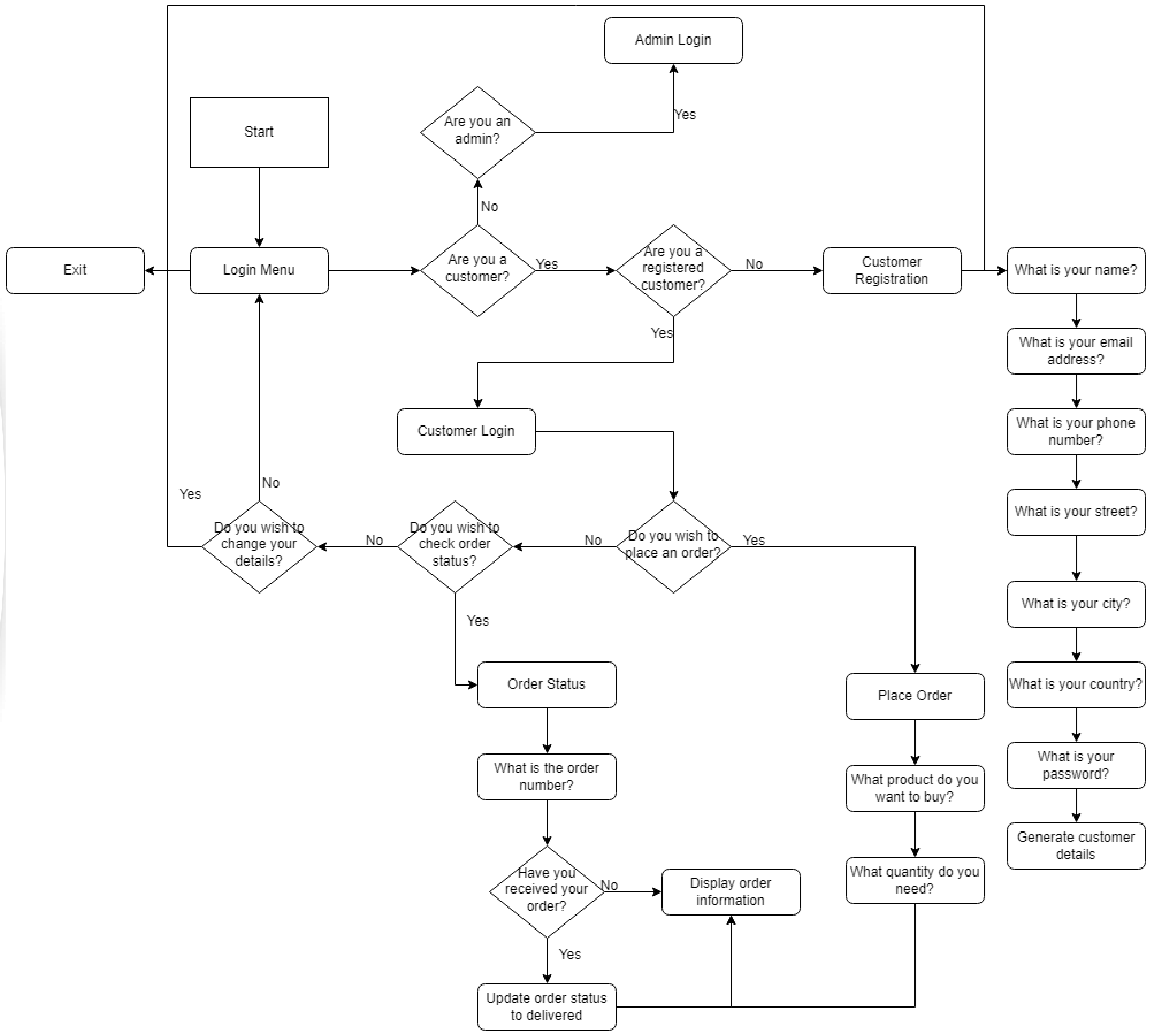
# It’s not a Bug it’s a Feature

by The Bugs Busters

# Introduction

- Creating a system that records admin/customer information as well as all orders placed, products details, and their relevant categories.
- Upon starting the program, it prompts you to login either as an existent customer or as an admin. If the customer is not registered, it will take them straight to registration and then back to the login menu.
- From the customer's perspective, they can place an order, check their order(s) status, change their personal details or logout and go back to the login menu.
- From the admin's perspective, they can insert a new category, new product, check sales or logout and go back to the login menu.

