SMS with HashMap:

Step1: Create Student Class.

Step2: Create Interface StudentOperations

Step3: Create an implementation class StudentOperationsImpl

Step4: Create a StudentManagementSystem class with main() method.

TreeMap:

Key Features:

- Sorted Order : Key are maintained in natural order(Ascending)
- Unique Keys: Keys must be unique, if we are attempting to insert duplicate keys it will overwrite the existing value.
- Null Handling:
  - Keys: TreeMap does not allow null keys
  - Values: It allows multiple null values
- Thread Safety:
  - It is not synchronized
  - For thread-safety we can wrap it inside Collections.synchronizedMap

What is the difference between collection and collections?

Commonly used methods:

Put(K key, V value)

Get(Object key)

firstKey()

lastKey()

```java
package mapExample;



import java.util.TreeMap;

public class TreeMapExample {
    public static void main(String[] args) {
        TreeMap<Integer,String> treeMap = new TreeMap<>();

        //Adding elements
        treeMap.put(3,"Three");
        treeMap.put(1, "one");
        treeMap.put(2,"Two");
        treeMap.put(4,"Four");
        treeMap.put(1, "Duplicate");

        //Displaying the map
        System.out.println("TreeMap : "+ treeMap);

        // Accessing the elements
        System.out.println("Value for key 2 : "+
treeMap.get(2));

        // Removing an element
        treeMap.remove(2);
        System.out.println("After Removal :"+ treeMap);


        System.out.println("First Key: "+ treeMap.firstKey());
        System.out.println("Last Key:  " + treeMap.lastKey());

    }
}
```

o/p:

TreeMap : {1=Duplicate, 2=Two, 3=Three, 4=Four}

Value for key 2 : Two

After Removal :{1=Duplicate, 3=Three, 4=Four}

First Key: 1

Last Key: 4

TreeMap (Java SE 21 & JDK 21)

TASK: What are the multiple ways
To reverse the order