JA111 C4 Evaluation

Note:
There are a total of 5 questions.
All the questions are compulsory.
*Duration of the test is **3 hours.***
Mode of Submission: Github Link
Don't seek help from any person/resource during the test.
Marks Distribution is as follows:

Question	*Marks*
1	*4*
2	*3*
3	*4*
4	*5*
5	*4*

Q1) Create a basic application named Library Management System having a Library class containing a collection of books present in the library .Choose the best collection to store books.

Book class having the following private instance variables:

```
bookId :int
bookName : String
author : String
```

Note: Create a suitable parameterized constructor of this Book class.

In the main method of the Library class display the list of books. If the books are duplicate (i.e. Books with the same id) then in that case it should not be added to your collection.

Note: Books can have the same bookName,author but not bookId.

```
class Library{

   public static void main(String[] args) {

      //Use suitable collection to store books by the  name booksCollection

      booksCollection.add(new Book(1,"Name1","Author1"));
      booksCollection.add(new Book(2,"Name2","Author1"));
      booksCollection.add(new Book(3,"Name1","Author1"));
      booksCollection.add(new Book(1,"Name4","Author2"));

      //Implement the logic to iterate through the collection and print the books
    }
}
```

Sample Output:

```
Book{bookId=1, bookName='Name1', author='Author1'}
Book{bookId=2, bookName='Name2', author='Author1'}
Book{bookId=3, bookName='Name1', author='Author1'}
```

Q2) Explain about Hashcode and Equals Method.

Q3) Create a bean class Student with the following fields :

```
rollNo,
name,
mathsMarks,
scienceMarks,
engMarks.
```

In the main class implement the following logic:
You have to sort the collection of students on the basis of the total Marks,
If the totalMarks are equal then you have to sort on the basis of Name and if names are the same then sort on the basis of the rollNo.

NOTE: Name should be sorted in descending order.

Q4) Create a product class which has following fields -
-name : String
-price : double
-company : String
-count : int

Note : Override the toString method.

Create a class Ecommerce with the following fields-

-List<Product> productList=new ArrayList<>();
-addProductToList(Product product) : void
-showAllProduct():List<Product>

Write a Main class with a main method to test the following functionalities.

- If the product already exists in the list then update the count of the product else add it to the list.

Main method-

```
public static void main(String[] args) {
   Ecommerce ecommerce=new Ecommerce();
   ecommerce.addProduct(new Product("Shoes",2000,"Adidas",12));
   ecommerce.addProduct(new Product("Ipad",100000,"Apple",2));
   ecommerce.addProduct(new Product("Ipad",100000,"Apple",4));
   ecommerce.addProduct(new Product("Shoes",4000,"Nike",4));
   ecommerce.addProduct(new Product("Shoes",2000,"Adidas",8));
   System.out.println(ecommerce.showAllProduct());

}
```

Sample output-

```
Product added successfully
Product added successfully
```

Count updated Product already exists
Product added successfully
Count updated Product already exists

[Product{name='Shoes', price=2000, company='Adidas', count=20}, Product{name='Ipad', price=100000, company='Apple', count=6}, Product{name='Shoes', price=4000, company='Nike', count=4}]

Q5) Miss.Jane, an experienced English professor, gives practice tests to her students to improve their written skills. Every day students write an article and they submit it to Jane. Jane is particular that the students use only special characters like , ; : . ? ! in the article. She counts the total number of words used and the number of new words used by the students in the article. Based on the analysis done on the new words used by the students she gives her feedback to the students. Jane finds it difficult when the number of students increases. So she wanted to automate the process in the following format. Help her to write a java program to display the new words using lower case and in alphabetical order.

Sample Input and

Sample Input and Output - 1
Student's Article
“Hello Everybody, welcome to collection in JAVA. Collection, is like a container and powerful concept in Java!
Output:
Number of words 17 Number of unique words 14
The words are 1. a 2. and 3. collection 4. concept 5. container 6. everybody 7. hello 8. in 9. is 10. Java 11. like 12. powerful 13. to 14. Welcome

Sample Input and Output - 2
Student's Article:
“hello Hello HEllo hi hi: hi! Welcome, welcome”
Output:
Number of words 8 Number of unique words 3
The words are 1. hello 2. hi 3. welcome