

# shopping-cart-3

August 19, 2024

Situation: You are a Python developer, building a shopping application or e-commerce application with login and public login features on the Python platform. The application should include categories such as footwear, clothing, and electronics. You should be able to add and update categories in the application. Additionally, the application must allow users to add or remove items from their cart. Finally, the program needs to support various payment options including UPI and debit cards. This is a backend implementation, and UX/UI and database connectivity are not required.

Task • Display a welcome message in the e-commerce application, such as “Welcome to the Demo Marketplace” • Create user login and admin login after writing the code for the welcome message. Create demo databases for “user” and “admin” verification and “session ID” creation. Once the databases are created, create admin and user logins • Create a sample product catalog with three to four product categories like Boots, Coats, Jackets, and Caps. The dummy database of the catalog should contain product ID, name, category ID, and price for each item. Both users and administrators can view the catalog • User login rights should include the ability to view cart contents, add items to the cart, and remove items from the cart. Users should be able to add or delete items in the cart using session ID, product ID, and quantity • Provide demo payment checkout options such as Net banking and PayPal. After selecting the payment option, display a checkout message like “Your order is successfully placed” or “You will be shortly redirected to the portal to make a payment of \$20”. • Only the admin can log in using their credentials. If the admin tries to log in using another set of credentials, an error notice must appear • The admin should not be able to perform the same functions as the user. An error should appear if the admin tries to carry out those tasks • The admin should be able to add new products to the catalog using the selected attributes. The program should also allow the admin to modify existing products using an admin session ID • The admin should have the ability to remove any existing products from the catalog • Finally, the admin should be able to add a new category of product and delete existing categories from the catalog, considering the dynamic demands of the market • Users should not have access to the ad

Action 1. Open the Python environment: • Open the Python File and start your work in a blank code file. 2. Print the welcome message: • Write the code `print(“Welcome to the Demo Marketplace”)` in your file to display the welcome message 3. Define the user and admin databases: • Create two dictionaries named `user_db` and `admin_db` to store the usernames and passwords 4. Define the user login function: • Write a function called `user_login` • Inside the function, use the input function to get the user’s username and password • Check if the entered username exists in the `user_db` and if the password matches • Print “Login successful as user” or “Login failed” accordingly 5. Define the admin login function: • Write a function called `admin_login` similar to the user login function. • Get the admin’s username and password • Verify the credentials against the `admin_db` • Print login status 6. Define the session ID generator function: • Create a function named `generate_session_id` • Inside the function, simply return a fixed session ID like “12345” 7. Add the functionality to login as a user or an admin: • Use the input function to ask the user whether they want to log in as a user or admin • Based on the input, call either the