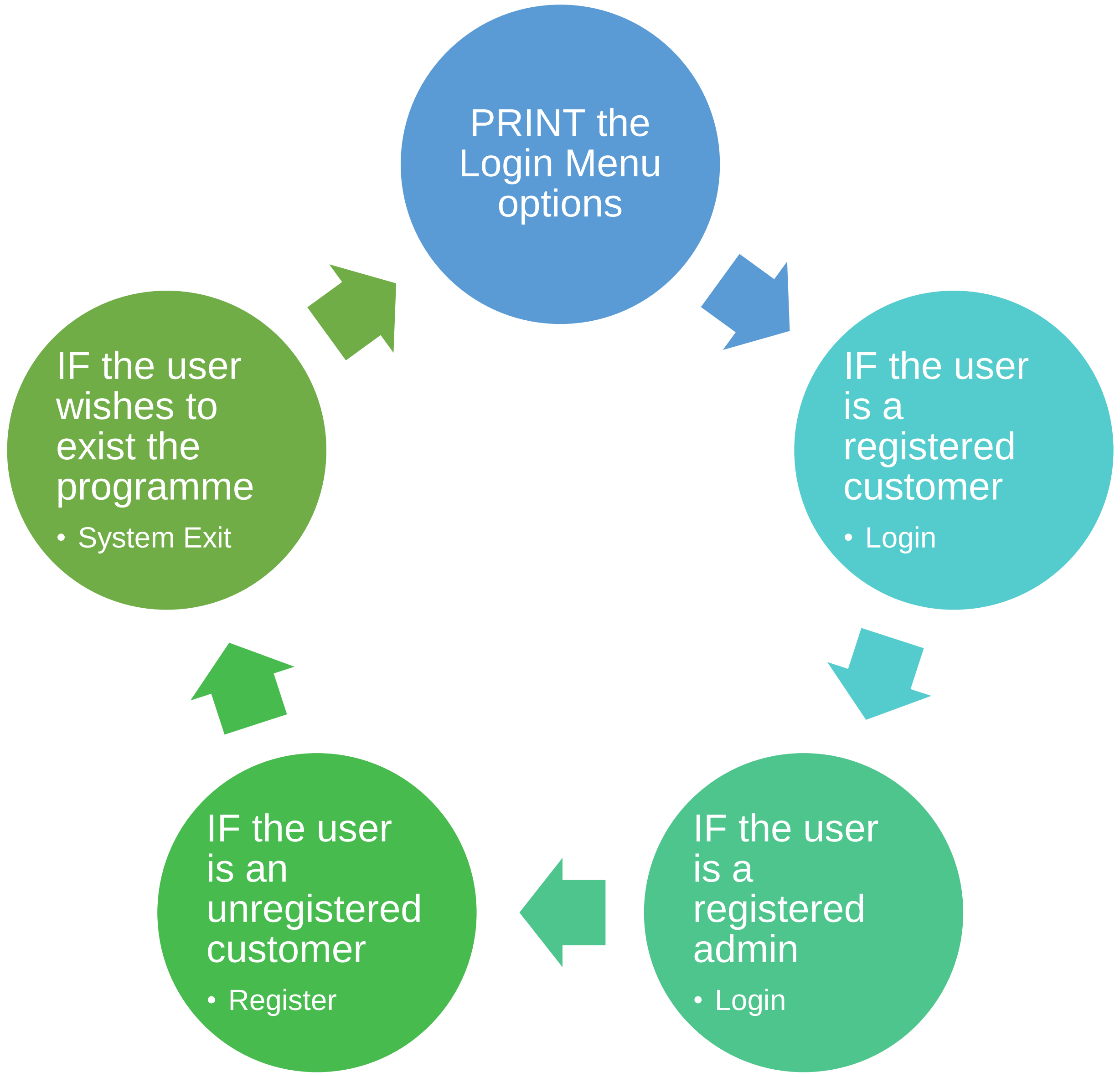
# Flowchart


Start
Login Menu
Exit
Are you a customer?
Yes
No
Are you an admin?
Yes
Admin Login
Are you a registered customer?
No
Customer Registration
What is your name?
What is your email address?
What is your phone number?
What is your street?
What is your city?
What is your country?
What is your password?
Generate customer details
Yes
Customer Login
Do you wish to place an order?
Yes
Place Order
What product do you want to buy?
What quantity do you need?
No
Do you wish to check order status?
Yes
Order Status
What is the order number?
Have you received your order?
No
Display order information
Yes
Update order status to delivered
No
Do you wish to change your details?
Yes
No

## Team Description

| | Abas | Beatrice | Haze |
|---|---|---|---|
| **Initial Tasks** | Product Function<br>Category Function | Order Function<br>Admin | Customer Details |
| **Worked on after:** | Product Function<br>Category Function<br>Login Menu<br>Dashboard Menu<br>Flowchart | Order Function<br>Login Menu<br>Dashboard Menu<br>Flowchart | Customer Details<br>Sales Functions<br>Bug Fixes<br>Pseudocode |

# Pseudocode Login


PRINT the Login Menu options
IF the user is a registered customer
• Login
IF the user is a registered admin
• Login
IF the user is an unregistered customer
• Register
IF the user wishes to exist the programme
• System Exit

