

Test Suite of Online Sales Portal

Authors : Group 8

(Anand Manojkumar Parikh, Soni Aditya Bharatbhai, Shashwat Naidu)

| | |
|------------------------------------|--------------------|
| Classes ----- | 1 |
| 1) Class Address ----- | 1 |
| 2) Class Category ----- | 2 |
| 3) Class User ----- | 2 |
| 4) Class Seller ----- | 4 |
| 5) Class Manager ----- | 6 |
| 6) Class Item ----- | 10 |
| 7) Class Buyer ----- | 11 |
| 8) Class Order ----- | 12 |
| Interfaces ----- | 12 |
| 9) Sign up Interface ----- | 12 |
| 10) Sign in Interface ----- | 13 |
| 11) Negotiation Interface ----- | 14 |
| 12) Change details interface ----- | 15 |

CLASSES:

1) Class Address

1. Testing constructor

- Empty / Invalid fields (one or more)

| Residence NO. | Street | Locality | City | State | Pincode |
|---------------|--------|----------|-----------|-------|---------|
| "" | "" | "" | Kharagpur | "" | -9999 |

OUTPUT : Exception in Residence NO, Street, Locality, Pincode -

- Mandatory fields like Residence NO. ,state ,street, and locality are empty.
- Pin Code has insufficient digits and is negative.
- Valid fields

| Residence NO. | Street | Locality | City | State | Pincode |
|---------------|--------|----------|------|-------|---------|
|---------------|--------|----------|------|-------|---------|

| | | | | | |
|----|------------------|---------------|-----------|-------------|--------|
| 5A | Tech Market Road | IIT Kharagpur | Kharagpur | West Bengal | 721302 |
|----|------------------|---------------|-----------|-------------|--------|

OUTPUT : OK, all entries are correct. New object constructed and added to database.

2) Class Category:

1. Testing constructor :

- Construct with invalid parameter:

| |
|------|
| Name |
| "" |

Output: Constructor throws an exception for empty category name.

- Construct with valid parameter:

| |
|--------------|
| Name |
| "category_1" |

OUTPUT : OK, all entries are correct. New object constructed and added to database i.e list of categories.

| |
|--------------|
| Name |
| "category_2" |

OUTPUT: OK, all entries are correct. New object constructed and added to database i.e list of categories.

Print categoryList: "category_1" , "category_2" .

Check: category_1.ID == category_2.ID, returns False.

3) Class User :

1. Testing constructor

- Empty / Invalid fields (one or more)

| Password | Name | Email | Address | Telephone |
|----------|------|-------|------------------|-----------|
| "" | "" | Hello | <Address_object> | 100 |

- **OUTPUT** : Exception raised for password (empty), name (empty), email (invalid format), telephone(invalid, not 10 digits)

| Password | Name | Email | Address | Telephone |
|---------------------|-------|------------------------|--|------------|
| #myOSP_password@123 | Anand | anandparikh4@gmail.com | C-212, LBS Hall, IIT Kharagpur, Kharagpur, West Bengal | 9876720201 |

- **OUTPUT** : OK, all entries are correct. New object constructed and added to database.
- **NOTE** : Invalid format of address is handled by class Address.

2. Change Data

| Password | Name | Email | Address | Telephone |
|---------------------|-------|------------------------|--|------------|
| #myOSP_password@123 | Anand | anandparikh4@gmail.com | C-212, LBS Hall, IIT Kharagpur, Kharagpur, West Bengal | 9876720201 |

Give new data : Address - B-112, RK Hall, IIT Kharagpur, Kharagpur, West Bengal, Telephone = 9988711209

OUTPUT :

| Password | Name | Email | Address | Telephone |
|---------------------|-------|------------------------|---|------------|
| #myOSP_password@123 | Anand | anandparikh4@gmail.com | B-112, RK Hall, IIT Kharagpur, Kharagpur, West Bengal | 9988711209 |

3. Change Password :

| Password | Name | Email | Address | Telephone |
|---------------------|-------|------------------------|---|------------|
| #myOSP_password@123 | Anand | anandparikh4@gmail.com | B-112, RK Hall, IIT Kharagpur, Kharagpur, West Bengal | 9988711209 |

Input : Enter old password : @password_OSP

Output : Exception raised - "Incorrect password"

Input : Enter old password : #myOSP_password@123

Enter new password : new_password_1234

OUTPUT : Password successfully changed

| Password | Name | Email | Address | Telephone |
|-------------------|-------|------------------------|---|------------|
| new_password_1234 | Anand | anandparikh4@gmail.com | B-112, RK Hall, IIT Kharagpur, Kharagpur, West Bengal | 9988711209 |

4) Class Seller :

1. **Testing constructor** : Invokes parent's constructor. All exceptions (if any) handled by the super class constructor.