user\_login or admin\_login function 8. Create the product catalog: • Define a list called catalog that contains dictionaries with details for each product • Each dictionary should have keys like “id”, “name”, “category”, and “price” 9. Define the catalog display function: • Write a function named display\_catalog • Loop through the catalog list and print product details in a tabular format 10. Create the cart management functionalities: • Create an empty list named cart to represent the user’s cart • Define functions display\_cart, add\_to\_cart, and delete\_from\_cart • display\_cart: Display the items in the cart • add\_to\_cart: Add items to the cart with session ID, product ID, and quantity • delete\_from\_cart: Remove items from the cart using session ID and product ID 11. Create a user interaction loop: • Start a loop to repeatedly interact with the user • Show options to view cart, add to cart, delete from cart, or log out • Use the input to get the user’s choice • Depending on the choice, call the corresponding functions defined earlier 12. Implement the checkout functionality: • Define a function called checkout • Ask the user to select a payment method using the input function • Based on the payment method, print a message indicating successful payment or redirection for payment 13. Implement the admin login verification functionality: • Create a function named admin\_login to verify admin credentials. • Check if the entered admin username exists in admin\_db and if the password matches. • Implement the admin verification functionality when the user enters the login credentials. 14. Define a function named admin\_func: • Display options for admin tasks like adding products or updating products. • Use the input function to obtain the admin’s selection. • Depending on the choice, call functions to perform admin actions. • Add restrictions to ensure admin actions are not accessible to normal users. • In the functions related to cart management and catalog management, add conditions to check if the user type is “admin” before proceeding. 15. Differentiate user and admin experience: • Inside the user interaction loop, ask if the user is an admin or a regular user. • Use conditional statements to direct the user to the appropriate set of actions. 16. Run the program and verify the results: • Run the program by executing the file in your Python environment. • Follow the prompts and inputs to simulate user interactions and admin functions/min’s rights as described above

```
[ ]: #user and admin db
user_db={"user1":"user1pass", "user2":"user2pass"}
admin_db={"admin1":"admin1pass", "admin2":"admin2pass"}

def generate_session_id():
    sid=21874
    return sid

def display_cart(cart):
    print("\t\t\tCART")
    print("-----")
    if len(cart)==0:
        print("Cart is Empty \n")
    else:
        print("SESSION-ID\t\t PRODUCT-ID\t\t QUANTITY")
        for prod in cart:
            print(f"{prod['session_id']}\t\t\t {prod['product_id']}\t\t\t {prod['quantity']}")
```

```

def add_to_cart(cart):
    s_id=generate_session_id()
    p_id=int(input("Enter product id :"))
    for item in cart:
        if item["product_id"]==p_id:
            qty=int(input("Enter quantity"))
            item["quantity"]+=qty
        return
    qty=int(input("Enter quantity"))
    cart.append({"session_id":s_id,"product_id":p_id,"quantity":qty})

def delete_from_cart(cart):
    s_id=generate_session_id()
    p_id=int(input("Enter product id :"))
    for item in cart:
        if item["session_id"]==s_id and item["product_id"]==p_id:
            cart.remove(item)

def checkout(cart):
    if len(cart)==0:
        print("Cart is Empty \n")
    else:
        print("\t\tCHECKOUT")
        print("-----")
        print("\tSelect Payment Method")
        print("\t1.Net Banking 2.PayPal")
        check=int(input("\tSelect the checkout method: "))
        if check==1:
            print("\tNet Banking successfull")
            cart=[]
            return cart
        elif check==2:
            print("\tPayPal payment successfull")
            cart=[]
            return cart
        else:
            print("\tPayment Failed...Try Again")

Catalog=[{"id":123, "name":"Crocs", "category":"footwear", "price":2000},
          {"id":131, "name":"Clock", "category":"electronics", "price":32000},
          {"id":145, "name":"Tshirt", "category":"clothing", "price":800},
          {"id":124, "name":"Nike", "category":"footwear", "price":3500},
          {"id":138, "name":"Phone", "category":"electronics", "price":21000},
          {"id":140, "name":"Jeans", "category":"clothing", "price":7000}]

def display_catalog():

```

```

print("ID\t NAME\t\t CATEGORY\t\t PRICE")
for prod in Catalog:
    print(f"{prod['id']}\t {prod['name']}\t\t {prod['category']}\t\t {prod['price']}")

def update_catalog():
    print("1.Add new product 2. Delete a product 3. Change product price")
    print("\n")
    upr=int(input("select one from above"))
    if upr==1:
        prid=int(input("Enter product id : "))
        prname=input("Enter product name: ")
        prcat=input("Enter product category: ")
        prp=int(input("Enter product price: "))
        Catalog.append({"id":prid,"name":prname,"category":prcat,"price":prp})
    elif upr==2:
        delid=int(input("Enter product id : "))
        for prod in Catalog:
            if prod["id"]==delid:
                Catalog.remove(prod)
    elif upr==3:
        upid=int(input("Enter product id : "))
        for prod in Catalog:
            if prod["id"]==upid:
                am=int(input("Enter new price: "))
                prod["price"]=am
    else:
        print("Incorrect option")

def admin_func():
    print("OPTIONS")
    print("\n1.View Catalog\n2.Update Catalog\n3.Log Out")
    opp=int(input("Select Option : "))
    if opp==1:
        print("\n")
        display_catalog()
        admin_func()
    elif opp==2:
        update_catalog()
        admin_func()
    elif opp==3:
        exit()
    else:
        print("\n\n Invalid Option. Select from the Listed!!! \n\n")
        admin_func()

