Table 1: Product

- product\_id (Primary Key)

- product\_name

- id\_user\_bought(Foreign Key references User(user\_id))

- category\_id (Foreign Key references Category(category\_id))

- discounted\_price

- actual\_price

- discount\_percentage

- rating

- rating\_count

- about\_product

- vendor\_id (Foreign Key references Vendor(vendor\_id))

Table 2: Category

- category\_id (Primary Key)

- category\_name

Table 3: User

- user\_id (Primary Key)

- user\_name

- user\_gender

- user\_age

- user\_wishlist\_count

Table 4: Review

- review\_id (Primary Key)

- product\_id (Foreign Key references Product(product\_id))

- user\_id (Foreign Key references User(user\_id))

- review\_title

- review\_content

- review\_date

Table 5: Vendor

- vendor\_id (Primary Key)

- vendor\_name

- vendor\_loc

Table 6: Transaction

- transaction\_id (Primary Key)

- user\_id (Foreign Key references User(user\_id))

- product\_id (Foreign Key references Product(product\_id))

- purchase\_date

Table 7: PaymentMethod

- user\_id (Foreign Key references User(user\_id))

- payment\_method\_id (Primary Key)

- payment\_method\_name

Table 8: ProductImageLink

- product\_id (Primary Key, Foreign Key references Product(product\_id))

- img\_link

- product\_link

Table 9: StockStatus

- stock\_status

- product\_id (Primary Key, Foreign Key references Product(product\_id))

- stock\_quantity

Table 10: PaymentStatus

- transaction\_id (Foreign Key references Transaction(transaction\_id))

- payment\_status

SQL – QUERIES

**Table 1: Product**

CREATE TABLE Product (

product\_id VARCHAR(20) PRIMARY KEY,

product\_name VARCHAR(255) NOT NULL,

id\_user\_bought VARCHAR(25) REFERENCES user\_table(user\_id),

category\_id INT,

discounted\_price DECIMAL(10, 2),

actual\_price DECIMAL(10, 2),

discount\_percentage DECIMAL(5, 2),

rating DECIMAL(3, 2),

rating\_count INT,

about\_product TEXT,

vendor\_id VARCHAR(20),

stock\_status VARCHAR(255),

FOREIGN KEY (category\_id) REFERENCES Category(category\_id),

FOREIGN KEY (vendor\_id) REFERENCES Vendor(vendor\_id)

);

**Table 2: Category**

CREATE TABLE Category (

category\_id INT PRIMARY KEY,

category\_name VARCHAR(255) NOT NULL

);

### Table 3: User

CREATE TABLE User (

user\_id VARCHAR(30) PRIMARY KEY,

user\_name VARCHAR(255) NOT NULL,

user\_gender VARCHAR(10),

user\_age INT,

user\_wishlist\_count INT

);

### Table 4: Review

CREATE TABLE Review (

review\_id VARCHAR(30) PRIMARY KEY,

product\_id VARCHAR(20),

user\_id VARCHAR(30),

review\_title VARCHAR(255) NOT NULL,

review\_content TEXT,

review\_date DATE,

rating DECIMAL(3, 2),

FOREIGN KEY (product\_id) REFERENCES Product(product\_id),

FOREIGN KEY (user\_id) REFERENCES User(user\_id)

);

### Table 5: Vendor

CREATE TABLE Vendor (

vendor\_id VARCHAR(20) PRIMARY KEY,

vendor\_name VARCHAR(255) NOT NULL,

vendor\_loc VARCHAR(255)

);

### Table 6: Transaction

CREATE TABLE Transaction (

transaction\_id VARCHAR(30) PRIMARY KEY,

user\_id VARCHAR(30),

product\_id VARCHAR(20),

purchase\_date DATE,

FOREIGN KEY (user\_id) REFERENCES User(user\_id),

FOREIGN KEY (product\_id) REFERENCES Product(product\_id)

);