2. **Add_product** : All exceptions (if any) handled by the Item class constructor.

Item_list of the seller :

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|--------------|-------------|-------------|-------|---------|-------------|-----------|----------|
| HP laptop | Electronics | laptop.jpg | 50000 | 2 years | 14' screen | Hyderabad | True |
| Levis' Jeans | Clothes | clothes.jpg | 1500 | - | 32" waist | Surat | False |

Call add_product, which constructs a new product as per entered details.

New Item to be added : Paracetamol, Medicine, medicine.jpg, - , "15 tablets" , Jaipur, False

Output :

Item_list :

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|--------------|-------------|--------------|-------|---------|-------------|-----------|----------|
| HP laptop | Electronics | laptop.jpg | 50000 | 2 years | 14' screen | Hyderabad | True |
| Levis' Jeans | Clothes | clothes.jpg | 1500 | - | 32" waist | Surat | False |
| Paracetamol | Medicine | medicine.jpg | 200 | - | 15 tablets | Jaipur | False |

- 3. Delete_product :** Only items on sale by this seller can be deleted, so no exceptions raised here, under any circumstances.

Item_list :

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|--------------|-------------|--------------|-------|---------|-------------|-----------|----------|
| HP laptop | Electronics | laptop.jpg | 50000 | 2 years | 14' screen | Hyderabad | True |
| Levi's Jeans | Clothes | clothes.jpg | 1500 | - | 32" waist | Surat | False |
| Paracetamol | Medicine | medicine.jpg | 200 | - | 15 tablets | Jaipur | False |

Choose any item to delete, say Levi's Jeans

Output :

Item_list :

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|-------------|-------------|--------------|-------|---------|-------------|-----------|----------|
| HP laptop | Electronics | laptop.jpg | 50000 | 2 years | 14' screen | Hyderabad | True |
| Paracetamol | Medicine | medicine.jpg | 200 | - | 15 tablets | Jaipur | False |

- 4. Negotiation :** Covered in Negotiations interface.

- 5. Update_order_status :** Done in class Order.

Sample test case :

Sales_history :

<None>

Pending_orders :

| | | | |
|---------|---------|--------------|---------|
| Order_1 | Buyer_1 | Price : 4000 | Pending |
| Order_2 | Buyer_1 | Price : 200 | Pending |
| Order_3 | Buyer_2 | Price : 1500 | Pending |

Choose Order_1, and accept it

Output :

Sales_history:

Order_1

Pending_orders :

| | | | |
|---------|---------|--------------|---------|
| Order_2 | Buyer_1 | Price : 200 | Pending |
| Order_3 | Buyer_2 | Price : 1500 | Pending |

Choose Order_2, and reject it

Output :

Sales_history:

Order_1

Pending_orders :

| | | | |
|---------|---------|--------------|---------|
| Order_3 | Buyer_2 | Price : 1500 | Pending |
|---------|---------|--------------|---------|

5) Class Manager:

1. Testing constructor:

- Construct with invalid parameters:
Manager class is inherited from user class. So it invokes the constructor of its parent class and any exceptions in the initialization of the parameters inherited from user class are handled by the user class constructor.

Exception handling for additional parameters is done by this class.

| | |
|--------|---------------|
| Gender | Date of Birth |
|--------|---------------|

| | |
|---|---|
| - | - |
|---|---|

OUTPUT: Exceptions raised for empty fields in Gender and Date of Birth.

Note that the gender and the date of birth are chosen from an enumeration in the form a drop-down and a calendar respectively, so there is no possibility of choosing an invalid gender or date of birth.

- Construct with valid parameters:

| Gender | Date of Birth |
|--------|----------------|
| Male | 8 October 2002 |
| Female | 21 August 2002 |

Output: OK, all parameters valid for both the objects. Creates 2 manager objects and adds them to the database.

