Name: Abdullah Al Mamun Sagor
ID: IT22059

1. Find the Kth smallest element in ArrayList.

```
import Java. util. * ;
public class KthSmallestElement {
   public static int findKthSmallest (List<Integer> List, int k){
       collection. sort (List) ;
       return list. get (k-1) ;
   }
 public static void main (String [] args){
   List <Integer> number= Array.asList (7,10,14,3,20,15);
    int k = 3;
  System.out.println (" The " +k+ "rd smallest
                element is " + findKthSmallest (numbers, k));
 }
}
```

2. Tree map to stored word frequency

```
import Java.util.*;
public class WordfrequencyTreeMap {
  public static void main (String [] args){
String text = "Hello world hello Java world java Java";
  String [] words = text. split (regex: " ");
TreeMap<String, Integer> FrequencyMap = new TreeMap<>();
 for (string word : words) {
     frequencyMap. put (word, frequencyMap..getOrDefault
                              (word, defaultValue:0) + 1);
 }

 System. out. println ("Word Frequencies: ");
 for (Map. Entry <String, Integer> entry:
                      frequencyMap. entrySet()) {
  System.out. println (entry. getkey()+": "+ entry
                                      getvalue ()) ;
  }
 }
}
```

3. Implement a queue and stack using priority queue.

```
import java.util.*;
public class PriorityQueueStackQueue{
  static class QueueUsing PriorityQueue {
  private PriorityQueu <int[]> pq = new
      priorityQueue<>(comperator.comparingInt(int[]a->a[0]));
     private int count = 0;
public void add (int value) {
 pq. add (new int[]{count++, value});
}
```

IT22059

```
public int remove () {
   return pq. poll () [1] ;
  }
 }

pr static class StackUsingPriorityQueu {
    private Priority Queu <int[]> pq= new PriorityQueu
                                   <> ((int[]a, [int[]b)
                        - integer. compare(b[0], a[0]));
    private int count =0 ;
  public void push (int value) {
      pq. add (new. int[] {count++, value});
   }
  public int pop () {
     return pq. poll () [1];
   }
 }

public static void main (string [] args) {
   Queue UsingPriorityQueu queu = new QueeUsingPriority
                                              Queue();
    queue. add (10);
    queue. add (20);
    queue. add (30);
  System. out. printh ("Queu: ");
 System. out. println (queue. remove ());
 System. out. println (queue. remove ());
```

IT 22059

```
Stack Using Priority Queu stack = new StackUsingPriorityQueue()
      Stack push (10);
      stack (push (20);
      stack (push (30);
   System. out. println ("Stack:");
   System. out. println (stack . pop ());
   System. out. println (stack . pop ());
   }
}
```

4. Treemap mapping the student ID

```
import java .util. * ;
  Class student {
  String name;
    int age;
 Student (string name, int age) {
     this name = name;
     this age = age;
 }
 public String toString () {
    return name + " (" +age + " years)";
 }
}
```

```
public class StudentDetailsTreeMap {
   public static void main (String [] args ){
 TreeMap <Integer , Student> Student Map= new Treemap<>();
 StudentMap.put (101, new student ("Alice" , 20));
 StudentMap.put (105, new student ("Bob", 22));
 StudentMap.put (102 , new Student ("Charlie", 21));
   for (Map.Entry <Integer, student> entry :
                           StudentMap.entry set()) {
   System.out.println ( "ID: " + entry.getKey ()+
                          " - " +entry.getValue ());
  }
 }
}
```

5. Linked List equality check:

```
import Java util.*;
public class LinkelistEqualityChecker {
   public static boolean areEqual (LinkedList <Integer> L1,
                                  LinkedList <Integer> L2){
      return L1, equals (L2);
  }

public static void main (String [] args) {
  LinkedList1<Integer> List1 = new LinkedList <>
                             Arrays. asList (1, 2, 3));
LinkedList 2 = newLinkedList <> Arrays. asList (1,2,3));
Linkelist 3 =. newLinkedList <> Arrays. as.List (3,2,1));

System.out. println ("List1 equals List2? "+ areEqual
                                        (List1, List2));
System.out.. println("List1 equals List 3 ?" + are Equal
                                        (List1, List 3));
  }
}
```

6. Create a HashMap to store the employee IDs to their departments.

```
import Java. util. * ;
public class EmployeeDepatrtment Map{
   public static void main(string [] args) {
 HashMap< string, String> empDept = new Hashmap< >();
      empDept. put ("E001", "HR");
      empDept. put ("E002", "Finance");
      empDept put ("E003" "IT");
for (Map. Entry<String, String> Entry :
                         empDept. entry Set()){
System. out. println (" Employee ID; " + entry - getkey()+
  }                 Department : " +entry. getValue());
 }
}
```