```

```

def user_login():
    uname=input(f"Enter Username: ")
    upass=input(f"Enter password for {uname} : ")
    if uname in user_db.keys() and user_db[uname] == upass:
        print("Login successful as user")
        cart=[]
        while True:
            print("\t\t\tHOME")
            print("-----")
            display_catalog()
            print("OPTIONS")
            print("\n1.View Cart\n2.Add to Cart\n3.Delete from Cart\n4.Log_
↪Out\n5.Checkout")
            op=int(input("Select Option : "))
            if op==1:
                print("\n")
                display_cart(cart)
            elif op==2:
                add_to_cart(cart)
            elif op==3:
                delete_from_cart(cart)
            elif op==4:
                exit()
            elif op==5:
                cart=checkout(cart)
            else:
                print("\n\n\t Invalid Option. Select from the Listed !!! \n\n")
        else:
            print("Login Failed")

def admin_login():
    aname=input(f"Enter Username: ")
    apass=input(f"Enter password for {aname} : ")
    if aname in admin_db.keys() and admin_db[aname] == apass:
        print("\tLogin successful as Admin")
        admin_func()
    else:
        print("\tLogin Failed")

print("\t\tWelcome to the Demo Marketplace")
print("-----")
print("\t\t\tLOGIN")
print("\t 1.Login as User\t 2.Login as Admin")
ch=int(input("\nEnter your choice: "))
if ch==1:
    user_login()

```

```

elif ch==2:
    admin_login()

else:
    print("\n \t\tInvalid Choice")

```

```

[ ]: """
                                Welcome to the Demo Marketplace
-----

                                LOGIN
                1.Login as User          2.Login as Admin

Enter your choice:  1
Enter Username:  user1
Enter password for user1 :  user1pass
Login successful as user

                                HOME
-----

ID          NAME          CATEGORY          PRICE
123         Crocs         footwear         2000
131         Clock         electronics      32000
145         Tshirt        clothing         800
124         Nike          footwear         3500
138         Phone         electronics      21000
140         Jeans         clothing         7000
OPTIONS

1.View Cart
2.Add to Cart
3.Delete from Cart
4.Log Out
5.Checkout
Select Option :  1

                                CART
-----

Cart is Empty

```

```

[ ]: """

                                HOME
-----

ID          NAME          CATEGORY          PRICE
123         Crocs         footwear         2000
131         Clock         electronics      32000
145         Tshirt        clothing         800
124         Nike          footwear         3500
138         Phone         electronics      21000

```

140            Jeans                    clothing                    7000  
OPTIONS

1.View Cart  
2.Add to Cart  
3.Delete from Cart  
4.Log Out  
5.Checkout  
Select Option : 2  
Enter product id : 124  
Enter quantity 3

HOME

```
-----  
ID          NAME          CATEGORY          PRICE  
123         Crocs         footwear         2000  
131         Clock         electronics      32000  
145         Tshirt        clothing         800  
124         Nike          footwear        3500  
138         Phone         electronics     21000  
140         Jeans         clothing        7000  
OPTIONS
```

1.View Cart  
2.Add to Cart  
3.Delete from Cart  
4.Log Out  
5.Checkout  
Select Option : 1

CART

```
-----  
SESSION-ID          PRODUCT-ID          QUANTITY  
21874                124                3
```

HOME

```
-----  
ID          NAME          CATEGORY          PRICE  
123         Crocs         footwear         2000  
131         Clock         electronics      32000  
145         Tshirt        clothing         800  
124         Nike          footwear        3500  
138         Phone         electronics     21000  
140         Jeans         clothing        7000  
OPTIONS
```

1.View Cart  
2.Add to Cart

```

3.Delete from Cart
4.Log Out
5.Checkout
Select Option : 2
Enter product id : 138
Enter quantity 5

```

HOME

```

-----
ID          NAME          CATEGORY          PRICE
123         Crocs          footwear          2000
131         Clock          electronics        32000
145         Tshirt         clothing           800
124         Nike           footwear          3500
138         Phone          electronics        21000
140         Jeans          clothing           7000
OPTIONS

```

```

1.View Cart
2.Add to Cart
3.Delete from Cart
4.Log Out
5.Checkout
Select Option : 2
Enter product id : 145
Enter quantity 6

```

[ ]:

```

"""
                                     HOME
-----
ID          NAME          CATEGORY          PRICE
123         Crocs          footwear          2000
131         Clock          electronics        32000
145         Tshirt         clothing           800
124         Nike           footwear          3500
138         Phone          electronics        21000
140         Jeans          clothing           7000
OPTIONS

1.View Cart
2.Add to Cart
3.Delete from Cart
4.Log Out
5.Checkout
Select Option : 3
Enter product id : 138
                                     HOME
-----

```



ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
140	Jeans	clothing	7000

OPTIONS

1.View Cart  
2.Add to Cart  
3.Delete from Cart  
4.Log Out  
5.Checkout  
Select Option : 1

#### CART

SESSION-ID	PRODUCT-ID	QUANTITY
21874	124	3
21874	145	6

""

[ ]:

""