2. Add category:

| |
|--------------------|
| List of Categories |
| Electronics |
| Clothes |
| Medicine |

Now we call the add category function with an input string which calls the constructor of Category class. Any exceptions in construction of object (like empty name) are handled by this constructor.

Input category name: Computers/Laptops

Output: A new Category class object is created and Computers/ Laptops is added to the list of categories.

| |
|--------------------|
| List of Categories |
| Electronics |
| Clothes |
| Medicine |

| |
|---------------------|
| Computers / Laptops |
|---------------------|

3. Change category:

| | | | | | | | |
|-----------|-------------|------------|-----------|------------|------------|-----------|------|
| HP laptop | Electronics | laptop.jpg | 5000 0 | 2 years | 14' screen | Hyderabad | True |
|-----------|-------------|------------|-----------|------------|------------|-----------|------|

Now, select any one of the available categories from the dropdown menu, say
Computers/Laptops.

Item category changed to Computers/Laptops successfully without any exception.

| | | | | | | | |
|-----------|------------------------|------------|-----------|------------|------------|-----------|------|
| HP laptop | Computers / Laptops | laptop.jpg | 5000 0 | 2 years | 14' screen | Hyderabad | True |
|-----------|------------------------|------------|-----------|------------|------------|-----------|------|

4. Remove category:

| |
|---------------------|
| List of Categories |
| Electronics |
| Clothes |
| Medicine |
| Computers / Laptops |

Call the remove_category function on any one of the available categories, say
Medicine.

OUTPUT :

| |
|---------------------|
| List of Categories |
| Electronics |
| Clothes |
| Computers / Laptops |

5. Reject Item:

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|--------------|-------------|-------------|-------|---------|-------------|-----------|----------|
| HP laptop | Electronics | laptop.jpg | 50000 | 2 years | 14' screen | Hyderabad | True |
| Levi's Jeans | Clothes | clothes.jpg | 1500 | - | 32" waist | Surat | False |

Call the reject item function for a particular item from the item list, say Levi's Jeans.

OUTPUT : Levi's Jeans is deleted successfully without any exceptions.

Item_list :

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|-----------|-------------|------------|-------|---------|-------------|-----------|----------|
| HP laptop | Electronics | laptop.jpg | 50000 | 2 years | 14' screen | Hyderabad | True |

6. Manage Seller:

Seller List

| Name | Email | Mobile Number |
|-------------------|---------------------|---------------|
| RK retail | rkretail@gmail.com | 9991234567 |
| Gupta Electronics | guptaece@gmail.com | 8887654321 |
| Keval medicine | kevaldava@gmail.com | 7775432190 |

Call the manage seller function for a particular seller from the seller list, say Keval medicine.

OUTPUT: Keval medicine is deleted successfully without any exceptions.

| Name | Email | Mobile Number |
|-----------|--------------------|---------------|
| RK retail | rkretail@gmail.com | 9991234567 |

| | | |
|-------------------|--------------------|------------|
| Gupta Electronics | guptaece@gmail.com | 8887654321 |
|-------------------|--------------------|------------|

7. Manage Buyer:

| Name | Email | Telephone |
|----------------|-----------------------------|------------|
| Anand Parikh | anandparikh_osp@gmail.com | 9988776655 |
| Aditya Soni | adityasoni_osp@gmail.com | 9898984541 |
| Shashwat Naidu | shashwatnaidu_osp@gmail.com | 9094436752 |

Call the manage buyer function for a particular buyer from the buyer list, say Shashwat Naidu.

OUTPUT: Shashwat Naidu is deleted successfully without any exceptions.

| Name | Email | Telephone |
|--------------|---------------------------|------------|
| Anand Parikh | anandparikh_osp@gmail.com | 9988776655 |
| Aditya Soni | adityasoni_osp@gmail.com | 9898984541 |

8. Negotiation: Tested in Negotiation Interface.

9. Audit : Prints all the finished transactions so far.

| Order info | Buyer info | Seller info |
|--------------------------|---|--|
| HP Laptop, Price = 45000 | Anand Parikh, anandparikh_OSP@gmail.com , 9988776655 | RK retail, rkretail@gmail.com , 9991234567 |
| Levi's Jeans = 2300 | Aditya Soni, adityasoni_OSP@gmail.com , 9898984541 | Gupta Electronics, guptaece@gmail.com , 8887654321 |

6) Class Item:

1. Testing constructor

- Constructor with invalid parameters:

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|------|----------|-------|-------|-----|-------------|-----------|----------|
| "" | Clothes | None | -100 | -10 | "" | Hyderabad | FALSE |

OUTPUT: Exception raised for empty fields in Name, Photo. Exception raised in price and age for entering negative numbers.