# Pseudocode Admin

PRINT the Dashboard options

IF the user wishes to insert a new category

- Call Insert Category

IF the user wishes to insert a new product

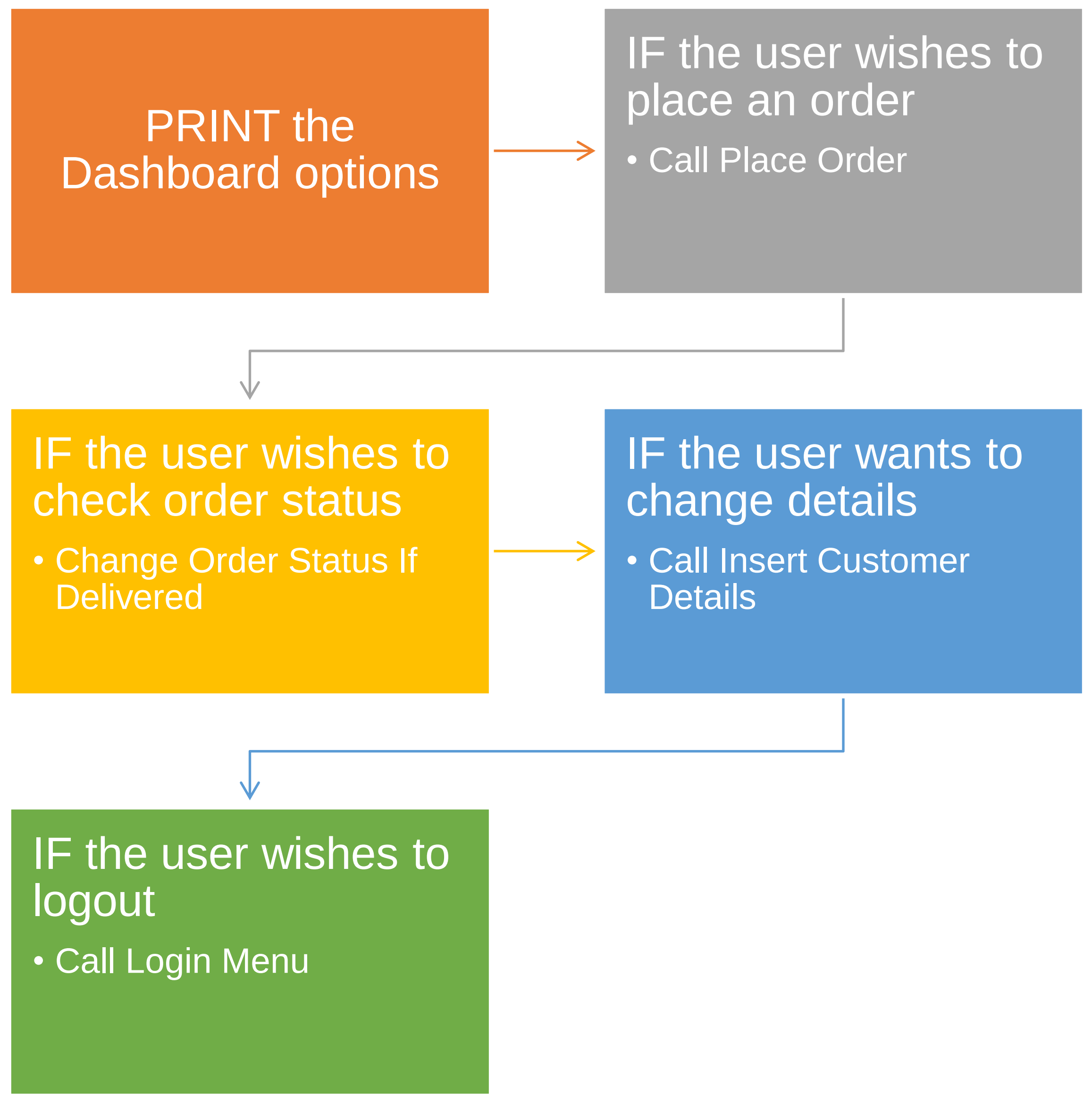
- Call Insert Product

IF the user wants to view total sales

- Get More Filter Options
  - Call The Function That Returns Filter Selection

IF the user wishes to logout

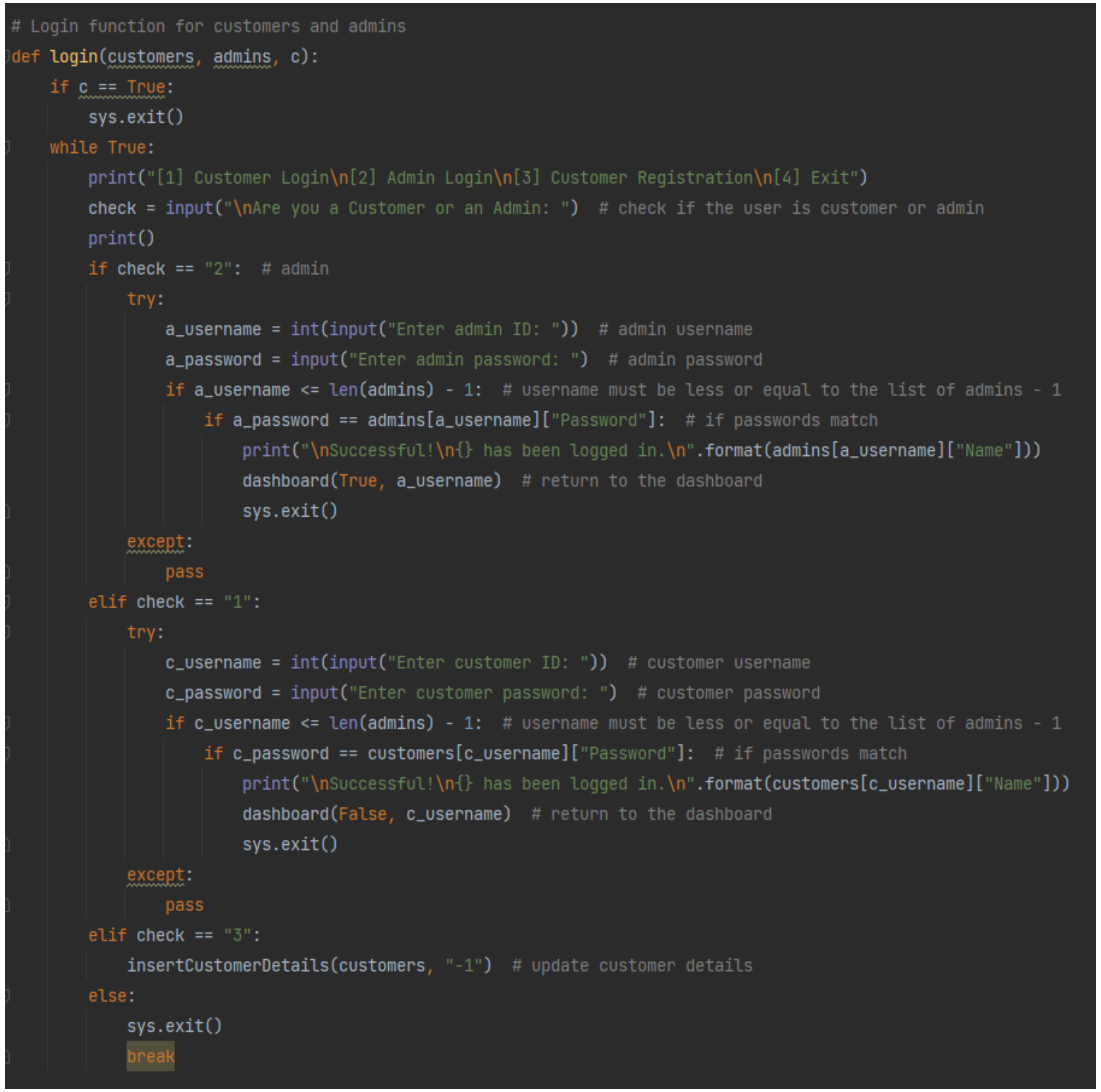
- Call Login Menu


PRINT the Dashboard options
IF the user wishes to place an order
• Call Place Order
IF the user wishes to check order status
• Change Order Status If Delivered
IF the user wants to change details
• Call Insert Customer Details
IF the user wishes to logout
• Call Login Menu

