLD CITY, JAIPUR
JAMAL	JAY MALHOTRA	910292150	MOBI_210817	JAY, C-32, TILAK NAGAR, DELHI
JAY109	JAY MALHOTRA	982031154	COD_161111	JAY, A-31, MANA MARG, LUCKNOW
JAY12	JAY MALHOTRA	918236454	PAYPAL_VISA_211015	JAY, A-21, NIRMAN NAGAR, JAIPUR
KAM43	KAMAL JAIN	992031154	COD_040413	KAMAL, C-1, VAISHALI, KOTA
KAMAL	KAMAL JAIN	918276454	ICICI_VISA_100213	KAMAL, D-10, JAIL ROAD, DELHI
MANK1	MAYANK KUMAR	890706152	YES_VISA_311214	MAYANK, B-4, VAISHALI, KOTA
MAYANK67	MAYANK KUMAR	929011441	PAYTM_060112	MAYANK, B-3, MALL ROAD, HYDERABAD
NIN098	NINA MALIK	900131414	COD_100511	NINA, A-56, TAGORE NAGAR, JAIPUR
NULL	NULL	NULL	NULL	NULL

```sql
CREATE TABLE PRODUCT
(
  product_id VARCHAR(20) NOT NULL,
  order_id VARCHAR(20) NOT NULL,
  total_price INT(10) NOT NULL,
  quantity INT(5) NOT NULL,
  shipment_address VARCHAR(50) NOT NULL,
  product_name VARCHAR(20) NOT NULL,
  PRIMARY KEY (product_id)
);

insert into PRODUCT values("RING09", "REIF83", 4500, 2, "JAY, A-21, NIRMAN NAGAR, JAIPUR", "MAGNUM RING");
insert into PRODUCT values("NIKE30", "CH38", 6500, 3, "MAYANK, B-4, VAISHAI, KOTA", "NIKE SHOE");
insert into PRODUCT values("PUMA78", "CEW83B4", 1500, 2, "NINA, A-56, TAGORE NAGAR, JAIPUR", "PUMA SHOE");
insert into PRODUCT values("TOY67", "NJ069N", 500, 5, "JAY, C-32, TILAK NAGAR, DELHI", "TOY CAR");
insert into PRODUCT values("NOVA10", "BN895", 90, 1, "KAMAL, C-1, VAISHALI, KOTA", "NOVA PLAY");
insert into PRODUCT values("LENOVO32", "VH94", 2700, 1, "MAYANK, B-3, MALL ROAD, HYDERABAD", "LENOVO SCREEN");
insert into PRODUCT values("PEN90", "F3P8", 300, 30, "HEMA, A-2, OLD CITY, JAIPUR", "BALL PEN");
insert into PRODUCT values("HP_MOUSE27", "XM298", 250, 1,  "BOBBY, D-32, KAYAK MARG, MUMBAI", "HP MOUSE");
insert into PRODUCT values("XIAOMI8", "3FM0", 150, 1, "KAMAL, D-10, JAIL ROAD, DELHI", "XIAOMI SCREENGUARD");
insert into PRODUCT values("BOX2", "4GK89", 400, 2, "JAY, A-31, MANA MARG, LUCKNOW", "BLUE BOX");
```

product_id	order_id	total_price	quantity	shipment_address	product_name
BOX2	4GK89	400	2	JAY, A-31, MANA MARG, LUCKNOW	BLUE BOX
HP_MOUSE27	XM298	250	1	BOBBY, D-32, KAYAK MARG, MUMBAI	HP MOUSE
LENOVO32	VH94	2700	1	MAYANK, B-3, MALL ROAD, HYDERABAD	LENOVO SCREEN
NIKE30	CH38	6500	3	MAYANK, B-4, VAISHAI, KOTA	NIKE SHOE
NOVA10	BN895	90	1	KAMAL, C-1, VAISHALI, KOTA	NOVA PLAY
PEN90	F3P8	300	30	HEMA, A-2, OLD CITY, JAIPUR	BALL PEN
PUMA78	CEW83B4	1500	2	NINA, A-56, TAGORE NAGAR, JAIPUR	PUMA SHOE
RING09	REIF83	4500	2	JAY, A-21, NIRMAN NAGAR, JAIPUR	MAGNUM RING
TOY67	NJ069N	500	5	JAY, C-32, TILAK NAGAR, DELHI	TOY CAR
XIAOMI8	3FM0	150	1	KAMAL, D-10, JAIL ROAD, DELHI	XIAOMI SCREENGUARD
NULL	NULL	NULL	NULL	NULL	NULL

```sql
CREATE TABLE EMPLOYEE
(
  employee_id VARCHAR(20) NOT NULL,
  salary INT(6) NOT NULL,
  designation VARCHAR(20) NOT NULL,
  PRIMARY KEY (employee_id)
);

insert into EMPLOYEE values("MANOJ12", 4800, "SALES");
insert into EMPLOYEE values("KARAN55", 10000, "SALES MANAGER");
insert into EMPLOYEE values("MANAN", 2300, "POST BOY");
insert into EMPLOYEE values("SANJAY", 1500, "TEA BOY");
insert into EMPLOYEE values("CHARLIE", 50000, "BRANCH HEAD");
insert into EMPLOYEE values("KIRAN", 30000, "HR OFFICER");
insert into EMPLOYEE values("MANOJ", 10000, "BRANCH COORDINATOR");
insert into EMPLOYEE values("FARHAN", 20000, "ASSISTANT");
insert into EMPLOYEE values("VINAY", 4000, "SALES");
insert into EMPLOYEE values("SHAMAK", 5000, "SALES");
```

