Assignment 4

1. Write a java program to show the prices of the product. Design a class for the product with the product name and the price. Show the products to the user in the ascending order of the price or in the descending order of the price based on user's choice
   Hint: Make use of Collections.sort() and Collections.reverse(). You can create product instances in the main(). No need to take the product details from the user.

2. Consider the following code

```
CourseRegistration course = new CourseRegistration("CS101");

Student s1 = new Student("S001", "Alice", "alice@uni.edu");
Student s2 = new Student("S002", "Bob", "bob@uni.edu");
Student s3 = new Student("S003", "Charlie", "charlie@uni.edu");
Student s4 = new Student("S001", "Alice Smith",
"newemail@uni.edu"); // Same ID as s1

course.registerStudent(s1);  // true
course.registerStudent(s2);  // true
course.registerStudent(s3);  // true
course.registerStudent(s4);  // false (duplicate ID)

course.displayRegistrationOrder();
// Output:
// 1. Alice (S001) - alice@uni.edu
// 2. Bob (S002) - bob@uni.edu
// 3. Charlie (S003) - charlie@uni.edu

System.out.println(course.getWaitlistPosition("S002")); //
Output: 2
course.removeStudent("S002");
System.out.println(course.getWaitlistPosition("S003")); //
Output: 2 (moved up)
```

   Implement the functions registerStudent(), getWaitlistPosition() and removeStudent(). An id once added should not get added again. The order in which the registrations are done is to be preserved without storing any additional information.
   Hint: use LinkedHashSet

3. A library management software has the following class to hold the information of a book.

```
class Book {
    private String isbn;
    private String title;
```

```
    private String author;
}
```

There can be multiple copies of a book. The software should have a datastructure to manage the count of each book when added. addCopy(String isbn) function should help to add a copy of a book.
Hint: use HashMap

4. Write a java program to implement a Task Manager which uses a priority queue to manage a collection of tasks, each with a priority level. Task Manager should be able to perform the following operations:
    1. addTask(): add tasks to the priority queue.
    2. processTask(): process task will process the task and remove task based on their priority.
    3. displayTask(): display the current task in the queue.
    4. exit( ): exit from the program

   In the main method, create an instance of your TaskManager class and demonstrate the use of the methods by performing the following operation which is mentioned above.

Input/Output Format:

Select the operation:

1

Add Task with their priority and description

1, Meeting at 2pm

Select the operation:

1

Add Task with their priority and description

2, Buy groceries

Select the operation:

2

Select the operation:

3

Buy groceries

Select the operation:

4

5. Write a java program to iterate through a list of Flowers
   public class Flower {
      private String name;
      private String color;
   }
   Use Iterator to iterate through the instances.

6. You are building a dynamic pricing system for an e-commerce platform. The system needs to categorize products into price tiers.
   BUDGET: Rs.0 - Rs. 100
   STANDARD: Rs.101 - Rs.500
   PREMIUM: Rs.501 - Rs. 1500
   LUXURY: Rs. 1501+

   When the user inputs a price, the program should be able to find the name of the price range.
   Hint: Use TreeMap.