

User Manual_29704154

Legend: "+" = positive cases | "-" = negative cases | "x.x.x_(User)_yyyyy" = user action | (User) = loop action

1.0 main menu

1.1 Starting the program (calls main function)

Ensure text file paths exist, create an admin user if it does not already exist and load products data if products file is empty.

- a) Functions used:
 - i. extract_products_from_files – load the file with the products data
 - ii. register_admin – attempts to create an admin user
 - iii. login_control – manages the user login and navigation of the system
 - iv. print_message – used throughout the files to notify user of system actions
 - v. print_error_message – used throughout the files to notify user of specific error
 - vi. __init__(self) and __str__(self) used throughout all model classes

1.2 Select a main menu action

1.2.1 (User): enter the main menu options selection

"1" to login, "2" to register a new customer, "3" to exit. All other inputs will re prompt the user to enter another selection.

- a) Possible resulting actions:
 - + 1.3.1 (User): enter username and password (main menu choice 1)
 - + 1.4.1 (User): enter new user details (main menu choice 2)
 - + 1.5.1 (System): exit (main menu choice 3)
 - 1.2.1 (User): enter the main menu options selection (if input not 1, 2 or 3)
- b) Functions used:
 - vii. io.main_menu – display the login menu options for the user
 - viii. get_user_input – capture user input for selection

1.3 Login (main menu choice 1)

1.3.1 (User): enter username and password

- a) Possible resulting actions:
 - + 2.1.1 (User): select a admin action
 - + 3.1.1 (User): select an customer action
 - 1.2.1 (User): enter the main menu options selection (if the login was unsuccessful)
- b) Functions used:
 - i. get_user_input – display message and capture user input for username and password
 - ii. login – test user name and password to return user's record as an object
 - iii. decrypt_password – decrypt_password
 - iv. clean_line – prepare the object returned from login for reading

1.4 Register (main menu choice 2)

1.4.1 (User): enter new user details

- a) Possible resulting actions:
 - + 1.2.1 (User): enter the main menu options selection (Ask user to proceed to login)
 - 1.2.1 (User): enter the main menu options selection (if customer details input were not valid)
- b) Functions used:
 - i. add_customer – prompts user input for new user details
 - ii. register_customer – creates a new customer
 - iii. set_customer – assigned validated inputs to a customer object
 - generate_unique_user_id – creates a unique user id
 - validate_username – validates the user name
 - check_username_exist – validates if user name already exists
 - validate_password – validates the user password
 - encrypt_password – encrypts the user password
 - validate_email – validates customer email
 - validate_mobile – validates customer mobile

1.5 Quit (main menu choice 3)

1.5.1 (System): exit

- a) Possible resulting actions:
 - + No further action required – exit the code
- b) Functions used:
 - i. N/A

2.0 admin_menu

2.1 Select an admin menu action.

2.1.1 (User): select a customer action

- a) Possible resulting actions:
 - + 2.2.1 (User): select search type and input according to value if applicable (admin menu choice 1)
 - + 2.3.1 (User): enter new customer details
 - + 2.4.1 (User): enter customer page number to view
 - + 2.5.1 (User): enter order page number to view
 - + 2.6.1 (System): generate test data
 - + 2.8.1 (User): select to delete either customer, product or order
 - + 2.9.1 (System): deletes all data in user(except for admin), products and order files
 - + 2.10.1 (System): logout
 - 2.2.1 (User): select a customer action (if input was not numbers 1 to 8)
- b) Functions used:
 - i. admin_control – manages the admins actions and navigation
 - ii. admin_menu – displays the possible admin actions
 - iii. get_user_input – display message and capture admin action choice input

2.2 Show products (admin menu choice 1)

2.2.1 (User): select search type and input according value if applicable

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if search unsuccessful)
 - 2.2.1 (User): select search type and input according value if applicable (if action choice not as expected)
- b) Functions used:
 - i. show_products – responsible for the handling of product search methods
 - ii. get_user_input – obtain user search choice then page number, keyword or id
 - iii. get_product_list – get the one page of products from the products database
 - iv. get_product_list_by_keyword – get all the products whose name contains the keyword
 - v. get_product_by_id – get one product object based on the specified
 - key_vals – extract key values and creates a dictionary
 - set_product – create and set new values for a product
 - vi. show_list – print to the screen the tuple values from product list and product list by keyword
 - vii. print_object – print to the screen the product by id object

2.3 Add customer (admin menu choice 2)

2.3.1 (User): enter new customer details

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if customer details input were not valid)
- b) Functions used:
 - i. Refer to 1.4.1 -> b) Functions used

2.4 Show customers (admin menu choice 3)

2.4.1 (User): enter customer page number to view

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if page number not valid)
- b) Functions used:
 - i. get_customer_list – get one page of customer from the user file
 - set_customer – create and set new values for a customer
 - clean_line – prepare the customer object for reading
 - page_creation – creates list of objects that are to be displayed on a page
 - ii. show_list – print to the screen the tuple values from get_customer_list

2.5 Show orders (admin menu choice 4)

2.5.1 (User): enter order page number to view

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if no orders exist or page number not valid)
- b) Functions used:
 - i. get_order_list – get one specified page of orders from the order file
 - create_data_frame -> key_vals, set_product
 - page_creation – creates list of objects that are to be displayed on a page
 - ii. show_list – print to the screen the tuple values from get_order_list

