E-commerce

Entities

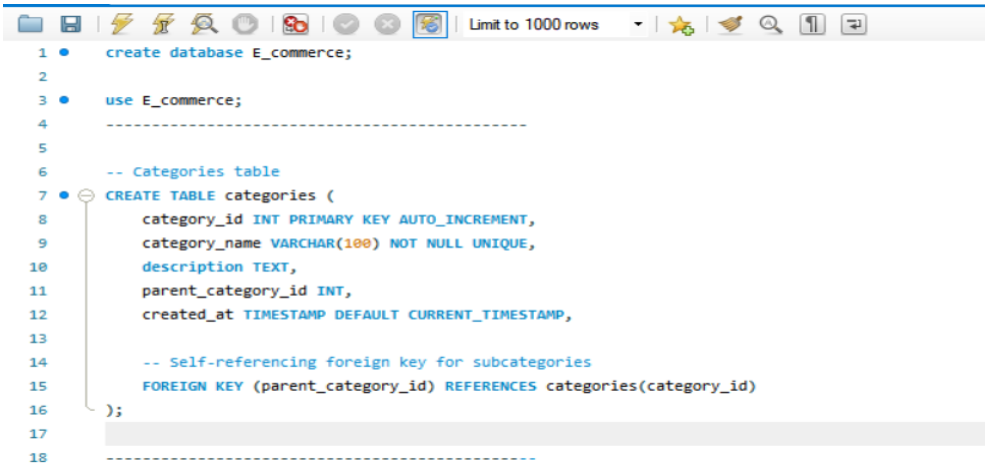
- Users → Customers who register and shop.
- Addresses → Each user can have multiple shipping/billing addresses.
- Categories → Organize products into groups and subgroups.
- Products → Items available for purchase.
- Orders → Transactions when a user buys something.
- Order Items → Specific products included in an order.
- Payments → Payment details for an order.
- Reviews → Feedback from users about products.

Relationships

- A user can have many addresses.
- A category can have subcategories (self-referencing).
- A category can contain many products.
- A user can place many orders.
- An order contains many products (via order_items).
- An order has one or more payments.
- A user can leave many reviews on products.

Tables we used :

Categories – it defined types of categories are there



Output :

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content

category_id	category_name	description	parent_category_id	created_at
1	Electronics	Devices and gadgets	NULL	2025-09-22 15:05:22
2	Fashion	Clothing and accessories	NULL	2025-09-22 15:05:22
3	Home Appliances	Appliances for home and kitchen	NULL	2025-09-22 15:05:22
4	Books	Educational and entertainment books	NULL	2025-09-22 15:05:22
5	Sports	Sports gear and equipment	NULL	2025-09-22 15:05:22
6	Mobiles	Smartphones and mobile phones	1	2025-09-22 15:05:22
7	Laptops	Personal and gaming laptops	1	2025-09-22 15:05:22
8	Cameras	DSLR and digital cameras	1	2025-09-22 15:05:22
9	Men Clothing	Apparel for men	2	2025-09-22 15:05:22
10	Women Clothing	Apparel for women	2	2025-09-22 15:05:22
11	Accessories	Fashion accessories	2	2025-09-22 15:05:22
12	Kitchen Appliances	Appliances for cooking and food stor...	3	2025-09-22 15:05:22
13	Cleaning Applia...	Vacuum, washing machines, etc.	3	2025-09-22 15:05:22
14	Fiction	Novels and stories	4	2025-09-22 15:05:22
15	Non-Fiction	Knowledge and learning	4	2025-09-22 15:05:22
16	Indoor Sports	Board games, chess, etc.	5	2025-09-22 15:05:22
17	Outdoor Sports	Football, cricket, etc.	5	2025-09-22 15:05:22
NULL	NULL	NULL	NULL	NULL

Users :

```sql
-- Users table
CREATE TABLE users (
    user_id INT PRIMARY KEY AUTO_INCREMENT,
    username VARCHAR(50) NOT NULL UNIQUE,
    email VARCHAR(100) NOT NULL UNIQUE,
    password_hash VARCHAR(255) NOT NULL,
    first_name VARCHAR(50) NOT NULL,
    last_name VARCHAR(50) NOT NULL,
    phone VARCHAR(20),
    date_of_birth DATE,
    registration_date TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    is_active BOOLEAN DEFAULT TRUE,

    -- Indexes for performance
    INDEX idx_email (email),
    INDEX idx_username (username)
);
```

Output ;