employee_id	salary	designation
CHARLIE	50000	BRANCH HEAD
FARHAN	20000	ASSISTANT
KARAN55	10000	SALES MANAGER
KIRAN	30000	HR OFFICER
MANAN	2300	POST BOY
MANOJ	10000	BRANCH COORDINATOR
MANOJ12	4800	SALES
SANJAY	1500	TEA BOY
SHAMAK	5000	SALES
VINAY	4000	SALES
NULL	NULL	NULL

```sql
CREATE TABLE ORDERS
(
    order_id VARCHAR(20) NOT NULL,
    customer_id VARCHAR(20) NOT NULL,
    placed_on DATE NOT NULL,
    price INT(10),
    employee_id VARCHAR(20) NOT NULL,
    PRIMARY KEY(order_id),
    FOREIGN KEY (customer_id) REFERENCES CUSTOMER (customer_id),
    FOREIGN KEY (employee_id) REFERENCES EMPLOYEE (employee_id)
);

insert into ORDERS values("REIF83", "JAY12", "2015-10-21", 4500, "MANOJ12");
insert into ORDERS values("CH38", "MANK1", "2014-12-31", 6500, "KARAN55");
insert into ORDERS values("CEW83B4", "NIN098", "2011-05-10", 1500, "MANOJ12");
insert into ORDERS values("NJ069N", "JAMAL", "2017-08-21", 500, "VINAY");
insert into ORDERS values("BN895", "KAM43", "2013-04-04", 90, "KARAN55");
insert into ORDERS values("VH94", "MAYANK67", "2012-01-06", 2700, "VINAY");
insert into ORDERS values("F3P8", "HEMA56", "2016-09-30", 300, "MANOJ12");
insert into ORDERS values("XM298", "BOB007", "2017-03-12", 250, "SHAMAK");
insert into ORDERS values("3FM0", "KAMAL", "2013-02-10", 150, "KARAN55");
insert into ORDERS values("4GK89", "JAY109", "2011-11-16", 400, "SHAMAK");
```

order_id	customer_id	placed_on	price	employee_id
3FM0	KAMAL	2013-02-10	150	KARAN55
4GK89	JAY109	2011-11-16	400	SHAMAK
BN895	KAM43	2013-04-04	90	KARAN55
CEW83B4	NIN098	2011-05-10	1500	MANOJ12
CH38	MANK1	2014-12-31	6500	KARAN55
F3P8	HEMA56	2016-09-30	300	MANOJ12
NJ069N	JAMAL	2017-08-21	500	VINAY
REIF83	JAY12	2015-10-21	4500	MANOJ12
VH94	MAYANK67	2012-01-06	2700	VINAY
XM298	BOB007	2017-03-12	250	SHAMAK
NULL	NULL	NULL	NULL	NULL

```sql
CREATE TABLE TRANSPORT
(
  mode_of_transport VARCHAR(20) NOT NULL,
  weight INT(4) NOT NULL,
  distance INT(5) NOT NULL,
  product_id VARCHAR(20) NOT NULL,
  PRIMARY KEY (mode_of_transport,product_id),
  FOREIGN KEY (product_id) REFERENCES PRODUCT (product_id)
);

insert into TRANSPORT values(" BY ROAD", 0.5, 50, "RING09");
insert into TRANSPORT values(" BY ROAD", 3, 98, "NIKE30");
insert into TRANSPORT values(" BY AIR", 2, 400, "PUMA78");
insert into TRANSPORT values(" BY AIR", 2.5, 230, "TOY67");
insert into TRANSPORT values(" BY ROAD", 1, 42, "NOVA10");
insert into TRANSPORT values(" BY AIR", 0.2, 2800, "LENOVO32");
insert into TRANSPORT values(" BY ROAD", 1, 50, "PEN90");
insert into TRANSPORT values(" BY AIR", 0.4, 100, "HP_MOUSE27");
insert into TRANSPORT values(" BY ROAD", 0.2, 20, "XIAOMI8");
insert into TRANSPORT values(" BY ROAD", 5, 5, "BOX2");
```

