# DDL SCRIPT

```
CREATE SCHEMA Shopping_Mania;
SET SEARCH_PATH TO Shopping_Mania;

CREATE TABLE employee
(
        emp_id INTEGER,
        name VARCHAR (30) NOT NULL,
        shift VARCHAR (20),
        salary INTEGER,
        hire_date DATE,
        date_of_birth DATE,
        state VARCHAR (20),
        city VARCHAR (20),
        sid INTEGER NOT NULL,
        super_id INTEGER,
        PRIMARY KEY (emp_id),
        FOREIGN KEY (super_id) REFERENCES employee (emp_id)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE emp_email
(
        email_id VARCHAR (30),
        emp_id INTEGER,
        PRIMARY KEY (email_id, emp_id),
        FOREIGN KEY (emp_id) REFERENCES employee (emp_id)
        ON UPDATE CASCADE
        ON DELETE CASCADE
```

```
);

CREATE TABLE section
(
        sid INTEGER,
        sname VARCHAR (40) NOT NULL,
        mgr_id INTEGER NOT NULL,
        PRIMARY KEY (sid),
        FOREIGN KEY (mgr_id) REFERENCES employee (emp_id)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE customer
(
        cust_id INTEGER,
        cust_name VARCHAR (20),
        mobile_no BIGINT,
        payment_mode VARCHAR (20),
        PRIMARY KEY (cust_id)
);

CREATE TABLE supplier
(
        supplier_name VARCHAR (20),
        state VARCHAR (20),
        city VARCHAR (20),
        point_of_contact VARCHAR (20),
        PRIMARY KEY (supplier_name)
);
```

```
CREATE TABLE item
(
        item_code INTEGER,
        item_name VARCHAR (30) NOT NULL,
        stock INTEGER,
        selling_price INTEGER,
        sid INTEGER NOT NULL,
        PRIMARY KEY (item_code),
        FOREIGN KEY (sid) REFERENCES section (sid)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE bill
(
        bill_id INTEGER,
        cust_id INTEGER,
        date DATE,
        PRIMARY KEY(bill_id),
        FOREIGN KEY(cust_id) REFERENCES customer(cust_id)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE bill_details
(
        bill_id INTEGER,
        item_code INTEGER,
        quantity INTEGER,
        PRIMARY KEY(bill_id, item_code),
        FOREIGN KEY(item_code) REFERENCES item(item_code),
```

```
        FOREIGN KEY(bill_id) REFERENCES bill(bill_id)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE warranty
(
        warranty_id INTEGER,
        service_provider VARCHAR (40),
        warranty_period VARCHAR (20),
        item_code INTEGER NOT NULL,
        PRIMARY KEY (warranty_id),
        FOREIGN KEY (item_code) REFERENCES item (item_code)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE return_policy
(
        policy_id INTEGER,
        refund_percent INTEGER,
        item_code INTEGER NOT NULL,
        PRIMARY KEY (policy_id),
        FOREIGN KEY (item_code) REFERENCES item (item_code)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE offers
(
        item_code INTEGER NOT NULL,
```

```
        start_date DATE,
        end_date DATE,
        discount_percent INTEGER,
        PRIMARY KEY (item_code, start_date, end_date),
        FOREIGN KEY (item_code) REFERENCES item (item_code)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE supplier_email
(
        email_id VARCHAR (30),
        supplier_name VARCHAR (20),
        PRIMARY KEY (email_id, supplier_name),
        FOREIGN KEY (supplier_name) REFERENCES supplier (supplier_name)
        ON UPDATE CASCADE
        ON DELETE CASCADE
);

CREATE TABLE order_details
(
        order_id INTEGER,
        supplier_name VARCHAR(20),
        item_code INTEGER,
        quantity_ordered INTEGER,
        order_date DATE,
        arrival_date DATE,
        purchase_price INTEGER,
        PRIMARY KEY(order_id),
        FOREIGN KEY (item_code) REFERENCES item(item_code),
        FOREIGN KEY (supplier_name) REFERENCES supplier(supplier_name)
```

```
        ON UPDATE CASCADE
        ON DELETE CASCADE
);
```