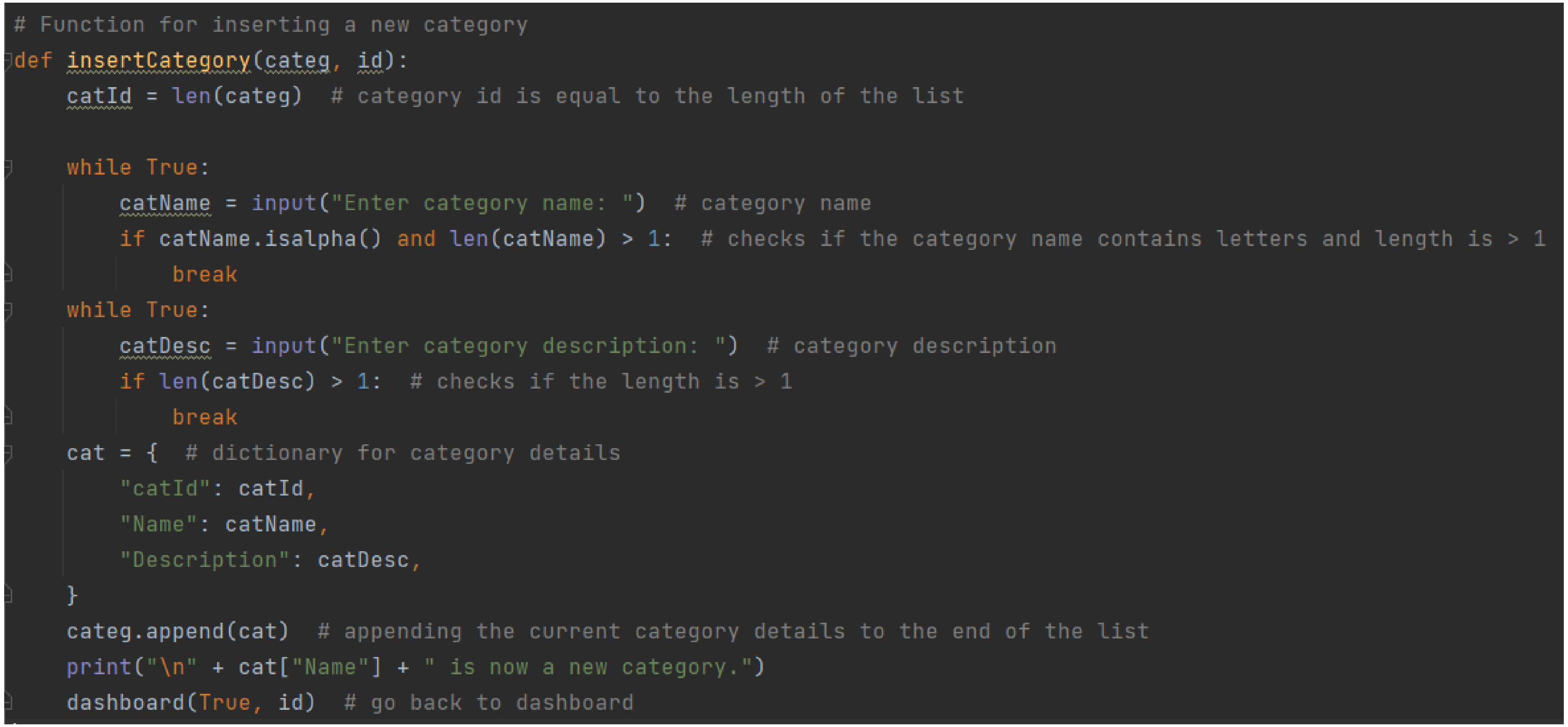
## Pseudocode Customer

Login Menu Function

## Code Snippets



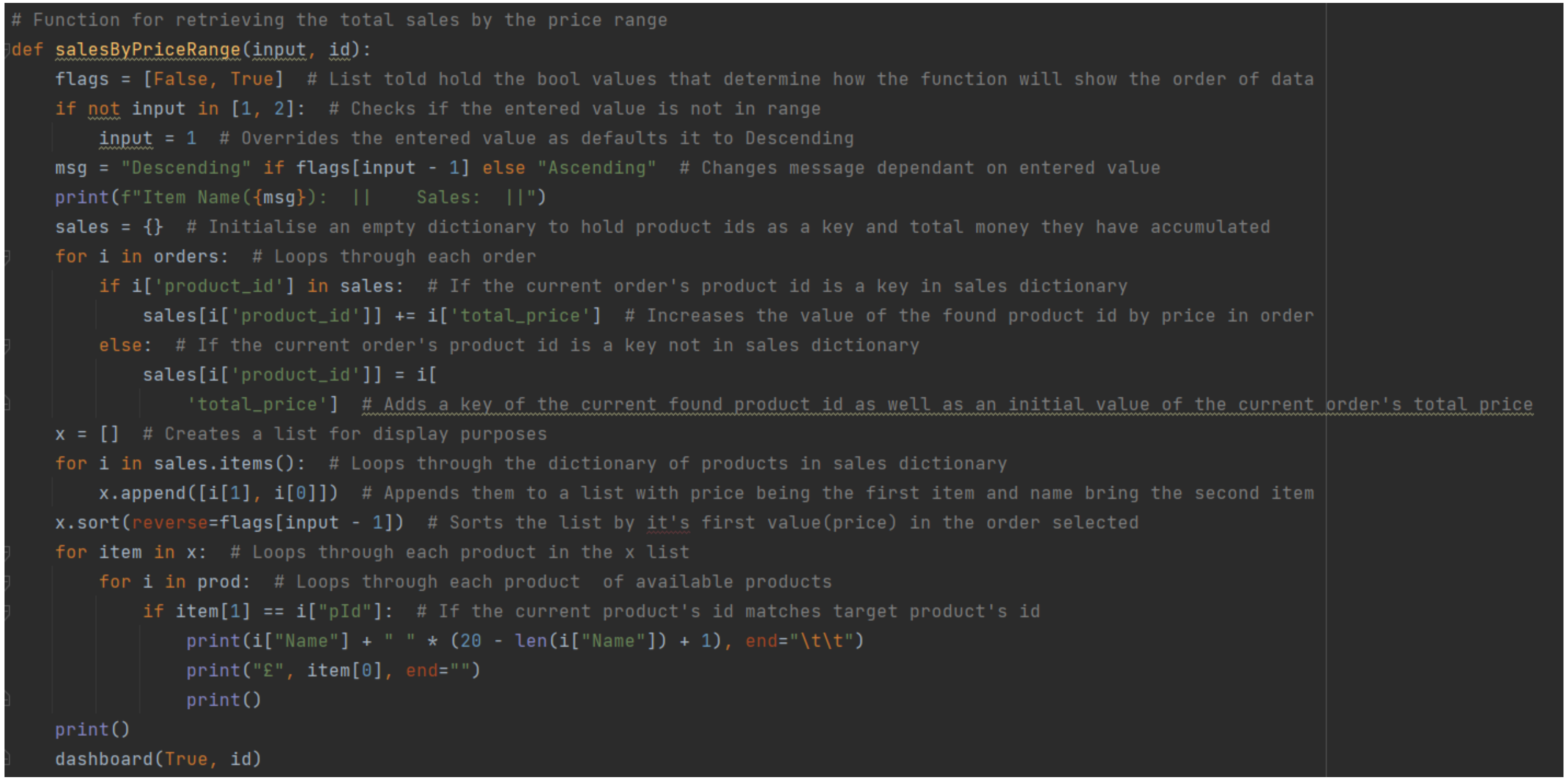
```python
# Login function for customers and admins
def login(customers, admins, c):
    if c == True:
        sys.exit()
    while True:
        print("[1] Customer Login\n[2] Admin Login\n[3] Customer Registration\n[4] Exit")
        check = input("\nAre you a Customer or an Admin: ")  # check if the user is customer or admin
        print()
        if check == "2":  # admin
            try:
                a_username = int(input("Enter admin ID: "))  # admin username
                a_password = input("Enter admin password: ")  # admin password
                if a_username <= len(admins) - 1:  # username must be less or equal to the list of admins - 1
                    if a_password == admins[a_username]["Password"]:  # if passwords match
                        print("\nSuccessful!\n{} has been logged in.\n".format(admins[a_username]["Name"]))
                        dashboard(True, a_username)  # return to the dashboard
                        sys.exit()
            except:
                pass
        elif check == "1":
            try:
                c_username = int(input("Enter customer ID: "))  # customer username
                c_password = input("Enter customer password: ")  # customer password
                if c_username <= len(admins) - 1:  # username must be less or equal to the list of admins - 1
                    if c_password == customers[c_username]["Password"]:  # if passwords match
                        print("\nSuccessful!\n{} has been logged in.\n".format(customers[c_username]["Name"]))
                        dashboard(False, c_username)  # return to the dashboard
                        sys.exit()
            except:
                pass
        elif check == "3":
            insertCustomerDetails(customers, "-1")  # update customer details
        else:
            sys.exit()
            break
```



```python
# Function for inserting a new category
def insertCategory(categ, id):
    catId = len(categ)  # category id is equal to the length of the list

    while True:
        catName = input("Enter category name: ")  # category name
        if catName.isalpha() and len(catName) > 1:  # checks if the category name contains letters and length is > 1
            break
    while True:
        catDesc = input("Enter category description: ")  # category description
        if len(catDesc) > 1:  # checks if the length is > 1
            break
    cat = {  # dictionary for category details
        "catId": catId,
        "Name": catName,
        "Description": catDesc,
    }
    categ.append(cat)  # appending the current category details to the end of the list
    print("\n" + cat["Name"] + " is now a new category.")
    dashboard(True, id)  # go back to dashboard
```

