

Assignment - Creating a Shopping App Using Python

Overview

In this project, participants will put the full spectrum of Python programming fundamentals to practice, from data preparation, conditional statements, and loops. The focus is on building the backend functionality of a simple shopping application for an e-commerce website. Participants will use Python to manage login, add or update product categories, and manage items on user carts.

Task

1. Display a Welcome Message

- Show a welcome message in the e-commerce application, such as "Welcome to the Demo Marketplace".

2. Create User and Admin Login

- Implement user and admin login functionality.
- Create demo databases for user and admin verification, and session ID creation.
- After creating the databases, set up admin and user logins.

3. Product Catalog

- Create a sample product catalog with three to four product categories, such as Boots, Coats, Jackets, and Caps.
- The dummy database for the catalog should include:
 - Product ID
 - Name
 - Category ID
 - Price
- Both users and administrators should be able to view the catalog.

4. User Login Rights

- Users should have the ability to:
 - View cart contents
 - Add items to the cart
 - Remove items from the cart
- Users should be able to add or delete items in the cart using:
 - Session ID
 - Product ID
 - Quantity

5. Payment Checkout Options

- Provide demo payment checkout options such as:
 - Net banking
 - PayPal

- After selecting the payment option, display a checkout message like:
 - "Your order is successfully placed"
 - "You will be shortly redirected to the portal to make a payment of \$20"

6. Admin Login

- Only the admin can log in using their credentials.
- If the admin tries to log in using incorrect credentials, an error notice should appear.

7. Admin Functionality

- The admin should not be able to perform the same functions as the user.
- If the admin attempts to carry out user tasks, an error should appear.
- The admin should be able to:
 - Add new products to the catalog with selected attributes
 - Modify existing products using an admin session ID
 - Remove any existing products from the catalog
 - Add new product categories
 - Delete existing product categories

8. Access Control

- Users should not have access to the admin's rights as described above.

ccess to the admin's rights as described above

```
In [ ]: import time

print(f"\n{'-'*5}Welcome to the demo Marketplace{'-'*5}\n")

#Login db
user_db={'username':'user','password':'user@pass123'}
admin_db={'username':'admin','password':'root'}

#globals session storage
current_session_id = None
user = None

#Catalogs
catalog = [
    {'id': '101', 'name': 'Puma', 'category': 'Boots', 'price': 10000},
    {'id': '102', 'name': 'Reymond', 'category': 'Coats', 'price': 15000},
    {'id': '103', 'name': 'H&M', 'category': 'Jackets', 'price': 6000},
    {'id': '104', 'name': 'Urban Monkey', 'category': 'Caps', 'price': 5000},
]

def display_catalog():
    print(f"\n{'Item id':<10}|{'Name':<14}|{'Category':<10}|{'Price':>10}")
    print('{'*50)
    for item in catalog:
        print(f"{item['id']:<10}|{item['name']:<14}|{item['category']:<10}|{item['price']}")

#Cart Functionality
cart=[]
```

```

def display_cart():
    print(f"\n{'Item id':<10}|{'Name':<14}|{'Category':<10}|{'Price':<10}|")
    print('-'*90)
    if len(cart)>0:
        for item in cart:
            print(f"{item['id']:<10}|{item['name']:<14}|{item['category']:<10}|{item['price']:<10}|")
    else:
        print("No items in cart")
    pass

def add_to_cart(session_id, product_id, quantity):
    global current_session_id
    if current_session_id == session_id:

        #if item already in cart
        for item in cart:
            if item['id'] == product_id:
                item['quantity'] += quantity
                return

        #if item not already in cart
        for item in catalog:
            if item['id'] == product_id:
                user_item = item
                user_item['quantity'] = quantity
                cart.append(user_item)
        print(f"\n{'-'*5}Item added to cart successfully{'-'*5}\n")
    else:
        print("Session expired Login Again")

def delete_from_cart(session_id, product_id):
    global current_session_id
    if current_session_id == session_id:
        for item in cart:
            if item['id'] == product_id:
                cart.remove(item)
        print(f"\n{'-'*5}Item removed from cart successfully{'-'*5}\n")
        return

    else:
        print(f"\n{'-'*5} item not found {'-'*5}\n")

else:
    print("Session expired Login Again")

def get_total():
    total = 0
    for item in cart:
        total += (item['price'] * item['quantity'])
    return total

#Payment Gateway service
def checkout():
    if len(cart)<=0:
        print("\nYour cart is empty!\n")
        return
    while True:
        print(f"\n{'*' * 5}CHECKOUT{'*' * 5}\n")
        print("\n1. UPI\n2. Net Banking\n3. COD\n4. Cancel")
        ch = int(input("Select payment method: "))

