IRC_SKCT_Java2_COD_Lambda

Test Summary

- No. of Sections: 1
- No. of Questions: 2
- Total Duration: 120 min

Section 1 - Coding

Section Summary

- No. of Questions: 2
- Duration: 120 min

Additional Instructions:

None

Q1. Create a list of the cost that has to be paid for halls and print only the even values using lambda expressions.

Input Format

The first line of the input is the value of n.
Next input is the cost separated by a space.

Output Format

The output prints the even values separated by a space.

Sample Input

```
5
1004 5021 8540 9655 4853
```

Sample Output

```
1004 8540
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Q2. Write a program to concatenate two strings using lambda expressions.
The lambda expression takes two strings as argument and returns the concatenated string.

Input Format

The input consists of two strings.

Output Format

The output prints the concatenated string.

Sample Input

```
Hello
World
```

Sample Output

```
Hello World
```

Time Limit: - ms Memory Limit: - kb Code Size: - kb

Answer Key & Solution

Section 1 - Coding

Q1

Test Case

Input

```
5
1004 5021 8540 9655 4853
```

Output

```
1004 8540
```

Weightage - 20

Input

```
8
1004 5021 8540 9655 4853 52000 4852 1511
```

Output

```
1004 8540 52000 4852
```

Weightage - 20

Input

```
10
1004 5021 8540 9655 4853 52000 4852 1511 48484 84
```

Output

```
1004 8540 52000 4852 48484 84268
```

Weightage - 20

Input

```
12
1004 5021 8540 9655 4853 52000 4852 1511 48484 84
```

Output

```
1004 8540 52000 4852 48484 84268 888
```

Weightage - 20

Input

```
15
1004 5021 8540 9655 4853 52000 4852 1511 48484 84
```

Output

```
1004 8540 52000 4852 48484 84268 888 4862 6842
```

Weightage - 20

Sample Input

```
5
1004 5021 8540 9655 4853
```

Sample Output

```
1004 8540
```

Solution

```
import java.io.*;
import java.util.*;
class Main {
public static void main(String [] args) {
    int n,i;
    Scanner sc = new Scanner(System.in);
```

```
    n = sc.nextInt();
    ArrayList<Integer> e = new ArrayList<Integer>(n);
    for(i=0;i<n;i++) {
        e.add(sc.nextInt());
    }
    e.forEach(ele -> { if(ele%2 == 0) System.out.print(ele+" ");});
}
}
```

Q2

Test Case

Input

```
Take
Action
```

Output

```
Take Action
```

Weightage - 20

Input

```
Move
Away
```

Output

```
Move Away
```

Weightage - 20

Input

```
Do
it
```

Output

```
Do it
```

Weightage - 20

Input

```
Blimey
Careful
```

Output

```
Blimey Careful
```

Weightage - 20

Input

```
Excellent
Good
```

Output

```
Excellent Good
```

Weightage - 20

Sample Input

```
Hello
World
```

Sample Output

```
Hello World
```

Solution

```
import java.io.*;
import java.util.*;
interface StringConcat {

    public String sconcat(String a, String b);
}
class Example {

   public static void main(String args[]) {
        String s1,s2;
        Scanner sc = new Scanner(System.in);
        s1 = sc.nextLine();
        s2 = sc.nextLine();
        StringConcat s = (str1, str2) -> str1 +" "+ str2;
        System.out.println(s.sconcat(s1, s2));
    }
}
```