

Code snippets

1) Sort the Array List in ascending and descending order (Code Chef)

The screenshot shows the CodeChef problem page for 'Array construction'. The problem statement asks to output an array A sorted in a descending order. The input format specifies that the first line contains an integer T , denoting the number of test cases, and each test case consists of a single line of input - the integer N . A sample input shows $T=2$ and $N=3$ for the first test case, resulting in the output $[1, 2, 3]$ and $[3, 2, 1]$. The explanation states that for test case 1, N is 3, and $A = [1, 2, 3]$.

The solution is written in Java and uses a Scanner to read input. It uses a while loop to read integers t and b . For each t , it creates an ArrayList, adds b to it, and then sorts the list in ascending order using `Collections.sort(list)` and prints it. Finally, it reverses the list using `Collections.reverse(list)` and prints it again.

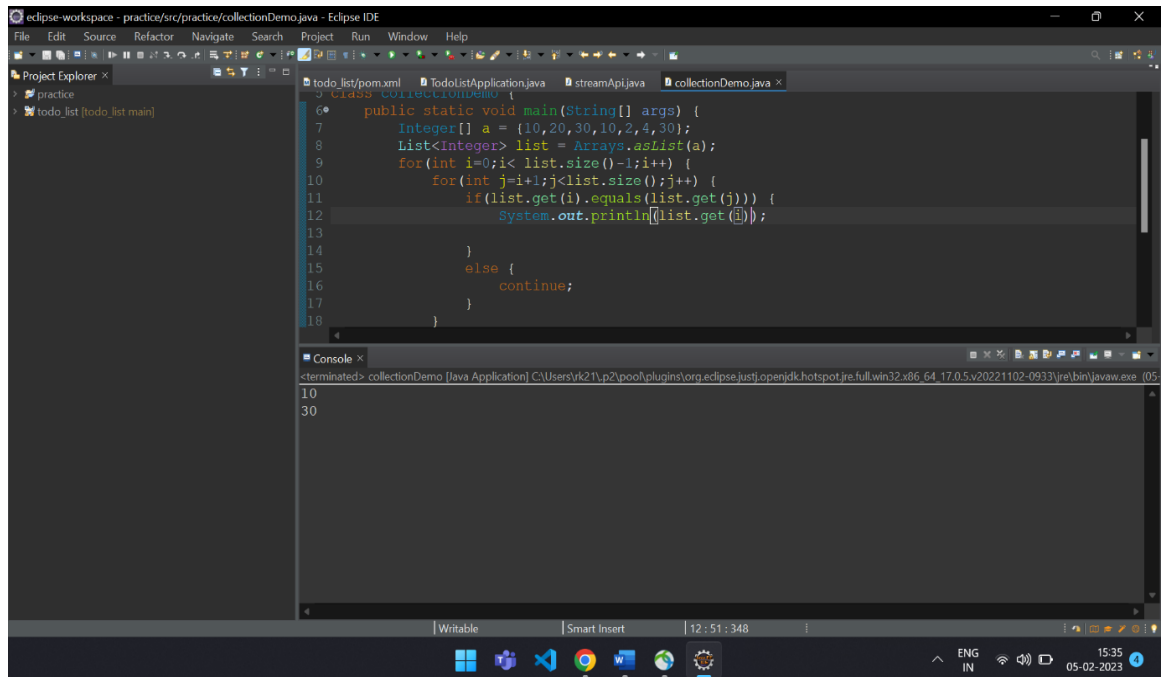
```
7 Scanner sc = new Scanner(System.in);
8 int t = sc.nextInt();
9 while(t-->0){
10     int b = sc.nextInt();
11     ArrayList<Integer> list = new ArrayList<>();
12     list.add(b);
13     while(b-->1){
14         list.add(b);
15     }
16     Collections.sort(list);
17     System.out.println(list);
18     Collections.reverse(list);
19     System.out.println(list);
20 }
21
```

2) Chef has a string S with him. Chef is happy if the string contains a **contiguous substring** of length **strictly greater** than 22 in which all its characters are vowels.