mode_of_transport	weight	distance	product_id
BY AIR	0	100	HP_MOUSE27
BY AIR	0	2800	LENOVO32
BY AIR	2	400	PUMA78
BY AIR	3	230	TOY67
BY ROAD	5	5	BOX2
BY ROAD	3	98	NIKE30
BY ROAD	1	42	NOVA10
BY ROAD	1	50	PEN90
BY ROAD	1	50	RING09
BY ROAD	0	20	XIAOMI8
NULL	NULL	NULL	NULL

```sql
CREATE TABLE BRANCHES_DECOMPOSED1
(
  branch_id VARCHAR(20) NOT NULL,
  branch_name VARCHAR(20) NOT NULL,
  branch_address VARCHAR(50) NOT NULL,
  PRIMARY KEY (branch_id)
);

CREATE TABLE BRANCHES_DECOMPOSED2
(
    employee_id VARCHAR(20) NOT NULL,
    branch_id VARCHAR(20) NOT NULL,
    branch_address VARCHAR(50) NOT NULL,
    PRIMARY KEY (employee_id, branch_id)
);
```

```sql
insert into BRANCHES_DECOMPOSED1 values("NCI34", "SOLICIT BUILDING", "KIRAN NAGAR, JAYPORE");
insert into BRANCHES_DECOMPOSED1 values("CN8490", "SATYAM TOWER", "SAHARA MARG, DELHI");
insert into BRANCHES_DECOMPOSED1 values("V59N4", "POST 31", "MONK ROAD, HYDERABAD");
insert into BRANCHES_DECOMPOSED1 values("F398", "NAYAK TOWER", "NAYAK ROAD, GANGTOK");
insert into BRANCHES_DECOMPOSED1 values("B4090", "SPEED", "NAMAN ROAD, MATHURA");
insert into BRANCHES_DECOMPOSED1 values("H50J", "VINAY TOWER", "JAPANESE ZONE, NEEMRANA");

insert into BRANCHES_DECOMPOSED2 values("CHARLIE", "NCI34", "KIRAN NAGAR, JAYPORE");
insert into BRANCHES_DECOMPOSED2 values("MANOJ", "CN8490", "SAHARA MARG, DELHI");
insert into BRANCHES_DECOMPOSED2 values("CHARLIE", "V59N4", "MONK ROAD, HYDERABAD");
insert into BRANCHES_DECOMPOSED2 values("FARHAN", "F398", "NAYAK ROAD, GANGTOK");
insert into BRANCHES_DECOMPOSED2 values("MANOJ", "B4090", "NAMAN ROAD, MATHURA");
insert into BRANCHES_DECOMPOSED2 values("MANOJ", "H50J", "JAPANESE ZONE, NEEMRANA");
```

branch_id	branch_name	branch_address
B4090	SPEED	NAMAN ROAD, MATHURA
CN8490	SATYAM TOWER	SAHARA MARG, DELHI
F398	NAYAK TOWER	NAYAK ROAD, GANGTOK
H50J	VINAY TOWER	JAPANESE ZONE, NEEMRANA
NCI34	SOLICIT BUILDING	KIRAN NAGAR, JAYPORE
V59N4	POST 31	MONK ROAD, HYDERABAD
NULL	NULL	NULL

