**Creating a Shopping App Using Python**

**Introduction:** In this project, we aim to develop a shopping application or e-commerce platform using Python. The application will feature both user and admin functionalities, including login systems, product catalogue management, cart management, and payment options. The focus will be solely on backend implementation, excluding UX/UI design and database connectivity.

**Problem Scenario:** The project requires the development of a shopping application that incorporates essential features such as user login, admin login, product categories, cart management, and payment options. The implementation will involve creating a welcome message, setting up user and admin authentication, constructing a product catalogue with various categories, enabling users to view and manage their carts, providing payment checkout options, and allowing admin users to manage products and categories.

**Project Implementation:**

1. **Welcome Message:**
   * A welcoming message, "Welcome to the Demo Marketplace, dhukan.com" will be displayed to users upon accessing the application.
2. **User and Admin Authentication:**
   * User and admin login systems will be established with demo databases for verification and session ID generation.
   * Users and admins will have distinct login credentials to access their respective functionalities.
3. **Product Catalogue Management:**
   * A sample product catalogue will be created, consisting of three to four categories such as footwear, clothing, and electronics.
   * Each product will have attributes including product ID, name, category ID, and price.
4. **User Functionalities:**
   * Users will be able to view the product catalogue, add items to their cart, remove items from the cart, and view cart contents.
   * Upon selecting items and quantities, users can proceed to checkout and choose from demo payment options like Net banking, PayPal, or UPI.
5. **Admin Functionalities:**
   * Admins will have exclusive access to functionalities such as adding new products, modifying existing products, and removing products from the catalogue.
   * Additionally, admins can add or delete product categories to adapt to dynamic market demands.

**Project Structure:** The project will be structured around classes representing users and admins, with methods corresponding to their respective functionalities. The main script will orchestrate the application's logic, handling user interactions and database operations.

#Class User to hold Cart and User Credentials

class User:

#Class Product to hold the Product info

class Product:

Class Catalogue holds List of products and Categories

class Catalog:

#Class ECommerceApp to perform operations on Demo Market place

class ECommerceApp:

**Conclusion:** By implementing the outlined functionalities and adhering to the provided guidelines, the project aims to deliver a functional shopping application using Python. The separation of user and admin roles ensures secure access to relevant features, while the inclusion of essential functionalities caters to the needs of both consumers and administrators in a simulated e-commerce environment.