### Table 7: PaymentMethod

CREATE TABLE PaymentMethod (

user\_id VARCHAR(30),

payment\_method\_id VARCHAR(20) PRIMARY KEY,

payment\_method\_name VARCHAR(255) NOT NULL,

FOREIGN KEY (user\_id) REFERENCES User(user\_id),

);

### Table 8: ProductImageLink

CREATE TABLE ProductImageLink (

product\_id VARCHAR(20) PRIMARY KEY,

img\_link VARCHAR(255),

product\_link VARCHAR(255),

FOREIGN KEY (product\_id) REFERENCES Product(product\_id)

);

### Table 9: StockStatus

CREATE TABLE StockStatus (

product\_id VARCHAR(20) PRIMARY KEY,

stock\_status VARCHAR(255) NOT NULL,

stock\_quantity INT NOT NULL,

FOREIGN KEY (product\_id) REFERENCES Product(product\_id)

);

### Table 10: PaymentStatus

CREATE TABLE PaymentStatus (

transaction\_id VARCHAR(30) PRIMARY KEY,

payment\_status VARCHAR(255) NOT NULL,

FOREIGN KEY (transaction\_id) REFERENCES Transaction(transaction\_id)

);

**Product – Category:**

Each product belongs to one category.

Each category can have at least one/multiple products.

**Product - Vendor:**

Each product is supplied by one/many vendors.

Each vendor can supply multiple products.

**Product - User:**

Each product can be purchased by multiple users.

Each user can purchase multiple products.

**Product - StockStatus:**

Each product has one stock status.

(Optional) Each stock status is associated with one product.

**User - Review:**

Each user can write ZERO to multiple reviews.

**Transaction - User:**

Each user can have ZERO to multiple transactions.

**Transaction - PaymentMethod:**

Each transaction is associated with one payment method.

Each payment method can be associated with multiple transactions.

**Transaction – payment status:**

Each transaction has one payment status.

**PaymentMethod - PaymentStatus:**

Each payment method has one default payment status. (1)

Each payment status can be associated with multiple payment methods. (M)

**Product(product\_id, product\_name, id\_user\_bought, category\_id, discounted\_price, actual\_price, discount\_percentage, rating, rating\_count, about\_product, vendor\_id)**

* product\_id: Primary key and NOT NULL
* id\_user\_bought: Foreign key (references User(user\_id))
* category\_id: Foreign key (references Category(category\_id))
* vendor\_id: Foreign key (references Vendor(vendor\_id))

**Category(category\_id, category\_name)**

* category\_id: Primary key and NOT NULL

**User(user\_id, user\_name, user\_gender, user\_age, user\_wishlist\_count)**

* user\_id: Primary key and NOT NULL

**Review(review\_id, product\_id, user\_id, review\_title, review\_content, review\_date)**

* review\_id: Primary key and NOT NULL
* product\_id: Foreign key (references Product(product\_id))
* user\_id: Foreign key (references User(user\_id))

**Vendor(vendor\_id, vendor\_name, vendor\_loc)**

* vendor\_id: Primary key and NOT NULL
* vendor\_name: NOT NULL
* vendor\_loc: NOT NULL

**Transaction(transaction\_id, user\_id, product\_id, purchase\_date)**

* transaction\_id: Primary key and NOT NULL
* user\_id: Foreign key (references User(user\_id))
* product\_id: Foreign key (references Product(product\_id))

**PaymentMethod(user\_id, payment\_method\_id, payment\_method\_name)**

* user\_id: Foreign key (references User(user\_id))
* payment\_method\_id: Primary key and NOT NULL

**ProductImageLink(product\_id, img\_link, product\_link)**

* product\_id: Primary key and Foreign key (references Product(product\_id))

**StockStatus(stock\_status, product\_id, stock\_quantity)**

* product\_id: Primary key and Foreign key (references Product(product\_id))

**PaymentStatus(transaction\_id, payment\_status)**

* transaction\_id: Foreign key (references Transaction(transaction\_id))