The screenshot shows a CodeChef solution for a problem. The solution is written in Java and uses a Scanner to read input. It reads a string S and checks if it contains a contiguous substring of length strictly greater than 22 in which all its characters are vowels. The solution uses a loop to iterate over the string and checks if the substring contains only vowels. If it does, it prints "Happy"; otherwise, it prints "Sad".

```
11 {
12     Scanner sc = new Scanner(System.in);
13     int total = sc.nextInt();
14     for(int i=0; i<total; i++){
15         String str = sc.next();
16         ArrayList<Character> list = new ArrayList<>();
17         list.add('a');
18         list.add('e');
19         list.add('i');
20         list.add('o');
21         list.add('u');
22         int count=0;
23         int max=0;
24         for(int j=0; j<str.length(); j++){
25             if(list.contains(str.charAt(j))){
26                 count++;
27                 if(count==3){
28                     max=3;
29                 }
30             }
31             else{
32                 count=0;
33             }
34         }
35         if(max>=3){
36             System.out.println("Happy");
37         }
38         else{
39             System.out.println("Sad");
40         }
41     }
42 }
43
```

3) Ways to Find Duplicate Elements in each Array in Java



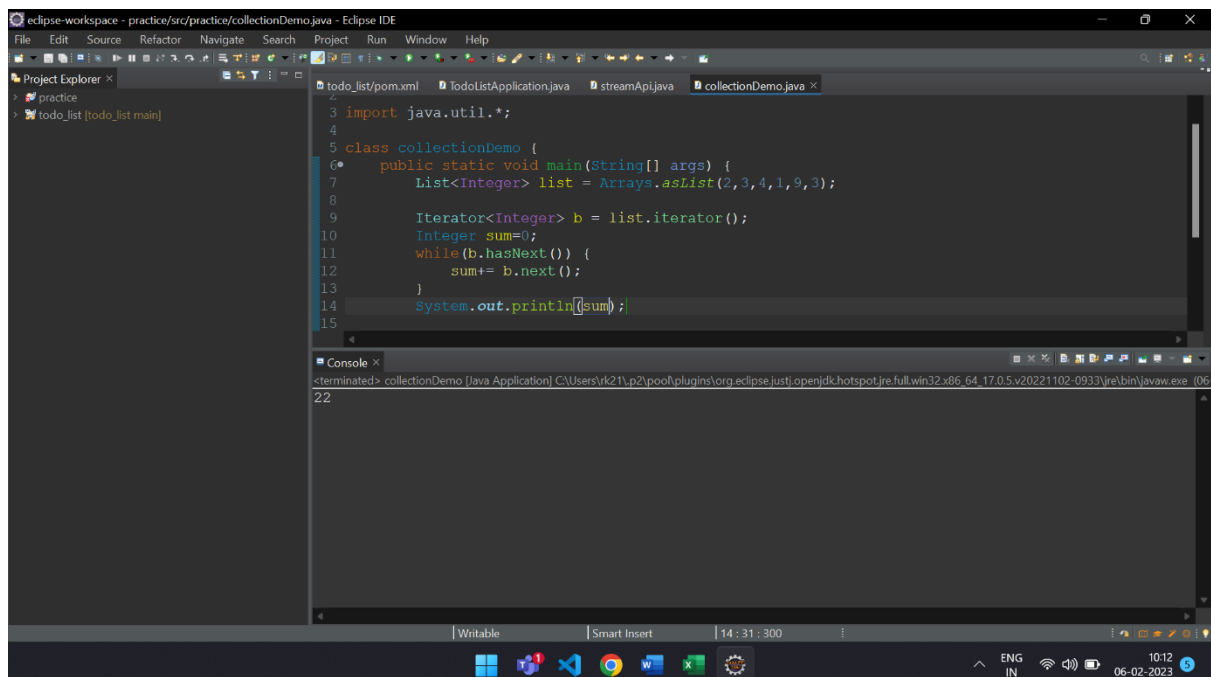
The screenshot shows the Eclipse IDE with a project named 'practice'. The 'Project Explorer' on the left shows the project structure. The main editor displays the file 'collectionDemo.java' with the following code:

```
1  class collectionDemo {
2
3      public static void main(String[] args) {
4          Integer[] a = {10,20,30,10,2,4,30};
5          List<Integer> list = Arrays.asList(a);
6          for(int i=0;i<list.size()-1;i++) {
7              for(int j=i+1;j<list.size();j++) {
8                  if(list.get(i).equals(list.get(j))) {
9                      System.out.println(list.get(i));
10                 }
11             }
12             else {
13                 continue;
14             }
15         }
16     }
17 }
```

The 'Console' at the bottom shows the output of the program:

```
<terminated> collectionDemo [Java Application] C:\Users\vk21\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.5.v20221102-0933\jre\bin\javaw.exe (05-02-2023)
10
30
```

4) How do you get the sum of all elements in an integer array in Java, using Iterator?



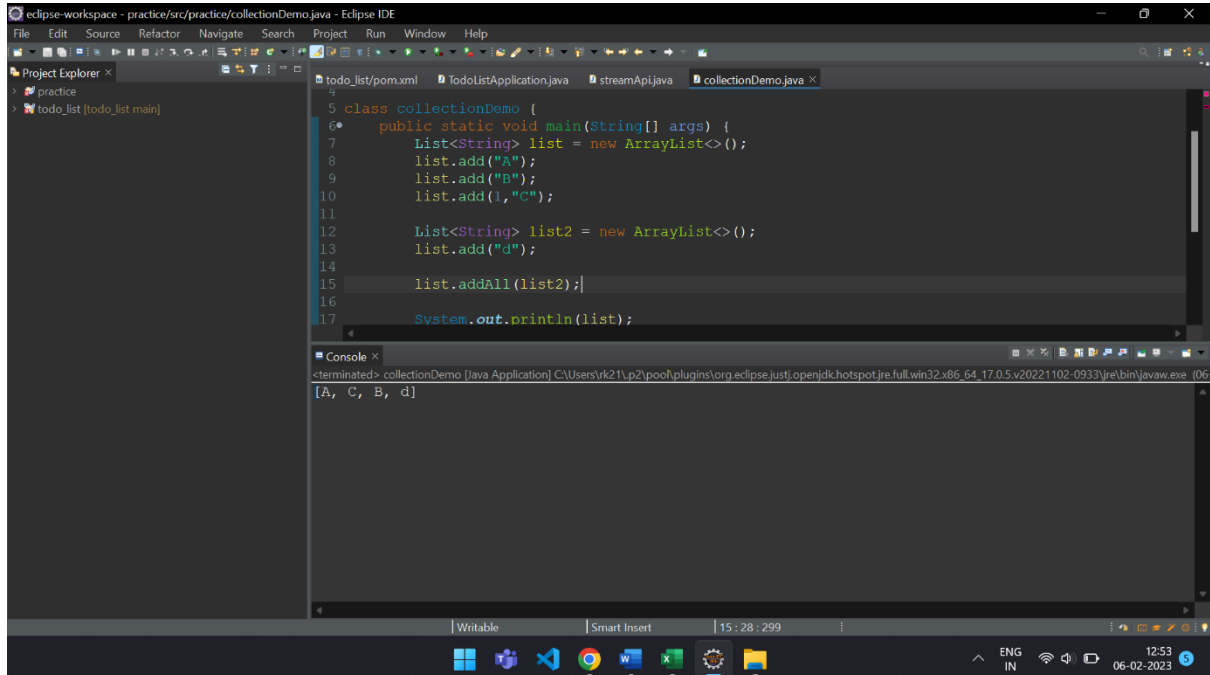
The screenshot shows the Eclipse IDE with a project named 'practice'. The 'Project Explorer' on the left shows the project structure. The main editor displays the file 'collectionDemo.java' with the following code:

```
1  import java.util.*;
2
3  class collectionDemo {
4
5      public static void main(String[] args) {
6          List<Integer> list = Arrays.asList(2,3,4,1,9,3);
7
8          Iterator<Integer> b = list.iterator();
9          Integer sum=0;
10         while(b.hasNext()) {
11             sum+= b.next();
12         }
13         System.out.println(sum);
14     }
15 }
```

The 'Console' at the bottom shows the output of the program:

```
<terminated> collectionDemo [Java Application] C:\Users\vk21\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.5.v20221102-0933\jre\bin\javaw.exe (06-02-2023)
22
```

5) Add elements In-Between the List in Java and add another List into this List



The screenshot shows the Eclipse IDE with a project named 'practice'. The 'Project Explorer' on the left shows the project structure. The main editor displays the file 'collectionDemo.java' with the following code:

```
4
5 class collectionDemo {
6     public static void main(String[] args) {
7         List<String> list = new ArrayList<>();
8         list.add("A");
9         list.add("B");
10        list.add(1, "C");
11
12        List<String> list2 = new ArrayList<>();
13        list.add("d");
14
15        list.addAll(list2);|
16
17        System.out.println(list);
18    }
19 }
```

The 'Console' at the bottom shows the output of the program:

```
<terminated> collectionDemo [Java Application] C:\Users\rk21\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_17.0.5.v20221102-0933\jre\bin\javaw.exe (06
[A, C, B, d]
```

The status bar at the bottom indicates the file is 'Writable', 'Smart Insert' is active, and the cursor is at line 15, column 28. The system tray shows the date and time as 06-02-2023, 12:53.