```

```

if ch == 1:
    print("\nTotal Amount: ",get_total())
    upi_id = input("\nEnter UPI Id: ")
    print("\nUPI payment process initiated\n")
    print("Processing",end=" ")
    for i in range(10):
        time.sleep(0.5)
        print('.', end=" ", flush=True)
    print(f"\n\n{'*'*5}UPI payment is successfully{'*'*5}\n")
    break
elif ch == 2:
    while True:
        #handling if user type string instead of integer
        try:
            card_no = int(input("Enter card number: "))
            cvv = int(input("Enter the CVV number: "))
            fullName = input("Enter Card owners name: ")
            print("\nNetbanking payment process initiated\n")
            print("Processing",end=" ")
            for i in range(10):
                time.sleep(0.5)
                print('.', end=" ", flush=True)
            print(f"\n\n{'*'*5}Payment is successfully{'*'*5}\n")
            break
        except ValueError as e:
            print("Only number are allowed!")
    break
elif ch == 3:
    print("\nOrder Placed successfully!\nPayment will be done by cash on delivery")
    break
elif ch == 4:
    print(f"\n{'*'*5}Exiting payment gateway{'*'*5}\n")
    break
else:
    print("Invalid option")

#ADMIN functionalities
#add product
def add_products_to_catalog():
    id = input("Enter id: ")
    for item in catalog:
        if item['id'] == id:
            print(f"\n{'*'*4}Item with id {id} Already exist{'*'*4}\n")
            return
    else:
        name = input("Enter name: ")
        category = input("Enter category: ")
        while True:
            try:
                price = int(input("Enter price: "))
                catalog.append({'id':id, "name": name, "category":category, "price":price})
                print(f"\n{'*'*5}Item added successfully{'*'*5}\n")
                return True
            except ValueError as e:
                print(f"\n{'*'*5}Invalid price format{'*'*5}\n")
#update product
def update_product():
    product_id = input("Enter product id: ")
    found= False

```

```

for item in catalog:
    if item['id'] == product_id:
        found = True
if found:
    for item in catalog:
        if item['id'] == product_id:
            name=input("Enter Item name: ")
            category = input("Enter item category: ")
            while True:
                try:
                    price = int(input("Enter item price: "))
                    item['name'] = name
                    item['category'] = category
                    item['price'] = price
                    print(f"\n{'-'*5}Product updated successfully{'-'*5}\n")
                    return
                except ValueError as e:
                    print("invalid format")
else:
    print(f"\n{'!'*5} Product not found {'!'*5}\n")

def delete_from_catalog():
    product_id = input("Enter product id: ")
    found= False
    for item in catalog:
        if item['id'] == product_id:
            found = True
    if found:
        for item in catalog:
            if item['id'] == product_id:
                catalog.remove(item)
        print(f"\n{'-'*5}Item removed from catalog successfully{'-'*5}\n")
    else:
        print(f"\n{'!'*5} Product not found {'!'*5}\n")

#user Login
def user_login():
    username = input("Enter username: ")
    password = input("Enter password: ")
    if username == user_db['username'] and password == user_db['password']:
        print(f"\n{'*' * 5}Login successful as user{'*' * 5}\n")
        global current_session_id
        current_session_id = generate_session_id()
        global user
        user = 'user'
        return True
    else:
        print("\n--login failed--\n")

#admin Login
def admin_login():
    username = input("Enter username: ")
    password = input("Enter password: ")
    if username == admin_db['username'] and password == admin_db['password']:
        print(f"\n{'*' * 5}Login successful as admin{'*' * 5}\n")
        global current_session_id
        current_session_id = generate_session_id()
        global user
        user = 'admin'
        return True

```

```

else:
    print("\n-- login failed--\n")

def generate_session_id():
    return "12345"

#Login menu
def login():
    global user
    while True:
        print(f"\n{'-'*5}Login{'-'*5}\n")
        print("1. User Login\n2. Admin Login\n3. Exit application\n")
        ch = input("Enter your choice: ")
        if ch == '1':
            status = user_login()
            if status == True:
                user = 'user'
                menu()
                break
        elif ch == '2':
            status = admin_login()
            if status == True:
                user = 'admin'
                menu()
                break
        elif ch == '3':
            print("\nExiting application...\n")
            exit()
    else:
        print("Invalid choice!")

def menu():
    #User menu
    global user
    global current_session_id
    if user == 'user':
        while True:
            print(f"\n{'**'*5}Menu{'**'*5}\n")
            print("\n1. View Catalog\n2. Add to cart\n3. View Cart\n4. Delete the")
            ch = input("Enter your choice: ")
            if ch == '1':
                display_catalog()
            elif ch == '2':
                product_id = input("Enter product_id: ")
                #if item already in cart
                found = False
                for item in catalog:
                    if item['id'] == product_id:
                        found = True
                if found == False:
                    print(f"\n{'-'*5}Product not found{'-'*5}\n")
                else:
                    quantity = int(input("Enter the quantity: "))
                    if add_to_cart(current_session_id, product_id, quantity):
                        break
            elif ch == '3':
                display_cart()
            elif ch == '4':
                product_id = input("Enter product_id: ")