Note that city can never be given invalid input as it is selected from a dropdown menu provided.

- Constructor with valid parameters:

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|---------|-----------------|-----------|-------|-----|-------------|-----------|----------|
| "TV" | Electronics | tv.png | 2000 | 5 | "Samsung" | Kolkata | TRUE |
| "Mixer" | Home appliances | mixer.png | 500 | 3 | "LG" | Kharagpur | False |

OUTPUT: Items added successfully to the database with a unique id assigned to them.

2. **Search** : Covered in interface search.

7) Class Buyer:

1. The **constructor** invokes the parent's constructor (No new parameters to be passed)
2. **Raise_purchase_request:**

| Buyer name | Item | Price | Output |
|----------------|--------------|-------|--|
| Anand Parikh | Levi's jeans | -10 | Exception thrown by item class (as price is less than 0) |
| Aditya Soni | TV | 1000 | Successful |
| Shashwat Naidu | HP Laptop | 10000 | Successful |

3. **Update _Delivery_status:** Covered in Class Order

4. **Negotiation** : Covered in interfaces

8) **Class Order** :

1. **Testing constructor**

| Offer_price | Item | Buyer | Seller | City_1 == City_2 && Item.heavy | OUTPUT |
|-------------|--------|---------|----------|-----------------------------------|-----------|
| -1200 | Item_2 | Buyer_1 | Seller_1 | False | Exception |
| 1500 | Item_1 | Buyer_2 | Seller_3 | False | Created |
| 3000 | Item_1 | Buyer_3 | Seller_3 | True | Exception |

2. **Negotiate** : Explained in Negotiations Interface

3. **Update payment status** : This is called by a Seller object

- Accept - OUTPUT : Negotiations can now be done for this order.
- Reject - OUTPUT : Order has been rejected and removed from the database.
The buyer is notified about the same.

4. **Update delivery status** : Called by a Buyer object

| Item | Buyer | Seller | Price | Delivery Status |
|--------|---------|----------|-------|-----------------|
| Item_1 | Buyer_1 | Seller_3 | 5000 | Not delivered |

When Update delivery status is called for Item_1, the delivery status is changed successfully.

| | |
|---------|-----------|
| Order_1 | Delivered |
|---------|-----------|

INTERFACES :

9) **Sign Up interface**

- Some successful and unsuccessful sign-ups are provided below.
- Sample signup Key : c455@6bqwerty9\$0

| Type | Email-ID | Key (only for | OUTPUT |
|------|----------|---------------|--------|
|------|----------|---------------|--------|

| | | | |
|---------|-----------------------------|--------------------|-----------------------------------|
| | | Managers) | |
| Manager | anandparikh_OSP@gmail.com | c455@6bqwerty9\$0 | Accepted |
| Manager | adityasoni_OSP@gmail.com | bnh55\$tt67#90905b | Rejected (wrong key) |
| Buyer | anandparikh_OSP@gmail.com | - | Accepted |
| Manager | adityasoni_OSP@gmail.com | c455@6bqwerty9\$0 | Accepted |
| Seller | anandparikh_OSP@gmail.com | - | Accepted |
| Seller | shashwatnaidu_OSP@gmail.com | - | Accepted |
| Seller | shashwatnaidu_OSP@gmail.com | - | Rejected (account already exists) |
| Seller | adityasoni_OSP@gmail.com | - | Accepted |
| Buyer | anandparikh_OSP@gmail.com | - | Rejected (account already exists) |
| Buyer | shashwatnaidu_OSP@gmail.com | - | Accepted |

