Name: Papon Biswas

ID: IT-22019

1. kth smallest element in an arraylist

```
import java .util.*;
plublic class kthsmallest {
public static int findkthsmallest (Arraylist< integer >list, int k){
    collections.sort (list);
    return list.get (k-1);
  }
  public static void main (String[] args){
    Arraylist <integer> list=New arraylist <>(array.aslist (7,2,1,6,8));
    int k = 3;
    system.out.println ("kth smallest :" + findkthsmallest (list, k));
  }
}
```

2. Treemap to map words to their frequencies.

```
→ import java.util.*;
  public class wordfrequency {
    public static void main (string[] args) {
      String text = " apple banana apple orange banana app
      string[] words = text.split(" ");

    Treemap <string, integer> frequencymap = new Treemap <> ()
    for (string word : words) {
      frequencymap. put (word, frequencymap. getorDefault (word,0)
    }
    System.out.println("Word Frequency:" + frequencymap);

  }
}
```

3. Implement Queue and stack using priorityQueue with custom comparator :

→

```
import java.util.*;
public class prioritystructure {
  public static void main (string[] args) {
    priorityQueue <integer> stack = new priorityQueue<>(comparator.reverseorder());
    stack. addAll (Array.aslist (1,2,3,4));
    while (! stack .isEmpty ()) {
      system.out.println("Stack pop :" + stack .poll ());
    }
    priorityQueue <integer> queue = new priorityQueue< >(comparator.naturalorder());
    Queue . addAll (Array.aslist (4,3,2,1));
    while (! queue. isEmpty) {
      System.out.println ("Queue poll : "+ queue .poll());
    }
  }
}
```

4. ~~Ma~~ Treemap of student IDs to their details;

⇒

```
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 +")";
 }
}
public class studentMap {
 public static void main (string[] args) {
  Treemap<integer, Student> students = new Treemap<>();
  students.put (102, new student("Alice", 20));
  students.put (101, new student ("Bob", 21));
  students put (103, new student ("Charlie", 19));
  for (Map. ~~Entry~~ Entry < integer, student > Entry : students.entry
{
system.out.println ("ID:"+entry getkey ()+ ", Info: "+entry.
        getvalue ());
    }
  }
}
```

5 Hashmap of employee IDs to department.

```
=> import java.util.* ;
 public class Employeemap {
  public static void main (string [] args ){
    Hashmap < Integer, string > employeeDepartment = new Hashmap<>();
   employeeDepartment .put (1001, "HR");
   employeeDepartment.put (1002, "IT");
  employeeDepartment .put (1003, "Finance");
 for (Map.Entry<Integer, string >entry : employeeDepartment.
                                              entrySet ()) {
 System.out.println ("Employee ID :"+ Entry.get key()+"
       , Department :"+ entry. getvalue ()) ;
   }
  }
}
```