**Assessment - Case Study**

An organization is interested in automating the process of SpeedPostSystem.

The system should be able to populate HeadOffice details and Pincodedetails.

Also to accept postId, postDate, receiptName, pinCode.

System should be able to mostOldestPost() according to date must be able to find 5 oldest post details.

System should be able to sortByAge() and write those details into “sortedPostDetails.txt”. This feature can be implemented in Client.java.

1. Implement populate() method as default method to populate details from file HeadOffice.txt and Post.txt.
2. Use forEach() method of Iterable interface to read show details in populate method.
3. In mostOldestPost() method used to find 5 oldest post -implement stream API to display 5oldestpost details.
4. Use Datetime api to check given postDate
5. Use BufferedReader.lines() to read to display sorted details into stream and process it.

**HeadOffice.java**

public class HeadOffice {

private String headOfficeId;

private String city;

private List<String> listOfPinCodes;

}

**Post.java**

public class Post {

private String postId;

private Date postDate;

private String recepientName;

private String pinCode;

}

**SpeedPost.java**

public interface SpeedPost {

Map<HeadOffice,List<Post>> populateData(String HeadOfficeFileName,String postFileName);

List<Post> mostOldestPost(Map<HeadOffice,List<Post>> map);//this will return the list of 5 post which oldest

HeadOffice trackPost(Map<HeadOffice,List<Post>> map,String postId);//this will return the associate HeadOffice

List<Post> sortByAge(Map<HeadOffice,List<Post>> map);

}

**SpeedPostImpl.java**

public class SpeedPostImpl implements SpeedPost {

public Map<HeadOffice, List<Post>> populateData(String HeadOfficeFileName,String postFileName) {

return map;

}

public List<Post> mostOldestPost(Map<HeadOffice, List<Post>> map) { return returnList.subList(0,5);

}

public HeadOffice trackPost(Map<HeadOffice, List<Post>> map, String postId) {

return null;

}

public List<Post> sortByAge(Map<HeadOffice, List<Post>> map) {

return returnList;

}

}

**Client.java**

public class Client {

public static void main(String[] args) {

SpeedPostImpl impl=new SpeedPostImpl();

Map<HeadOffice,List<Post>> map=impl.populateData("HeadOffice.txt", "Post.txt");

}

}