1. Consider the abstract class Teacher presented below to implement what is being asked in FullTimeTeacher and PartTimeTeacher classes.

public abstract class Teacher {

private static int idGenerator = 1;

private int id;

private String subject;

private int totalHoursWeek;

public Teacher(){

this("", 0);

}

public Teacher(String subject, int hours){

setSubject(subject);

setTotalHoursWeek(hours);

id = idGenerator;

idGenerator++;

}

public void setSubject(String subject){

this.subject = subject;

}

public String getSubject(){

return subject;

}

public void setTotalHoursWeek(int hours){

totalHoursWeek = hours;

}

public int getTotalHoursWeek(){

return totalHoursWeek;

}

public abstract double getSalary();

public String toString(){

return String.format("ID: %d Subject: %s Total Hours of Week: %d", id, subject, totalHoursWeek);

}

}

public class PartTimeTeacher extends Teacher{

private double hourSalary;

public PartTimeTeacher(){

this("",0, 0);

}

public PartTimeTeacher(String subject, int hours, double hourSalary){

//implements this constructor

}

public void setHourSalary(double value){

hourSalary = value;

}

public double getHourSalary() {

return hourSalary;

}

public double getSalary(){

//implement this method

}

public String toString(){

return super.toString() + "Hours salary: " + hourSalary;

}

}

public class FullTimeTeacher extends Teacher {

private double salary;

public FullTimeTeacher(){

this("", 0, 0);

}

public FullTimeTeacher(String subject, int hours, double salary){

//implements this method

}

public void setSalary(double salary){

this.salary = salary;

}

public double getSalary(){

//implements this method

}

public String toString(){

return super.toString() + "Salary: " + salary;

}

}

1. Analyze the classes below and implement what is being asked.

import java.io.File;  
import java.io.FileNotFoundException;  
import java.util.ArrayList;  
import java.util.Collections;  
import java.util.Scanner;  
public class ListCountry {  
 ArrayList<Country> countries = new ArrayList<Country>();  
  
 public void readCountries(String filename) {  
 try {  
 Scanner scnr = new Scanner(new File(filename));  
 while (scnr.hasNextLine()) {  
 String name = scnr.next();  
 double area = scnr.nextInt();  
 countries.add(new Country(name, area));  
 }  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
  
 public void sortList(String check) {  
 if(check.equals("name"))  
 Country.*compareData* = true;  
 else Country.*compareData* = false;  
 Collections.*sort*(countries);  
}  
 public String toString() {  
 StringBuilder s = new StringBuilder();  
 for(Country c : countries) {  
 s.append(c);  
 s.append('\n');  
 }  
 return s.toString();  
 }  
 public static void main(String args[]){  
 ListCountry listCountry = new ListCountry();  
 listCountry.readCountries("countries.txt");  
 listCountry.sortList("name");  
 System.*out*.println(listCountry);  
 listCountry.sortList("area");  
 System.*out*.println(listCountry);  
 }  
}

public class Country implements Comparable<Country> {  
 private String name;  
 private double area;  
 public static boolean *compareData*;  
  
 public Country(String name, double area) {  
 this.name = name;  
 this.area = area;  
 }  
  
 public double getArea() {  
 return area;  
 }  
  
 public String getName() {  
 return name;  
 }  
  
 @Override  
 */\*\*  
 \* if compareData is true - compare Country by names  
 \* if compareData is false - compare Country by area  
 \*/* public int compareTo(Country o) {  
 //implement this method

}  
  
 @Override  
 public boolean equals(Object obj) {  
 if(this == obj) return true;  
 //finalize the implementation of this method

}  
  
 public String toString() {  
 return "name: " + name + " area: " + area;  
 }  
}

Assume that countries.txt has this content:

Brazil 3288000  
USA 3797000  
Portugal 35603  
India 1269000