10) Sign-In interface

Database after above sign-up operations-

| User type | User mail-id | Login-ID | Password |
|-----------|-----------------------------|----------|---------------|
| Manager | anandparikh_OSP@gmail.com | M101 | qwerty567@# |
| Manager | adityasoni_OSP@gmail.com | M102 | 9****password |
| Seller | anandparikh_OSP@gmail.com | S101 | MNBOP_& |
| Seller | adityasoni_OSP@gmail.com | S102 | mypassword |
| Seller | shashwatnaidu_OSP@gmail.com | S103 | 18-01-2002 |
| Buyer | anandparikh_OSP@gmail.com | B101 | BNM+=<>789 |
| Buyer | shashwatnaidu_OSP@gmail.com | B102 | 12345 |

Different login attempt and outputs:

| User type | Login-ID | Current password | Entered password | OUTPUT |
|-----------|----------|------------------|------------------|------------------------------|
| Manager | M101 | qwerty567@# | qwerty345 | Failed (wrong password) |
| Manager | M101 | qwerty567@# | qwerty567@# | Successful |
| Manager | M102 | 9****password | 9****password | Successful |
| Seller | S102 | mypassword | myPASSWORD | Failed(wrong password) |
| Buyer | B104 | - | trial_pass | Failed (no such user exists) |
| Buyer | B101 | BNM+=<>789 | BNM+=<>789 | Successful |
| Seller | S113 | - | mypassword | Failed(no such user exists) |
| Buyer | B102 | 12345 | 123456 | Failed (wrong password) |
| Buyer | B102 | 12345 | password | Failed (wrong password) |
| Buyer | B102 | 12345 | 12345 | Failed (wrong password) |

11) Negotiation Interface

This interface can be called by both sellers and buyers appropriately for a pending purchase request.

Consider the following pending purchase requests in the database (just after being placed by a buyer, so no negotiations have started):

| Buyer | Seller | Item | Offering party | Offered price | Status |
|-------|--------|--------|----------------|---------------|---------|
| B102 | S101 | Item_1 | Buyer | 1200 | Pending |
| B101 | S102 | Item_3 | Buyer | 5000 | Pending |

Manager's interface to help in negotiations: All pending orders are printed and all the contact details are printed of the two parties.

| Buyer | Seller | Item | Buyer's contacts | Seller's contacts |
|-------|--------|--------|---|---|
| B102 | S101 | Item_1 | shashwatnaidu_OSP@gmail.com , 9094436752 | anandparikh_OSP@gmail.com , 9988776655 |

| | | | | |
|------|------|--------|--|--|
| B101 | S102 | Item_3 | anandparikh_OSP@gmail.com , 9988776655 | adityasoni_OSP@gmail.com , 9898984541 |
|------|------|--------|--|--|

Test case # 1 :

| Buyer | Seller | Item | Offering party | Offered price | Status |
|-------|--------|--------|----------------|---------------|---|
| B102 | S101 | Item_1 | Buyer | 1200 | Pending |
| B102 | S101 | Item_1 | Seller | 1000 | Pending |
| B102 | S101 | Item_1 | Buyer | 1100 | Pending |
| B102 | S101 | Item_1 | Seller | - | Rejected (seller rejects using update_status) |

After rejection, that purchase request is deleted from the database.

Test Case # 2 :

| Buyer | Seller | Item | Offering party | Offered price | Status |
|-------|--------|--------|----------------|---------------|---|
| B101 | S102 | Item_3 | Buyer | 5000 | Pending |
| B101 | S102 | Item_3 | Seller | 6000 | Pending |
| B101 | S102 | Item_3 | Buyer | 5750 | Pending |
| B101 | S102 | Item_3 | Seller | 5750 | Accepted (seller accepts using update_status) |

After acceptance, the option for payment is visible to the buyer on the portal interface.

12) Change Details Interface

This interface is accessed by the sellers to change the details of the products which they have uploaded on the portal.

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|------------------------|----------|----------|-------|---------|--------------|--------|----------|
| A Song of Ice and Fire | Books | book.jpg | 350 | 2 years | "GRR Martin" | Mumbai | False |

Seller uses the change details interface and modifies the following details:

| Detail | Old value | New value |
|--------|-----------|-----------|
| Price | 350 | 300 |
| Photo | book.jpg | book2.jpg |
| Age | 2 years | 3 years |

The item details have been changed successfully and the item is as shown:

| Name | Category | Photo | Price | Age | Description | City | Is_heavy |
|------------------------|----------|-----------|-------|---------|--------------|--------|----------|
| A Song of Ice and Fire | Books | book2.jpg | 300 | 3 years | "GRR Martin" | Mumbai | False |