employee_id	branch_id	branch_address
CHARLIE	NCI34	KIRAN NAGAR, JAYPORE
CHARLIE	V59N4	MONK ROAD, HYDERABAD
FARHAN	F398	NAYAK ROAD, GANGTOK
MANOJ	B4090	NAMAN ROAD, MATHURA
MANOJ	CN8490	SAHARA MARG, DELHI
MANOJ	H50J	JAPANESE ZONE, NEEMRANA
NULL	NULL	NULL

```
CREATE TABLE POST_BOY
(
  time_shift VARCHAR(20) NOT NULL,
  pb_name VARCHAR(20) NOT NULL,
  working_hours INT(2) NOT NULL,
  employee_id VARCHAR(20) NOT NULL,
  PRIMARY KEY (pb_name, employee_id),
  FOREIGN KEY (employee_id) REFERENCES EMPLOYEE (employee_id)
);

insert into POST_BOY values("9:00-15:00", "MANAN", 6, "MANAN");
insert into POST_BOY values("13:00-18:00", "VINAY", 5, "VINAY");
insert into POST_BOY values("10:00-18:00", "SHAMAK", 8, "SHAMAK");
insert into POST_BOY values("16:00-22:00", "MANOJ", 6, "MANOJ12");
insert into POST_BOY values("9:00-15:00", "SANJAY", 6, "SANJAY");
```

time_shift	pb_name	working_hours	employee_id
9:00-15:00	MANAN	6	MANAN
16:00-22:00	MANOJ	6	MANOJ12
9:00-15:00	SANJAY	6	SANJAY
10:00-18:00	SHAMAK	8	SHAMAK
13:00-18:00	VINAY	5	VINAY
NULL	NULL	NULL	NULL

```sql
CREATE TABLE SHIPMENT
(
  tracking_number VARCHAR(20) NOT NULL,
  expected_delivery_date DATE NOT NULL,
  PRIMARY KEY (tracking_number)
);

insert into SHIPMENT values("v98nvn0", "2015-10-31");
insert into SHIPMENT values("4v50540", "2015-01-10");
insert into SHIPMENT values("9bv434", "2011-05-20");
insert into SHIPMENT values("32f90b", "2017-08-30");
insert into SHIPMENT values("40ngn094", "2013-04-15");
insert into SHIPMENT values("vn034n4v", "2012-01-10");
insert into SHIPMENT values("n2309", "2016-10-05");
insert into SHIPMENT values("z23m239", "2017-03-15");
insert into SHIPMENT values("0239n", "2013-02-18");
insert into SHIPMENT values("m032c9", "2011-11-30");
```

tracking_number	expected_delivery_date
0239n	2013-02-18
32f90b	2017-08-30
40ngn094	2013-04-15
4v50540	2015-01-10
9bv434	2011-05-20
m032c9	2011-11-30
n2309	2016-10-05
v98nvn0	2015-10-31
vn034n4v	2012-01-10
z23m239	2017-03-15
NULL	NULL

```
CREATE TABLE Provides
(
  supplier_id VARCHAR(20) NOT NULL,
  product_id VARCHAR(20) NOT NULL,
  PRIMARY KEY (supplier_id, product_id),
  FOREIGN KEY (supplier_id) REFERENCES SUPPLIER_DECOMPOSED1(supplier_id),
  FOREIGN KEY (product_id) REFERENCES PRODUCT(product_id)
);

insert into Provides values("INFRACARE", "RING09");
insert into Provides values("SERVICER", "NIKE30");
insert into Provides values("CUREWO", "PUMA78");
insert into Provides values("BASE12", "TOY67");
insert into Provides values("OPERATE", "NOVA10");
insert into Provides values("MACHINE", "LENOVO32");
insert into Provides values("COMPUTERS", "PEN90");
insert into Provides values("ALLDEVICES", "HP_MOUSE27");
insert into Provides values("REPAIRER", "XIAOMI8");
insert into Provides values("LEATHERCARE", "BOX2");
```

supplier_id	product_id
LEATHERCARE	BOX2
ALLDEVICES	HP_MOUSE27
MACHINE	LENOVO32
SERVICER	NIKE30
OPERATE	NOVA10
COMPUTERS	PEN90
CUREWO	PUMA78
INFRACARE	RING09
BASE12	TOY67
REPAIRER	XIAOMI8
NULL	NULL

```sql
CREATE TABLE Maintains
(
  branch_id VARCHAR(20) NOT NULL,
  tracking_number VARCHAR(20) NOT NULL,
  PRIMARY KEY (branch_id, tracking_number),
  FOREIGN KEY (branch_id) REFERENCES BRANCHES_DECOMPOSED1(branch_id),
  FOREIGN KEY (tracking_number) REFERENCES SHIPMENT(tracking_number)
);

insert into Maintains values("NCI34", "v98nvn0");
insert into Maintains values("CN8490", "4v50540");
insert into Maintains values("V59N4", "9bv434");
insert into Maintains values("CN8490", "32f90b");
insert into Maintains values("F398", "40ngn094");
insert into Maintains values("B4090", "vn034n4v");
insert into Maintains values("B4090", "n2309");
insert into Maintains values("NCI34", "z23m239");
insert into Maintains values("F398", "0239n");
insert into Maintains values("H50J", "m032c9");
```

