Practice Test, Core Java

- All Questions are mandatory.
- Participants have to follow naming conventions.

Question 1:

Given File  StringPracticeTest.java .

Based on the file shared, there is class Documents which contains two documents PAN card and password number.

Verify the values of both properties during initialization time, below are the user requirements to validate the passport number and PAN card.

Pan Card should start with five letters then four numeric digits followed by one character. So based on that following variety of pan card numbers are valid or invalid

BJQPB6259N	Valid
ABCDE1234A	Valid
AAAAA1111A	Valid *(notice the occurrence of characters, numeric characters, and then character)*
AAAA123	Invalid
123ABCDE	Invalid

The passport should start from any of these single characters [A, B, C, D, or K] followed by six numeric digits followed by one single character, So based on that following variations of pan card numbers are valid or invalid

A123456B	Valid
A123456A	Valid
A111111A	Valid
123456A	Invalid
A123456	Invalid
AB123456AB	Invalid

If any of that document is an invalid PAN card number or passport then that document should be assigned to null.

Write that code to print the values of the Pan Card and Passport through the “Document” class object.

Use regular expression, Pattern, and Matcher class to implement the code.

Question 2

Below is the class structure of Customer and Address, implement the following use case through Stream API

Pre-requisites:- create an ArrayList of at least ten Customer records with different values.

1) Print all customers.
2) Return the list of all Customers from the city “New Delhi”.
3) Return the updated list of customers who belong to the city “Hydrabad” and the order value is greater than Rs. 1000/=, so for that customer order status should be changed to “Delivered”.
4) Return the customer with the maximum order value.

```java
package org.edu.app;

class Customer
{
    private String username;
    private String emailId;
    private Address address;
    private int orderValue;
    private String paymentType; // online , cash
    private String status; // delivered , pending ,  dispatched or cancelled
}
class Address
{
    private String houseNumber; // i.e ABC-123
    private String cityName;
    private String state;
}

public class StreamPracticeTest {

    public static void main(String[] args) {

    }
}

```

StreamPracticeTest.java

Question 3

Based on the given class structure , call following method . Note : While implementing the code you cannot change the class structure.

Call public float claimCashBack() from doUserThings() of PollymorphismDemo class

PollymorphismDemo.java

```java
package org.edu.pol;
abstract class Customer {
    public String username;
    public int billAmount;

    public abstract int payBill();
}
class OnlineCustomer extends Customer {
    public int payBill() {  return 1000; }
    public float claimCashBack() {
        return super.billAmount * 0.05f;
    }
}
public class PollymorphismDemo {
    public static void main(String[] args) {
        // call doUserThings methods
    }
    public void doUserThings(Customer customer)
    {
        // call claim cash back method
    }
}
```

Question 4:

MakeMyHoliday is a Hotel booking website, where users can search and book the hotels and admin of the website can add new hotel.

You have to implement the following use cases.

a) Insert hotel use case for Admin user.
b) Implement get hotel by city name for guest user. Which returns the map collection of Hotels based on String cityName.
c) Write new functionality in the HotelService class to return the List<Hotels> based on cityName & userRating above certain criteria.

```
class Hotels{

    String hotelName;
    String cityName;
    float userRatings;
}
class HotelService
{
    List<Hotels> allHotels;

    public boolean insertHotel(Hotels hotel) {
        return false;
    }
    public Map<String, List<Hotels>> getHotelsByCityName(String cityName)
    {
        return null;
    }
}
public class CollectionsPracticeTest {

    public static void main(String[] args) {
    }
}
```

d)

CollectionsPracticeTest.java

Question 5:

Based on the given class structures implement the following user case.

1) While creating the object of Customer class , if username is null or less than three characters then throw InvalidValidUsernameException and if balance is less than 0 then raise InvalidBalanceAmountException.
2) Customer can pay bill such as mobile bill payment or internet bill payment or recharge TV etc, wrt Customer has to call BillPaymentService class payBill() utility , which returns true if payment is successful . If Customer balance is less than the amount to be paid then raise the mentioned Exception.

```
3
4  class Customer
5  {
6      String username;
7      String balance;
8  }
9
10 class BillPaymentService
11 {
12     public boolean payBill(Customer customer,int amountToBePaid)throws InsufficientFundException
13     {
14         return false;
15     }
16 }
17
18
19 public class ExceptionHandlingPracticeTest {
20
21 }
22
```

ExceptionHandlingPracticeTest.java