Insert Category Function

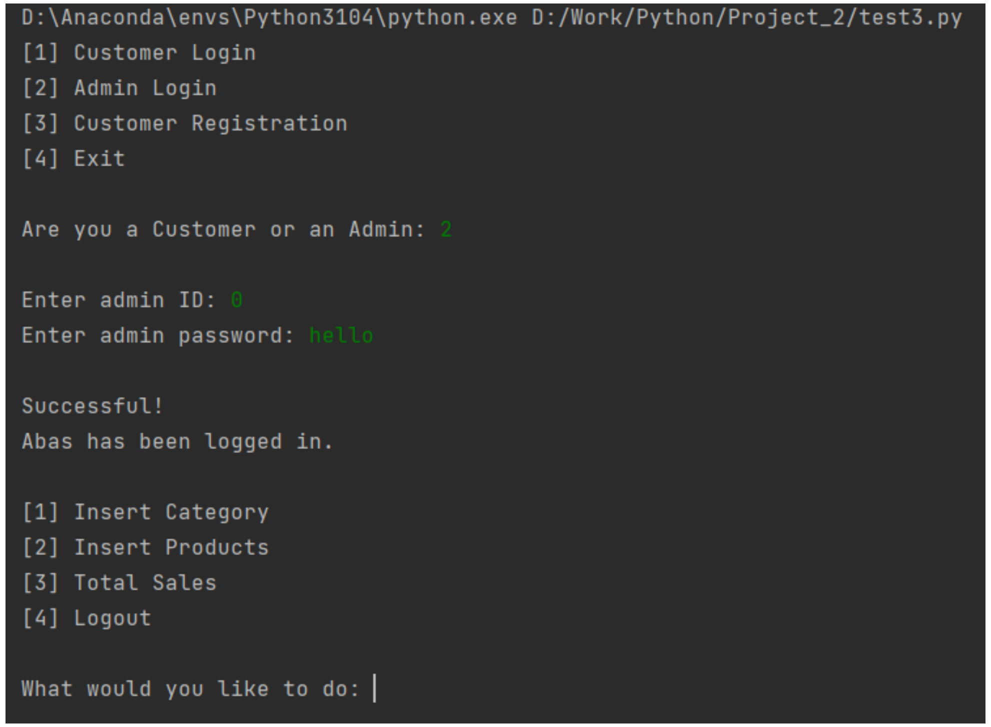
## Code Snippets



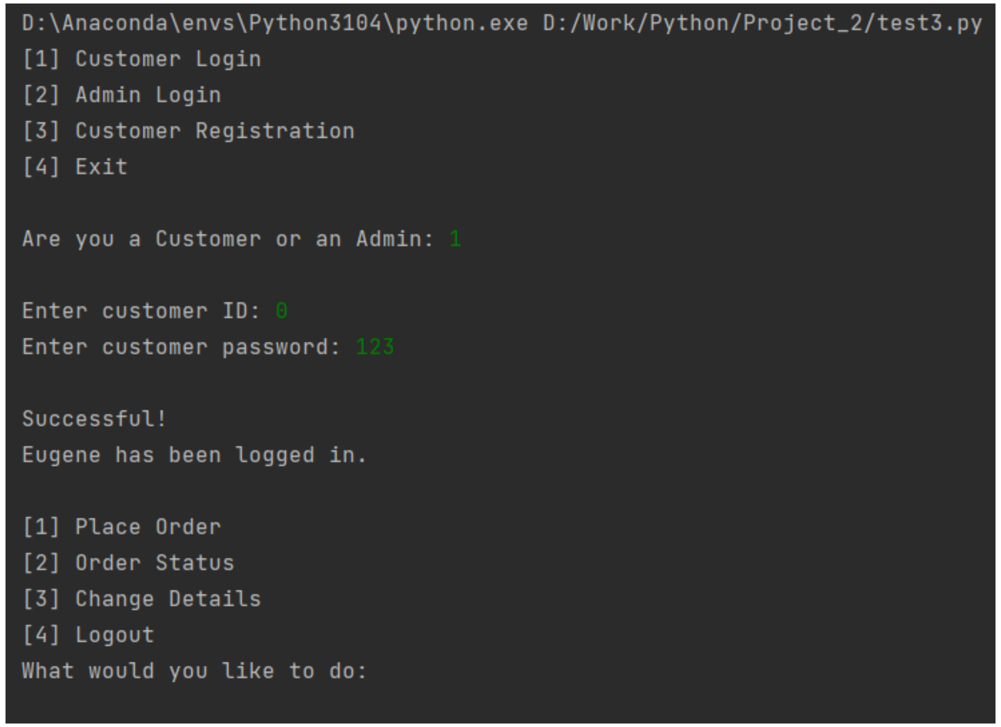
```python
# Function for retrieving the total sales by the price range
def salesByPriceRange(input, id):
    flags = [False, True]  # List told hold the bool values that determine how the function will show the order of data
    if not input in [1, 2]:  # Checks if the entered value is not in range
        input = 1  # Overrides the entered value as defaults it to Descending
    msg = "Descending" if flags[input - 1] else "Ascending"  # Changes message dependant on entered value
    print(f"Item Name({msg}):  ||    Sales:  ||")
    sales = {}  # Initialise an empty dictionary to hold product ids as a key and total money they have accumulated
    for i in orders:  # Loops through each order
        if i['product_id'] in sales:  # If the current order's product id is a key in sales dictionary
            sales[i['product_id']] += i['total_price']  # Increases the value of the found product id by price in order
        else:  # If the current order's product id is a key not in sales dictionary
            sales[i['product_id']] = i[
                'total_price']  # Adds a key of the current found product id as well as an initial value of the current order's total price
    x = []  # Creates a list for display purposes
    for i in sales.items():  # Loops through the dictionary of products in sales dictionary
        x.append([i[1], i[0]])  # Appends them to a list with price being the first item and name bring the second item
    x.sort(reverse=flags[input - 1])  # Sorts the list by it's first value(price) in the order selected
    for item in x:  # Loops through each product in the x list
        for i in prod:  # Loops through each product  of available products
            if item[1] == i["pId"]:  # If the current product's id matches target product's id
                print(i["Name"] + " " * (20 - len(i["Name"]) + 1), end="\t\t")
                print("£", item[0], end="")
                print()
    print()
    dashboard(True, id)
```