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

user_id	username	email	password_hash	first_name	last_name	phone	date_of_birth	registration_date	is_active
1	Hari	hari@example.com	hashed_password1	Hari	Macha	9876543210	1990-05-12	2025-09-22 15:05:22	1
2	Suresh	suresh@example.com	hashed_password2	Suresh	Thota	9876501234	1992-08-20	2025-09-22 15:05:22	1
3	Manoj	Manoj@example.com	hashed_password3	Manoj	Jinka	9876505678	1988-11-10	2025-09-22 15:05:22	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Addresses Table :

```sql
-- Addresses table
CREATE TABLE addresses (
    address_id INT PRIMARY KEY AUTO_INCREMENT,
    user_id INT NOT NULL,
    address_type ENUM('billing', 'shipping', 'both') DEFAULT 'shipping',
    street_address VARCHAR(200) NOT NULL,
    city VARCHAR(50) NOT NULL,
    state VARCHAR(50) NOT NULL,
    postal_code VARCHAR(20) NOT NULL,
    country VARCHAR(50) NOT NULL,
    is_default BOOLEAN DEFAULT FALSE,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    -- Foreign key relationship
    FOREIGN KEY (user_id) REFERENCES users(user_id) ON DELETE CASCADE
);
```

Output :

```sql
-- addresses
INSERT INTO addresses (user_id, address_type, street_address, city, state, postal_code, country, is_default)
VALUES
(1, 'shipping', 'proddatur , kesava nagar', 'kadapa', 'PDTR', '516360', 'INDIA', TRUE),
(1, 'billing', '', 'proddatur , kesava nagar', 'kadapa', '516360', 'INDIA', TRUE),
(2, 'shipping', 'sri ram nagar colony', 'Kadapa', 'PDTR', '90001', 'INDIA', TRUE),
(3, 'shipping', 'Sai baba street', 'Chittoor', 'PILER', '517217', 'INDIA', TRUE);
```

Products Table :

```sql
-- Products table
CREATE TABLE products (
    product_id INT PRIMARY KEY AUTO_INCREMENT,
    product_name VARCHAR(200) NOT NULL,
    description TEXT,
    category_id INT NOT NULL,
    price DECIMAL(10, 2) NOT NULL,
    cost DECIMAL(10, 2),
    sku VARCHAR(50) UNIQUE NOT NULL,
    stock_quantity INT NOT NULL DEFAULT 0,
    weight DECIMAL(8, 2),
    dimensions VARCHAR(50),
    brand VARCHAR(100),
    is_active BOOLEAN DEFAULT TRUE,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,

    -- Foreign key relationship
    FOREIGN KEY (category_id) REFERENCES categories(category_id),

    -- Indexes for performance
    INDEX idx_sku (sku),
    INDEX idx_category (category_id),
    INDEX idx_price (price)
```

Output:

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

product_id	product_name	description	category_id	price	cost	sku	stock_quantity	weight	dimensions	brand	is_active
1	iPhone 15	Latest Apple iPhone	2	999.99	700.00	SKU1001	50	NULL	NULL	Apple	1
2	Samsung Galaxy S24	Newest Samsung flagship phone	2	899.99	650.00	SKU1002	40	NULL	NULL	Samsung	1
3	Dell XPS 15	High performance laptop	3	1499.99	1000.00	SKU2001	30	NULL	NULL	Dell	1
4	Nike Air Max	Comfortable running shoes	4	120.00	60.00	SKU3001	100	NULL	NULL	Nike	1
5	LG Washing Machine	Front load washing machine	5	550.00	400.00	SKU4001	20	NULL	NULL	LG	1
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

created_at	updated_at
2025-09-22 15:05:22	2025-09-22 15:05:22
2025-09-22 15:05:22	2025-09-22 15:05:22
2025-09-22 15:05:22	2025-09-22 15:05:22
2025-09-22 15:05:22	2025-09-22 15:05:22
2025-09-22 15:05:22	2025-09-22 15:05:22
NULL	NULL

Orders table :

```sql
-- Order Items table
CREATE TABLE order_items (
    order_item_id INT PRIMARY KEY AUTO_INCREMENT,
    order_id INT NOT NULL,
    product_id INT NOT NULL,
    quantity INT NOT NULL DEFAULT 1,
    price DECIMAL(10, 2) NOT NULL, -- price at the time of purchase
    discount DECIMAL(10, 2) DEFAULT 0,
    subtotal DECIMAL(10, 2) GENERATED ALWAYS AS ((quantity * price) - discount) STORED,

    -- Foreign key relationships
    FOREIGN KEY (order_id) REFERENCES orders(order_id) ON DELETE CASCADE,
    FOREIGN KEY (product_id) REFERENCES products(product_id),

    INDEX idx_order (order_id),
    INDEX idx_product (product_id)
);
```

Output :