branch_id	tracking_number
F398	0239n
CN8490	32f90b
F398	40ngn094
CN8490	4v50540
V59N4	9bv434
H50J	m032c9
B4090	n2309
NCI34	v98nvn0
B4090	vn034n4v
NCI34	z23m239
NULL	NULL

```
CREATE TABLE Handled_by
(
  mode_of_transport VARCHAR(20) NOT NULL,
  pb_name VARCHAR(20) NOT NULL,
  product_id VARCHAR(20) NOT NULL,
  employee_id VARCHAR(20) NOT NULL,
  FOREIGN KEY (mode_of_transport,product_id) REFERENCES TRANSPORT(mode_of_transport,product_id),
  FOREIGN KEY (pb_name, employee_id) REFERENCES POST_BOY(pb_name, employee_id)
);

insert into Handled_by values(" BY ROAD", "MANAN", "RING09", "MANAN");
insert into Handled_by values(" BY ROAD", "SHAMAK", "NIKE30", "SHAMAK");
insert into Handled_by values(" BY AIR", "VINAY", "PUMA78", "VINAY");
insert into Handled_by values(" BY AIR", "MANAN", "TOY67", "MANAN");
insert into Handled_by values(" BY ROAD", "MANOJ", "NOVA10", "MANOJ12");
insert into Handled_by values(" BY AIR", "MANOJ", "LENOVO32", "MANOJ12");
insert into Handled_by values(" BY ROAD", "VINAY", "PEN90", "VINAY");
insert into Handled_by values(" BY AIR", "SANJAY", "HP_MOUSE27", "SANJAY");
insert into Handled_by values(" BY ROAD", "SANJAY", "XIAOMI8", "SANJAY");
insert into Handled_by values(" BY ROAD", "VINAY", "BOX2", "VINAY");
```

mode_of_transport	pb_name	product_id	employee_id
BY ROAD	MANAN	RING09	MANAN
BY ROAD	SHAMAK	NIKE30	SHAMAK
BY AIR	VINAY	PUMA78	VINAY
BY AIR	MANAN	TOY67	MANAN
BY ROAD	MANOJ	NOVA10	MANOJ12
BY AIR	MANOJ	LENOVO32	MANOJ12
BY ROAD	VINAY	PEN90	VINAY
BY AIR	SANJAY	HP_MOUSE27	SANJAY
BY ROAD	SANJAY	XIAOMI8	SANJAY
BY ROAD	VINAY	BOX2	VINAY
NULL	NULL	NULL	NULL

```
CREATE TABLE Exchanges_data_with
(
  branch_id VARCHAR(20) NOT NULL,
  pb_name VARCHAR(20) NOT NULL,
  employee_id VARCHAR(20) NOT NULL,
  FOREIGN KEY (branch_id) REFERENCES BRANCHES_DECOMPOSED1(branch_id),
  FOREIGN KEY (pb_name, employee_id) REFERENCES POST_BOY(pb_name, employee_id)
);

insert into Exchanges_data_with values("NCI34", "MANAN", "MANAN");
insert into Exchanges_data_with values("CN8490", "SHAMAK", "SHAMAK");
insert into Exchanges_data_with values("V59N4", "VINAY", "VINAY");
insert into Exchanges_data_with values("CN8490", "MANAN", "MANAN");
insert into Exchanges_data_with values("F398", "MANOJ", "MANOJ12");
insert into Exchanges_data_with values("B4090", "MANOJ", "MANOJ12");
insert into Exchanges_data_with values("B4090", "VINAY", "VINAY");
insert into Exchanges_data_with values("NCI34", "SANJAY", "SANJAY");
insert into Exchanges_data_with values("F398", "SANJAY", "SANJAY");
insert into Exchanges_data_with values("H50J", "VINAY", "VINAY");
```