Retrieve Total Sales by Price Range Function

## Code Snippets



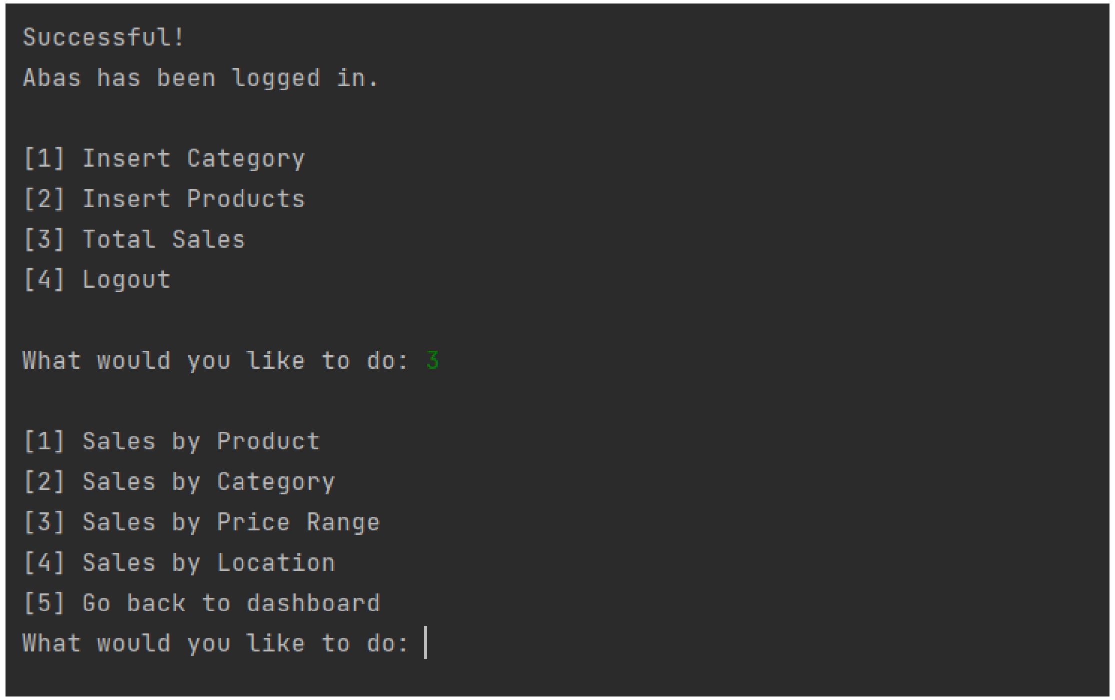
```
D:\Anaconda\envs\Python3104\python.exe D:/Work/Python/Project_2/test3.py
[1] Customer Login
[2] Admin Login
[3] Customer Registration
[4] Exit

Are you a Customer or an Admin: 2

Enter admin ID: 0
Enter admin password: hello

Successful!
Abas has been logged in.

[1] Insert Category
[2] Insert Products
[3] Total Sales
[4] Logout

What would you like to do: 
```



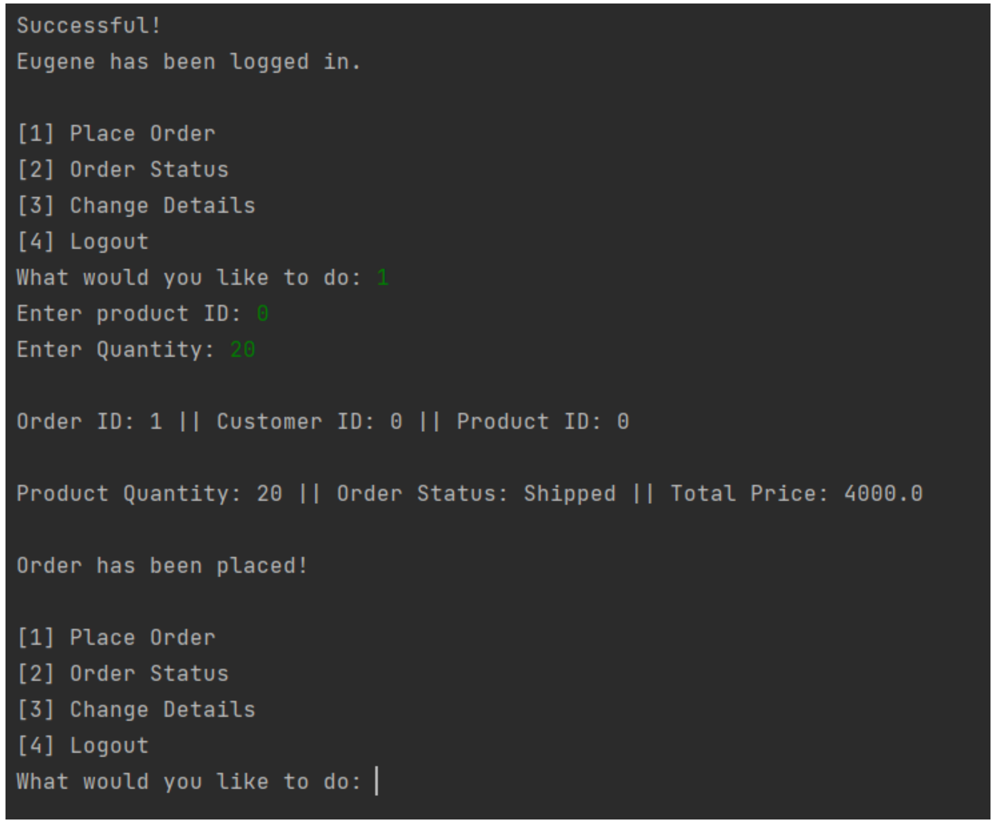
```
D:\Anaconda\envs\Python3104\python.exe D:/Work/Python/Project_2/test3.py
[1] Customer Login
[2] Admin Login
[3] Customer Registration
[4] Exit

Are you a Customer or an Admin: 1

Enter customer ID: 0
Enter customer password: 123

Successful!
Eugene has been logged in.

[1] Place Order
[2] Order Status
[3] Change Details
[4] Logout
What would you like to do: 
```