```

```

        delete_from_cart(current_session_id, product_id)
    elif ch == '5':
        checkout()
    elif ch == '6':
        print("\nLogging out...\n")
        current_session_id = None
        user = None
        # break
        login()
    else:
        print("Invalid Choice")

#Admin menu
if user == 'admin':
    while True:
        print(f"{'-'*5}MENU{'-'*5}")
        print("\n1. Add new product\n2. View Catalog\n3. Update a product\n4.
ch = int(input("Enter your choice: "))
if ch == 1:
    add_products_to_catalog()
elif ch == 2:
    display_catalog()
elif ch == 3:
    update_product()
elif ch == 4:
    delete_from_catalog()
elif ch == 5:
    print("\nLogging out...\n")
    user = None
    # break
    login()
else:
    print("Invalid choice")

if __name__ == "__main__":
    login()

```

-----Welcome to the demo Marketplace-----

-----Login-----

1. User Login
2. Admin Login
3. Exit application

*****Login successful as admin*****

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

Item id . . .	Name	Category . . .	Price
101	Puma	Boots	10000
102	Reymond	Coats	15000
103	H&M	Jackets	6000
104	Urban Monkey	Caps	5000

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

*****Item added successfully*****

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

Item id . . .	Name	Category . . .	Price
101	Puma	Boots	10000
102	Reymond	Coats	15000
103	H&M	Jackets	6000
104	Urban Monkey	Caps	5000
105	Arrow	Shirt	3000

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

-----Product updated successfully-----

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

Item id	Name	Category	Price
101	Puma1	Boots_High	120000
102	Reymond	Coats	15000
103	H&M	Jackets	6000
104	Urban Monkey	Caps	5000
105	Arrow	Shirt	3000

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

-----Item removed from catalog successfully-----

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

!!!! Product not found !!!!

-----MENU-----

1. Add new product
2. View Catalog
3. Update a product
4. Delete a product
5. Log out

Logging out...

-----Login-----

1. User Login
2. Admin Login
3. Exit application

*****Login successful as user*****

*****Menu*****

1. View Catalog
2. Add to cart
3. View Cart
4. Delete the Cart
5. Checkout
6. Log out

Item id . . .	Name	Category . . .	Price
101	Puma1	Boots_High	120000
102	Reymond	Coats . . .	15000
103	H&M	Jackets . . .	6000
104	Urban Monkey	Caps	5000

*****Menu*****

1. View Catalog
2. Add to cart
3. View Cart
4. Delete the Cart
5. Checkout
6. Log out

-----Item added to cart successfully-----

*****Menu*****

1. View Catalog
2. Add to cart
3. View Cart
4. Delete the Cart
5. Checkout
6. Log out

Item id . . .	Name	Category . . .	Price	Quantity . . .	Tot al Price
102	Reymond	Coats	15000	2	300 00

*****Menu*****

1. View Catalog
2. Add to cart
3. View Cart
4. Delete the Cart
5. Checkout
6. Log out

-----Item removed from cart successfully-----

*****Menu*****

1. View Catalog
2. Add to cart
3. View Cart
4. Delete the Cart
5. Checkout
6. Log out

Item id	Name	Category	Price	Quantity	Total Price
<hr/>					
No items in cart					

*****Menu*****

1. View Catalog
 2. Add to cart
 3. View Cart
 4. Delete the Cart
 5. Checkout
 6. Log out

Your cart is empty

*****Menu*****

1. View Catalog
 2. Add to cart
 3. View Cart
 4. Delete the Cart
 5. Checkout
 6. Log out

-----Item added to cart successfully-----

*****Menu*****

1. View Catalog
 2. Add to cart
 3. View Cart
 4. Delete the Cart
 5. Checkout
 6. Log out

*****CHECKOUT*****

*****CHECKOUT*****

1. UPI
 2. Net Banking
 3. COD
 4. _____

4. Cancel

UPI payment process initiated

Processing

*****UPI payment is successfully*****

*****Menu*****

1. View Catalog
 2. Add to cart
 3. View Cart
 4. Delete the Cart
 5. Checkout
 6. Log out
- Logging out...

-----Login-----

1. User Login
2. Admin Login
3. Exit application

Exiting application...

-----Login-----

1. User Login
2. Admin Login
3. Exit application

In []: