



Assignment Solutions till 13

[Browse files](#) gunjan Gunjan02 committed 16 days ago1 parent [a73d968](#)commit [f5408b59674a78a9a53b979b1235c461ae48b06a](#) Showing 12 changed files with 410 additions and 0 deletions.[Unified](#) [Split](#)

0 ■■■■■ Assignment10-Solutions/1.java

No changes.

75 ■■■■■ Assignment10-Solutions/2.java

```
...    ...    @@ -0,0 +1,75 @@
1      +package f;
2      +
3      +import java.io.File;
4      +import java.io.FileInputStream;
5      +import java.io.FileNotFoundException;
6      +import java.io.FileReader;
7      +import java.io.FileWriter;
8      +import java.io.IOException;
9      +import java.io.PrintStream;
10     +import java.util.*;
11     +
12     +public class FileListingApp{
13     +
14     +    public static void Files(String path,ArrayList al) {
15     +        System.out.println(path);
16     +        File f=new File(path);
17     +        File f1[]=f.listFiles();
18     +
19     +        for(File file:f1) {
20     +
21     +            if(file.isDirectory()==true)
22     +            {
23     +
24     +                Files(file.getAbsolutePath(),al);
25     +            }else {
26     +                al.add(file);
27     +            }
28     +        }
29     +    }
30     +
31     +    public static void main(String args[]) throws IOException{
32     +        Scanner s=new Scanner(System.in);
33     +        System.out.println("Enter a path: ");
34     +        String loc=s.nextLine();
35     +        String path1="";
36     +        String path2="";
37     +        ArrayList<File> al=new ArrayList<File>();
38     +        FileReader in = null;
39     +        try {
40     +            in=new FileReader(loc);
41     +            int ch;
42     +
43     +            while((ch=in.read())!=10) {
44     +                path1=path1+(char)ch;
45     +            }
46     +            System.out.println(path1);
47     +
48     +            while((ch=in.read())!=-1) {
49     +                path2=path2+(char)ch;
50     +            }
51     +            System.out.println(path2);
```

```

52 +         path1=path1.substring(0,path1.length()-1);
53 +         Files(path1,a1);
54 +         File file2=new File(path2);
55 +
56 +         FileWriter out=new FileWriter(file2,true);
57 +
58 +
59 +         for(int i=0;i<a1.size();i++) {
60 +             out.write("Name: "+a1.get(i).getName());
61 +             out.write(',');
62 +             out.write("Path: "+a1.get(i).getAbsolutePath());
63 +             out.write(10);
64 +             out.flush();
65 +         }
66 +     }catch(Exception e) {
67 +         System.out.println("Wrong Input");
68 +     }finally {
69 +         if(in!=null)
70 +             in.close();
71 +     }
72 +
73 +
74 +     }
75 +}

```

22 ■■■■■ Assignment11-Solutions/1.java

```

...  ...  @@ -0,0 +1,22 @@
1  +import java.util.*;
2  + public class Exercise {
3  + public static void main(String[] args) {
4  +     // Create an empty hash set
5  +     HashSet<String> h_set1 = new HashSet<String>();
6  +     // use add() method to add values in the hash set
7  +     h_set1.add("Red");
8  +     h_set1.add("Green");
9  +     h_set1.add("Black");
10 +     h_set1.add("White");
11 +     System.out.println("First HashSet content: "+h_set1);
12 +     HashSet<String>h_set2 = new HashSet<String>();
13 +     h_set2.add("Red");
14 +     h_set2.add("Pink");
15 +     h_set2.add("Black");
16 +     h_set2.add("Orange");
17 +     System.out.println("Second HashSet content: "+h_set2);
18 +     h_set1.retainAll(h_set2);
19 +     System.out.println("HashSet content:");
20 +     System.out.println(h_set1);
21 + }
22 +}

```

14 ■■■■■ Assignment11-Solutions/2.java

```

...  ...  @@ -0,0 +1,14 @@
1  +import java.util.*;
2  +public class Example {
3  + public static void main(String args[]) {
4  +     HashMap<Integer,String> hash_map= new HashMap<Integer,String>();
5  +     hash_map.put(1, "Red");
6  +     hash_map.put(2, "Green");
7  +     hash_map.put(3, "Black");
8  +     hash_map.put(4, "White");
9  +     hash_map.put(5, "Blue");
10 +     for(Map.Entry x:hash_map.entrySet()){
11 +         System.out.println(x.getKey()+" "+x.getValue());
12 +     }
13 + }
14 +}

```

105 ■■■■ Assignment12-Solutions/1.java

```
...    ...    @@ -0,0 +1,105 @@

1  +import java.util.*;
2  +
3  +public class date {
4  +
5  +    public static HashMap<String, String> date = new HashMap<String, String>();
6  +    public static HashMap<String, String> month = new HashMap<String, String>();
7  +    public static HashMap<String, String> yearLeft = new HashMap<String, String>();
8  +    public static HashMap<String, String> yearRight = new HashMap<String, String>();
9  +
10 +    public static void genData() {
11 +        date.put("01", "First");
12 +        date.put("02", "Second");
13 +        date.put("03", "Third");
14 +        date.put("04", "Fourth");
15 +        date.put("05", "Fifth");
16 +        date.put("06", "Sixth");
17 +        date.put("07", "Seventh");
18 +        date.put("08", "Eighth");
19 +        date.put("09", "Nineth");
20 +        date.put("10", "Tenth");
21 +        date.put("11", "Eleventh");
22 +        date.put("12", "Twelveth");
23 +        date.put("13", "Thirteenth");
24 +        date.put("14", "Fourteenth");
25 +        date.put("15", "Fifteenth");
26 +        date.put("16", "Sixteenth");
27 +        date.put("17", "Seventeenth");
28 +        date.put("18", "Eighteenth");
29 +        date.put("19", "Nineteenth");
30 +        date.put("20", "Twenty");
31 +        date.put("21", "Twenty First");
32 +        date.put("22", "Twenty Second");
33 +        date.put("23", "Twenty Third");
34 +        date.put("24", "Twenty Fourth");
35 +        date.put("25", "Twent Fifth");
36 +        date.put("26", "Twenty Sixth");
37 +        date.put("27", "Twenty Seventh");
38 +        date.put("28", "Twenty Eighth");
39 +        date.put("29", "Twenty Nineth");
40 +        date.put("30", "Thirty");
41 +        date.put("31", "Thirty First");
42 +
43 +        month.put("01", "January");
44 +        month.put("02", "February");
45 +        month.put("03", "March");
46 +        month.put("04", "April");
47 +        month.put("05", "May");
48 +        month.put("06", "June");
49 +        month.put("07", "July");
50 +        month.put("08", "August");
51 +        month.put("09", "September");
52 +        month.put("10", "October");
53 +        month.put("11", "November");
54 +        month.put("12", "December");
55 +
56 +        yearLeft.put("07", "Seven");
57 +        yearLeft.put("08", "Eight");
58 +        yearLeft.put("09", "Nine");
59 +        yearLeft.put("10", "Ten");
60 +        yearLeft.put("11", "Eleven");
61 +        yearLeft.put("12", "Twelve");
62 +        yearLeft.put("13", "Thirteen");
63 +        yearLeft.put("14", "Fourteen");
64 +        yearLeft.put("15", "Fifteen");
65 +        yearLeft.put("16", "Sixteen");
66 +        yearLeft.put("17", "Seventeen");
67 +        yearLeft.put("18", "Eighteen");
```

```

68 +         yearLeft.put("19", "Nineteen");
69 +         yearLeft.put("20", "Twenty");
70 +         yearLeft.put("21", "Twenty One");
71 +         yearLeft.put("22", "Twenty Two");
72 +
73 +         yearRight.put("00", "Hundred");
74 +         yearRight.put("10", "Ten");
75 +         yearRight.put("20", "Twenty");
76 +         yearRight.put("30", "Thirty");
77 +         yearRight.put("40", "Forty");
78 +         yearRight.put("50", "Fifty");
79 +         yearRight.put("60", "Sixty");
80 +         yearRight.put("70", "Seventy");
81 +         yearRight.put("80", "Eighty");
82 +         yearRight.put("90", "Ninety");
83 +     }
84 +
85 +     public static String convert(String dateInput) {
86 +         String d = dateInput.substring(0, 2);
87 +         String m = dateInput.substring(3, 5);
88 +         String yL = dateInput.substring(6, 8);
89 +         String yR = dateInput.substring(8, 10);
90 +         return date.get(d) + " " + month.get(m) + " " + yearLeft.get(yL) + " " + yearRight.get(yR);
91 +     }
92 +
93 +     public static void main(String args[]) {
94 +
95 +         genData();
96 +
97 +         System.out.println("22-02-1930");
98 +         System.out.println(convert("22-02-1930"));
99 +
100 +         System.out.println("\n21-11-1990");
101 +         System.out.println(convert("21-11-1990"));
102 +
103 +
104 +     }
105 + }

```

26 ■■■■■ Assignment12-Solutions/2.java

```

...  ...  @@ -0,0 +1,26 @@
1  +package s;
2  +
3  +import java.util.ArrayList;
4  +import java.util.Collections;
5  +
6  +public class Sort {
7  +
8  +     public static void main(String[] args) {
9  +         // TODO Auto-generated method stub
10 +         ArrayList<String> s=new ArrayList<String>();
11 +         s.add("Gunjan");
12 +         s.add("Ambika");
13 +         s.add("Ananyaa");
14 +         Collections.sort(s);
15 +         System.out.println(s);
16 +
17 +         ArrayList<Integer> i=new ArrayList<Integer>();
18 +
19 +         i.add(1);
20 +         i.add(22);
21 +         i.add(3);
22 +         Collections.sort(i);
23 +         System.out.println(i);
24 +     }
25 +
26 + }

```

24 ■■■■■ Assignment12-Solutions/3.java

```

...  ...  @@ -0,0 +1,24 @@
1  +public class MainClass {
2  + // generic method printArray
3  + public static <E> void printArray(E[] inputArray) {
4  + // display array elements
5  + for (E element : inputArray)
6  +     System.out.printf("%s ", element);
7  +
8  +     System.out.println();
9  + }
10 +
11 + public static void main(String args[]) {
12 + // create arrays of Integer, Double and Character
13 + Integer[] integerArray = { 1, 2, 3, 4, 5, 6 };
14 + Double[] doubleArray = { 1.1, 2.2, 3.3, 4.4, 5.5, 6.6, 7.7 };
15 + Character[] characterArray = { 'H', 'E', 'L', 'L', 'O' };
16 +
17 + System.out.println("Array integerArray contains:");
18 + printArray(integerArray); // pass an Integer array
19 + System.out.println("\nArray doubleArray contains:");
20 + printArray(doubleArray); // pass a Double array
21 + System.out.println("\nArray characterArray contains:");
22 + printArray(characterArray); // pass a Character array
23 + } // end main
24 +}

```

60 ■■■■■ Assignment12-Solutions/4.java

```

...  ...  @@ -0,0 +1,60 @@
1  +import java.util.*;
2  +import java.io.*;
3  +
4  +class Simple{
5  +public static void main(String args[]){
6  +
7  +    ArrayList al=new ArrayList();
8  +    al.add(new Student(101,"Vijay",23));
9  +    al.add(new Student(106,"Ajay",27));
10 +    al.add(new Student(105,"Jai",21));
11 +
12 +    System.out.println("Sorting by Name...");
13 +
14 +    Collections.sort(al,new NameComparator());
15 +    Iterator itr=al.iterator();
16 +    while(itr.hasNext()){
17 +        Student st=(Student)itr.next();
18 +        System.out.println(st.rollno+" "+st.name+" "+st.age);
19 +    }
20 +
21 +    System.out.println("sorting by age...");
22 +
23 +    Collections.sort(al,new AgeComparator());
24 +    Iterator itr2=al.iterator();
25 +    while(itr2.hasNext()){
26 +        Student st=(Student)itr2.next();
27 +        System.out.println(st.rollno+" "+st.name+" "+st.age);
28 +    }
29 +
30 +
31 +
32 +
33 +
34 +
35 +    ///Comparing on the basis on Name
36 +    import java.util.*;
37 +    class NameComparator implements Comparator{
38 +    public int compare(Object o1,Object o2){
39 +        Student s1=(Student)o1;
40 +        Student s2=(Student)o2;
41 +

```

```

42 +return s1.name.compareTo(s2.name);
43 +}
44 +}
45 +
46 +//Comparing on the basis on Age
47 +import java.util.*;
48 +class AgeComparator implements Comparator{
49 +public int compare(Object o1,Object o2){
50 +Student s1=(Student)o1;
51 +Student s2=(Student)o2;
52 +
53 +if(s1.age==s2.age)
54 +return 0;
55 +else if(s1.age>s2.age)
56 +return 1;
57 +else
58 +return -1;
59 +}
60 +}

```

45 ■■■■■ Assignment13-Solutions/1.java

```

...  ...  @@ -0,0 +1,45 @@
1  +package t;
2  +
3  +public class Task extends Thread{
4  + int i=1;
5  + public synchronized void run() {
6  + while(i<=500) {
7  + System.out.println(i);
8  + i++;
9  + }
10 + try {
11 + Thread.sleep(100);
12 + }catch(InterruptedException e) {
13 + e.printStackTrace();
14 + }
15 +
16 + }
17 +
18 + public static void main(String[] args) throws InterruptedException {
19 + // TODO Auto-generated method stub
20 + Task task1=new Task();
21 + Thread t1=new Thread(task1);
22 + t1.start();
23 + Task2 task2=new Task2();
24 + Thread t2=new Thread(task2);
25 + t1.join();
26 + t2.start();
27 + t2.join();
28 +
29 + }
30 +
31 +}
32 +class Task2 implements Runnable{
33 + int i=501;
34 + public synchronized void run() {
35 + while(i<=1000) {
36 + System.out.println(i);
37 + i++;
38 + }
39 + try {
40 + Thread.sleep(100);
41 + }catch(InterruptedException e) {
42 + e.printStackTrace();
43 + }
44 + }
45 +}

```

37 ■■■■■ Assignment9-Solutions/1.java

... ... @@ -0,0 +1,37 @@

1 +package f;

2 +import java.io.File;

3 +import java.io.FileReader;

4 +import java.io.FileWriter;

5 +import java.io.IOException;

6 +import java.util.*;

7 +import java.util.InputMismatchException;

8 +public class File1 {

9 +

10 +

11 + public static void main(String[] args) throws IOException {

12 + // TODO Auto-generated method stub

13 + File file=new File("C:\\Users\\Kanchan Panda\\Desktop\\xyz.txt");

14 + FileReader in = null;

15 + FileWriter out=new FileWriter("C:\\Users\\Kanchan Panda\\Desktop\\abc.txt");

16 + try {

17 + in=new FileReader(file);

18 + int ch;

19 + String a="";

20 + while((ch=in.read())!=-1) {

21 + System.out.print((char)ch);

22 + a=a+(char)ch;

23 + }

24 + char b[]=a.toCharArray();

25 + for(int i=0;i<b.length;i++){

26 + out.write(b[i]);

27 + }

28 + out.close();

29 + }catch(InputMismatchException e) {

30 + System.out.println("Wrong Input");

31 + }finally{

32 + if(in!=null)

33 + in.close();

34 + }

35 + }

36 +

37 +}

1 ■■■■■ Assignment9-Solutions/abc.txt

... ... @@ -0,0 +1 @@

1 +C:\\Users\\Kanchan Panda\\Desktop\\Kanchan

1 ■■■■■ Assignment9-Solutions/xyz.txt

... ... @@ -0,0 +1 @@

1 +C:\\Users\\Kanchan Panda\\Desktop\\Kanchan

0 comments on commit

f5408b5