2.6 Generate test data (admin menu choice 5)

2.6.1 (System): generate test data

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if no products exist)
- b) Functions used:
 - i. generate_test_order_data – generate 10 users and 50 to 200 orders for each
 - register_customer – refer to 1.4.1 -> b) Functions used except for point i
 - create_data_frame -> key_vals, set_product
 - create_an_order -> generate_unique_order_id

2.7 Generate all statistical figures (admin menu choice 6)

2.7.1 (System): generate all statistical figures

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if no orders/products exist or page number not valid)
- b) Functions used:
 - i. generate_category_figure
 - key_vals – extract key values and creates a dictionary
 - ii. generate_discount_figures
 - key_vals – extract key values and creates a dictionary
 - iii. generate_likes_count_figure
 - create_data_frame -> key_vals, set_product
 - iv. generate_discount_likes_count_figure
 - create_data_frame -> key_vals, set_product
 - v. generate_all_top_10_best_sellers_figure
 - create_data_frame -> key_vals, set_product

2.8 Delete specific data (admin menu choice 7)

2.8.1 (User): select to delete either customer, product or order

- a) Possible resulting actions:
 - + 2.1.1 (User): select a customer action (ready for next action)
 - 2.1.1 (User): select a customer action (if invalid option, id reference does not exist or no data to search)
- b) Functions used:
 - i. get_user_input – used to obtain deletion type, then separately obtain corresponding identifier
 - ii. delete_customer – deletes a customer by id
 - iii. delete_product – deletes a product by id
 - key_vals → set_product
 - iv. delete_order – deletes an order by id
 - create_data_frame → key_vals, set_product

2.9 Delete all data (admin menu choice 8)

2.9.1 (System): deletes all data in user(except for admin), products and order files

- a) Possible resulting actions:
 - + (User): select a customer action (ready for next action)
- b) Functions used:
 - i. delete_all_customers – removes all customers from the user file
 - clean_line – prepare the customer object for reading
 - ii. delete_all_products
 - iii. delete_all_orders

2.10 Logout

2.10.1 (System): logout

- a) Possible resulting actions:
 - + 1.2.1 (User): enter the main menu options selection
- b) Functions used:
 - i. N/A

3.0 customer_menu

3.1 Select a customer menu action

3.1.1 (User): select an customer action

- a) Possible resulting actions:
 - + 3.2.1 (System): show the customers profile
 - + 3.3.1 (User): input the attribute and value
 - + 3.4.1 (User): select search type and input according value if applicable
 - + 3.5.1 (User): select page of history orders to view
 - + 3.6.1 (System): generate all consumption figures
 - + 3.7.1 (System): logout
 - 3.1.1 (User): select an customer action (if input was no number 1 to 6)
- b) Functions used:
 - i. customer_control – manages the customer actions and navigation
 - ii. clean_line – prepare the object returned from login for reading
 - iii. customer_menu – show the customer menu options
 - iv. get_user_input – obtain the customer menu selection

3.2 Show profile (customer menu choice 1)

3.2.1 (System): show the customers profile

- a) Possible resulting actions:
 - + 3.1.1 (User): select an customer action (ready for next action)
- b) Functions used:
 - i. print_object – prints the string representation of the customer object

3.3 Update profile (customer menu choice 2)

3.3.1 (User): input the attribute and value

- a) Possible resulting actions:
 - + 3.1.1 (User): select an customer action (ready for next action)
 - 3.1.1 (User): select an customer action (if input for attribute or value invalid)
- b) Functions used:
 - i. get_user_input – obtain user input for attribute and value
 - ii. update_profile – updates the customers profile with the attribute and value
 - clean_line – prepares the object for reading
 - validate_username – validates the user name
 - check_username_exist – validates if user name already exists
 - validate_password – validates the user password
 - encrypt_password – encrypts the user password
 - validate_email – validates customer email
 - validate_mobile – validates customer mobile

3.4 Show products (customer menu choice 3)

3.4.1 (User): select search type and input according to value if applicable

- a) Possible resulting actions:
 - + 3.1.1 (User): select an customer action (ready for next action)
 - 3.1.1 (User): select an customer action (if page number not valid)
- b) Functions used:
 - i. Refer to 2.4.1 -> b) Functions used

3.5 Show history orders (customer menu choice 4)

3.5.1 (User): select page of history orders to view

- a) Possible resulting actions:
 - + 3.1.1 (User): select an customer action (ready for next action)
 - 3.1.1 (User): select an customer action (if customer has no order history or page number not valid)
- b) Functions used:
 - ii. get_user_input – obtain user input for page number
 - iii. get_order_list – shows the page of orders specified
 - create_data_frame -> key_vals, set_product
 - page_creation – creates list of objects that are to be displayed on a page
 - iv. show_list – print to the screen the tuple values from product list and product list by keyword

3.6 Generate all consumption figures (customer menu choice 5)

3.6.1 (System): generate all consumption figures

- a) Possible resulting actions:
 - + 3.1.1 (User): select an customer action (ready for next action)
 - 3.1.1 (User): select an customer action (if customer has no order history or no order data exists)
- b) Functions used:
 - i. generate_single_customer_consumption_figure
 - create_data_frame -> key_vals, set_product
 - get_price_month – gets the current price and month of a product
 - sum_price_month – calculates the total current price by month
 - ii. generate_all_customers_consumption_figure
 - create_data_frame -> key_vals, set_product

3.7 Logout (customer menu choice 6)

3.7.1 (System): logout

- a) Possible resulting actions:
 - + 1.2.1 (User): enter the main menu options selection
- b) Functions used:
 - i. N/A