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

order_item_id	order_id	product_id	quantity	price	discount	subtotal
1	1	1	1	999.99	0.00	999.99
2	1	4	1	120.00	20.00	100.00
3	2	3	1	1499.99	0.00	1499.99
4	3	4	1	120.00	0.00	120.00
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Payments table ;

```
CREATE TABLE payments (
    payment_id INT PRIMARY KEY AUTO_INCREMENT,
    order_id INT NOT NULL,
    amount DECIMAL(10, 2) NOT NULL,
    payment_method ENUM('credit_card', 'debit_card', 'paypal', 'bank_transfer', 'cash_on_delivery'),
    payment_status ENUM('pending', 'completed', 'failed', 'refunded') DEFAULT 'pending',
    transaction_id VARCHAR(100) UNIQUE,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (order_id) REFERENCES orders(order_id) ON DELETE CASCADE,
    INDEX idx_order_payment (order_id),
    INDEX idx_payment_status (payment_status)
);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

payment_id	order_id	amount	payment_method	payment_status	transaction_id	created_at
1	1	1099.99	credit_card	com…	…12345	2025-09-22 15:05:22
2	2	1550.00	paypal	completed	TXN12346	2025-09-22 15:05:22
3	3	120.00	cash_on_delivery	pending	NULL	2025-09-22 15:05:22
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Edit current row

Review table :

```
CREATE TABLE reviews (
    review_id INT PRIMARY KEY AUTO_INCREMENT,
    product_id INT NOT NULL,
    user_id INT NOT NULL,
    rating INT CHECK (rating BETWEEN 1 AND 5),
    review_text TEXT,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (product_id) REFERENCES products(product_id) ON DELETE CASCADE,
    FOREIGN KEY (user_id) REFERENCES users(user_id) ON DELETE CASCADE,

    INDEX idx_product_review (product_id),
    INDEX idx_user_review (user_id)
);
```

Output :

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

review_id	product_id	user_id	rating	review_text	created_at
1	1	1	5	Amazing phone, worth the price!	2025-09-22 15:05:22
2	2	2	4	Great performance but battery could be better.	2025-09-22 15:05:22
3	3	3	5	Excellent laptop for work and gaming.	2025-09-22 15:05:22
4	4	1	4	Comfortable shoes, but slightly expensive.	2025-09-22 15:05:22
NULL	NULL	NULL	NULL	NULL	NULL

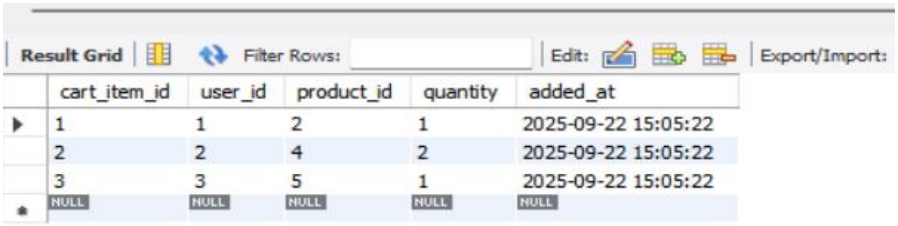
Cart items Table :

```
CREATE TABLE cart_items (
    cart_item_id INT PRIMARY KEY AUTO_INCREMENT,
    user_id INT NOT NULL,
    product_id INT NOT NULL,
    quantity INT DEFAULT 1,
    added_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,

    FOREIGN KEY (user_id) REFERENCES users(user_id) ON DELETE CASCADE,
    FOREIGN KEY (product_id) REFERENCES products(product_id),

    UNIQUE (user_id, product_id)
);
```

Output :



cart_item_id	user_id	product_id	quantity	added_at
1	1	2	1	2025-09-22 15:05:22
2	2	4	2	2025-09-22 15:05:22
3	3	5	1	2025-09-22 15:05:22
NULL	NULL	NULL	NULL	NULL

Foreign kay and primary key :

Primary Key (PK):
A primary key is a unique identifier for each record in a table. It ensures that every row can be uniquely identified. In your database schema, each table has a primary key that usually is an auto-increment integer, such as:

- category_id in the categories table
- user_id in the users table
- address_id in the addresses table
- product_id in the products table
- order_id in the orders table
- order_item_id in the order_items table
- payment_id in the payments table
- review_id in the reviews table
- cart_item_id in the cart_items table

The primary key uniquely identifies each record and ensures no duplicate entries for that key column.

Foreign Key (FK):
A foreign key is a column (or set of columns) in one table that refers to the primary key in another table. It establishes a relationship between the two tables and ensures referential integrity—meaning the foreign key value must exist as a primary key value in the referenced table or be null if allowed. In your schema, examples include:

- parent_category_id in categories references category_id in the same categories table (self-referencing for category hierarchy).
- user_id in addresses references user_id in users.
- category_id in products references category_id in categories.
- user_id in orders references user_id in users.
- shipping_address_id and billing_address_id in orders reference address_id in addresses.
- order_id in order_items references order_id in orders.
- product_id in order_items references product_id in products.
- order_id in payments references order_id in orders.
- product_id and user_id in reviews reference product_id in products and user_id in users respectively.
- user_id and product_id in cart_items reference user_id in users and product_id in products.

In summary:

- **Primary Key:** Uniquely identifies each row within its own table.
- **Foreign Key:** Links a column in one table to the primary key of another (or the same) table to create relational integrity.

This guarantees consistency and enforces correct relationships between entities in the e-commerce system database.

Server : MySQL

E-R diagram :