branch_id	pb_name	employee_id
NCI34	MANAN	MANAN
CN8490	SHAMAK	SHAMAK
V59N4	VINAY	VINAY
CN8490	MANAN	MANAN
F398	MANOJ	MANOJ12
B4090	MANOJ	MANOJ12
B4090	VINAY	VINAY
NCI34	SANJAY	SANJAY
F398	SANJAY	SANJAY
H50J	VINAY	VINAY

```sql
CREATE TABLE SPECIAL_PACKAGE
(
  extra_charges INT(5) NOT NULL,
  gift_or_not INT NOT NULL,
  product_id VARCHAR(20) NOT NULL,
  FOREIGN KEY (product_id) REFERENCES PRODUCT(product_id),
  CONSTRAINT chk_gift CHECK (gift_or_not>=0 AND gift_or_not<=1)
);

insert into SPECIAL_PACKAGE values(10, 0, "RING09");
insert into SPECIAL_PACKAGE values(20, 1, "NIKE30");
insert into SPECIAL_PACKAGE values(17, 1, "PUMA78");
insert into SPECIAL_PACKAGE values(50, 0, "TOY67");
insert into SPECIAL_PACKAGE values(30, 0, "NOVA10");
insert into SPECIAL_PACKAGE values(10, 0, "LENOVO32");
insert into SPECIAL_PACKAGE values(40, 1, "PEN90");
insert into SPECIAL_PACKAGE values(72, 0, "HP_MOUSE27");
insert into SPECIAL_PACKAGE values(49, 0, "XIAOMI8");
insert into SPECIAL_PACKAGE values(15, 0, "BOX2");
```

extra_charges	gift_or_not	product_id
10	0	RING09
20	1	NIKE30
17	1	PUMA78
50	0	TOY67
30	0	NOVA10
10	0	LENOVO32
40	1	PEN90
72	0	HP_MOUSE27
49	0	XIAOMI8
15	0	BOX2

```
CREATE TABLE Assigns
(
  employee_id VARCHAR(20) NOT NULL,
  supplier_id VARCHAR(20) NOT NULL,
  FOREIGN KEY (employee_id) REFERENCES EMPLOYEE(employee_id),
  FOREIGN KEY (supplier_id) REFERENCES SUPPLIER_DECOMPOSED1(supplier_id)
);

insert into Assigns values("MANAN", "INFRACARE");
insert into Assigns values("SHAMAK", "SERVICER");
insert into Assigns values("VINAY", "CUREWO");
insert into Assigns values("MANAN", "BASE12");
insert into Assigns values("MANOJ12", "OPERATE");
insert into Assigns values("MANOJ12", "MACHINE");
insert into Assigns values("VINAY", "COMPUTERS");
insert into Assigns values("SANJAY", "ALLDEVICES");
insert into Assigns values("SANJAY", "REPAIRER");
insert into Assigns values("VINAY", "LEATHERCARE");
```

employee_id	supplier_id
MANAN	INFRACARE
SHAMAK	SERVICER
VINAY	CUREWO
MANAN	BASE12
MANOJ12	OPERATE
MANOJ12	MACHINE
VINAY	COMPUTERS
SANJAY	ALLDEVICES
SANJAY	REPAIRER
VINAY	LEATHERCARE
NULL	NULL

```
CREATE TABLE Delivers_to
(
  pb_name VARCHAR(20) NOT NULL,
  customer_id VARCHAR(20) NOT NULL,
  FOREIGN KEY (pb_name) REFERENCES POST_BOY(pb_name),
  FOREIGN KEY (customer_id) REFERENCES CUSTOMER(customer_id)
);

insert into Delivers_to values("MANAN", "JAY12");
insert into Delivers_to values("SHAMAK", "MANK1");
insert into Delivers_to values("VINAY", "NIN098");
insert into Delivers_to values("MANAN", "JAMAL");
insert into Delivers_to values("MANOJ", "KAM43");
insert into Delivers_to values("MANOJ", "MAYANK67");
insert into Delivers_to values("VINAY", "HEMA56");
insert into Delivers_to values("SANJAY", "BOB007");
insert into Delivers_to values("SANJAY", "KAMAL");
insert into Delivers_to values("VINAY", "JAY109");
```

pb_name	customer_id
MANAN	JAY12
SHAMAK	MANK1
VINAY	NIN098
MANAN	JAMAL
MANOJ	KAM43
MANOJ	MAYANK67
VINAY	HEMA56
SANJAY	BOB007
SANJAY	KAMAL
VINAY	JAY109
NULL	NULL