## Output

## Admin/Customer Login Feature



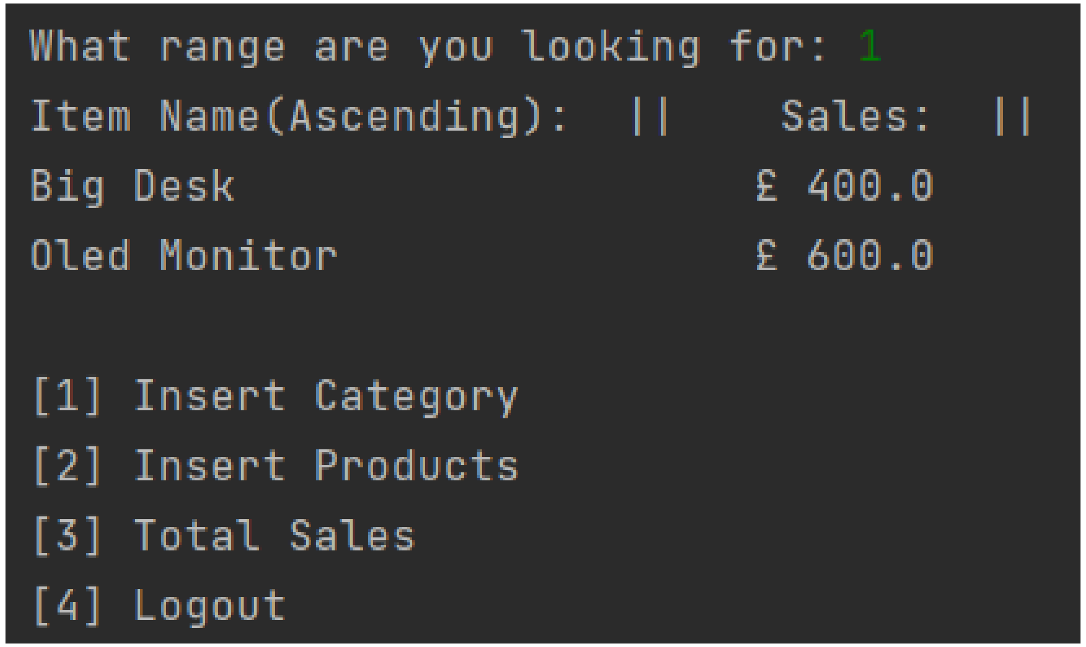
```
Successful!
Abas has been logged in.

[1] Insert Category
[2] Insert Products
[3] Total Sales
[4] Logout

What would you like to do: 3

[1] Sales by Product
[2] Sales by Category
[3] Sales by Price Range
[4] Sales by Location
[5] Go back to dashboard
What would you like to do: 
```



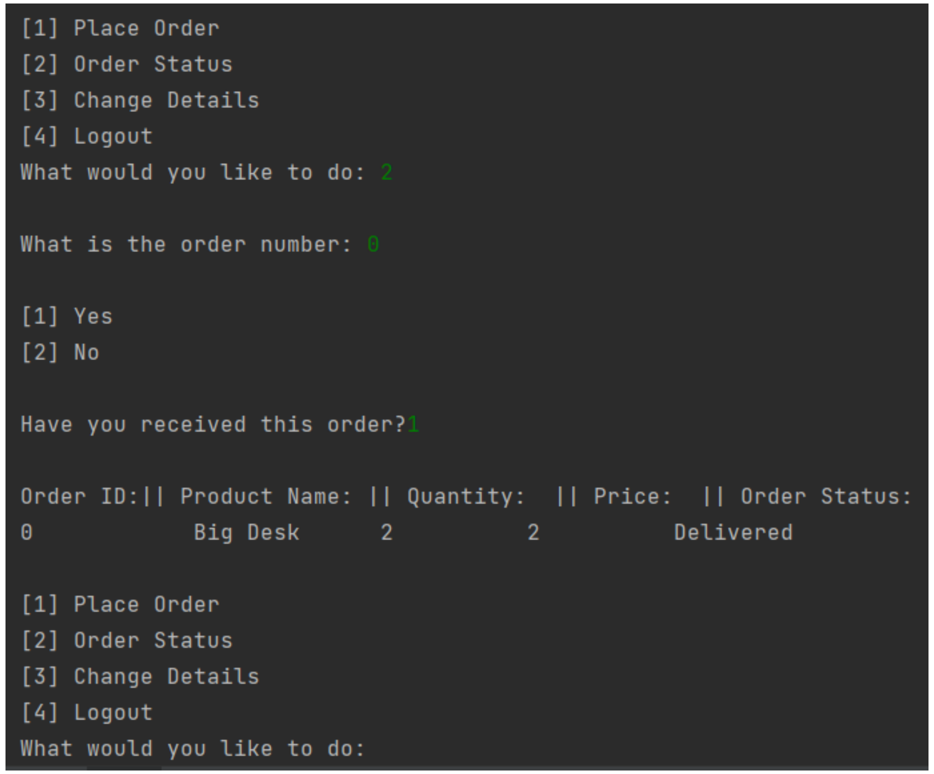
```
Successful!
Eugene has been logged in.

[1] Place Order
[2] Order Status
[3] Change Details
[4] Logout
What would you like to do: 1
Enter product ID: 0
Enter Quantity: 20

Order ID: 1 || Customer ID: 0 || Product ID: 0

Product Quantity: 20 || Order Status: Shipped || Total Price: 4000.0

Order has been placed!

[1] Place Order
[2] Order Status
[3] Change Details
[4] Logout
What would you like to do: 
```

## Output

## Admin/Customer Dashboard Feature



```
What range are you looking for: 1
Item Name(Ascending):  ||    Sales:  ||
Big Desk                    £ 400.0
Oled Monitor                £ 600.0


[1] Insert Category
[2] Insert Products
[3] Total Sales
[4] Logout
```



```
[1] Place Order
[2] Order Status
[3] Change Details
[4] Logout
What would you like to do: 2

What is the order number: 0

[1] Yes
[2] No

Have you received this order?1

Order ID:|| Product Name: || Quantity:  || Price:  || Order Status:
0            Big Desk      2           2          Delivered

[1] Place Order
[2] Order Status
[3] Change Details
[4] Logout
What would you like to do:
```

## Output

## Admin/Customer Dashboard Options

# Conclusion

We split the coding tasks evenly and worked on them individually for the first hour or so before merging our codes.

Then we worked on bug fixes as well as implementing new features to the existing code base.

Finally, we started the documentation process and collaborated on finishing all the deliverables on time.