#### HOME

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
140	Jeans	clothing	7000

OPTIONS

1.View Cart  
2.Add to Cart  
3.Delete from Cart  
4.Log Out  
5.Checkout  
Select Option : 5

#### CHECKOUT

Select Payment Method

```
1.Net Banking 2.PayPal
Select the checkout method: 2
PayPal payment successfull
HOME
```

```
-----
ID          NAME          CATEGORY          PRICE
123         Crocs          footwear          2000
131         Clock          electronics       32000
145         Tshirt          clothing          800
124         Nike            footwear          3500
138         Phone           electronics       21000
140         Jeans           clothing          7000
OPTIONS
```

```
1.View Cart
2.Add to Cart
3.Delete from Cart
4.Log Out
5.Checkout
Select Option : 1
```

```
CART
```

```
-----
Cart is Empty
```

```
HOME
```

```
-----
ID          NAME          CATEGORY          PRICE
123         Crocs          footwear          2000
131         Clock          electronics       32000
145         Tshirt          clothing          800
124         Nike            footwear          3500
138         Phone           electronics       21000
140         Jeans           clothing          7000
OPTIONS
```

```
1.View Cart
2.Add to Cart
3.Delete from Cart
4.Log Out
5.Checkout
Select Option : 4
```

```
[ ]: '''
Welcome to the Demo Marketplace
-----
```

```

                                LOGIN
1.Login as User                2.Login as Admin

```

```

Enter your choice:  2
Enter Username:  admin1
Enter password for admin1 :  admin1pass
Login successful as Admin

```

```

OPTIONS

```

```

1.View Catalog
2.Update Catalog
3.Log Out
Select Option :  1

```

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
140	Jeans	clothing	7000

```

""""

```

```

[ ]:  '''
OPTIONS

1.View Catalog
2.Update Catalog
3.Log Out
Select Option :  2
1.Add new product 2. Delete a product 3. Change product price

select one from above 1
Enter product id :  235
Enter product name:  Ring
Enter product category:  Styling
Enter product price:  100
OPTIONS

1.View Catalog
2.Update Catalog
3.Log Out
Select Option :  1

```

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
140	Jeans	clothing	7000
235	Ring	Styling	100

OPTIONS

1.View Catalog

2.Update Catalog

3.Log Out

Select Option : 2

1.Add new product 2. Delete a product 3. Change product price

select one from above 2

Enter product id : 140

OPTIONS

1.View Catalog

2.Update Catalog

3.Log Out

Select Option : 1

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
235	Ring	Styling	100

OPTIONS

1.View Catalog

2.Update Catalog

3.Log Out

Select Option : 2

1.Add new product 2. Delete a product 3. Change product price

select one from above 3

Enter product id : 235

Enter new price: 150

OPTIONS

```
1.View Catalog
2.Update Catalog
3.Log Out
Select Option : 1
```

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
235	Ring	Styling	150

OPTIONS

```
1.View Catalog
2.Update Catalog
3.Log Out
Select Option : 3
```

```
[ ]: '''
OPTIONS

1.View Catalog
2.Update Catalog
3.Log Out
Select Option : 2
1.Add new product 2. Delete a product 3. Change product price
```

```
select one from above 1
Enter product id : 235
Enter product name: Ring
Enter product category: Styling
Enter product price: 100
OPTIONS
```

```
1.View Catalog
2.Update Catalog
3.Log Out
Select Option : 1
```

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000

145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
140	Jeans	clothing	7000
235	Ring	Styling	100

OPTIONS

1.View Catalog

2.Update Catalog

3.Log Out

Select Option : 2

1.Add new product 2. Delete a product 3. Change product price

select one from above 2

Enter product id : 140

OPTIONS

1.View Catalog

2.Update Catalog

3.Log Out

Select Option : 1

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
235	Ring	Styling	100

'''

[ ]:

'''

OPTIONS

1.View Catalog

2.Update Catalog

3.Log Out

Select Option : 2

1.Add new product 2. Delete a product 3. Change product price

select one from above 3

Enter product id : 235

Enter new price: 150

OPTIONS

1.View Catalog  
2.Update Catalog  
3.Log Out  
Select Option : 1

ID	NAME	CATEGORY	PRICE
123	Crocs	footwear	2000
131	Clock	electronics	32000
145	Tshirt	clothing	800
124	Nike	footwear	3500
138	Phone	electronics	21000
235	Ring	Styling	150

OPTIONS

1.View Catalog  
2.Update Catalog  
3.Log Out  
Select Option : 3  
'''