

Problem

Submissions

Leaderboard

Discussions

Editorial

hackerank.com/challenges/java-currency-formatter/problem?isFullScreen=true

HackerRank

Prepare > Java > Introduction > Java Currency Formatter

Exit Full Screen View

Given a double-precision number, **payment**, denoting an amount of money, use the **NumberFormat** class' **getCurrencyInstance** method to convert **payment** into the US, Indian, Chinese, and French currency formats. Then print the formatted values as follows:

US: formattedPayment
India: formattedPayment
China: formattedPayment
France: formattedPayment

where **formattedPayment** is **payment** formatted according to the appropriate **Locale**'s currency.

Note: India does not have a built-in **Locale**, so you must **construct one** where the language is **en** (i.e., English).

Input Format

A single double-precision number denoting **payment**.

Constraints

- $0 \leq \text{payment} \leq 10^9$

Output Format

On the first line, print US: **u** where **u** is **payment** formatted for US currency.

On the second line, print India: **i** where **i** is **payment** formatted for Indian currency.

On the third line, print China: **c** where **c** is **payment** formatted for Chinese currency.

On the fourth line, print France: **f**, where **f** is **payment** formatted for French currency.

Sample Input

12324.134

Sample Output

US: \$12,324.13
India: Rs.12,324.13
China: ¥12,324.13
France: 12 324,13 €

Change Theme

Language

Java 7

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6 public class Solution {
7
8     public static void main(String[] args) {
9         Scanner scanner = new Scanner(System.in);
10        double payment = scanner.nextDouble();
11        scanner.close();
12        String us = NumberFormat.getCurrencyInstance(Locale.US).format(payment);
13        String india = NumberFormat.getCurrencyInstance(new Locale("en", "in")).format(payment);
14        String china = NumberFormat.getCurrencyInstance(Locale.CHINA).format(payment);
15        String france = NumberFormat.getCurrencyInstance(Locale.FRANCE).format(payment);
16
17        System.out.println("US: " + us);
18        System.out.println("India: " + india);
19        System.out.println("China: " + china);
20        System.out.println("France: " + france);
21    }
22 }
```

Line: 22 Col: 2

Upload Code as File

Test against custom input

Run Code

Submit Code

88°F Sunny

Search

ENG IN

1:52 PM 2/13/2023

Problem

Submissions

Leaderboard

Discussions

Editorial

hackerank.com/challenges/java-currency-formatter/problem?isFullScreen=true

HackerRank

Prepare > Java > Introduction > Java Currency Formatter

Exit Full Screen View

Given a [double-precision](#) number, *payment*, denoting an amount of money, use the [NumberFormat](#) class' [getCurrencyInstance](#) method to convert *payment* into the US, Indian, Chinese, and French currency formats. Then print the formatted values as follows:

```
US: formattedPayment
India: formattedPayment
China: formattedPayment
France: formattedPayment
```

where *formattedPayment* is *payment* formatted according to the appropriate [Locale](#)'s currency.

Note: India does not have a built-in Locale, so you must [construct one](#) where the language is `en` (i.e., English).

Input Format

A single double-precision number denoting *payment*.

Constraints

- $0 \leq \textit{payment} \leq 10^9$

Output Format

On the first line, print US: *u* where *u* is *payment* formatted for US currency.

On the second line, print India: *i* where *i* is *payment* formatted for Indian currency.

On the third line, print China: *c* where *c* is *payment* formatted for Chinese currency.

On the fourth line, print France: *f*, where *f* is *payment* formatted for French currency.

Sample Input

```
12324.134
```

Sample Output

```
US: $12,324.13
India: Rs.12,324.13
China: ¥12,324.13
France: 12 324,13 €
```

Line: 22 Col: 2

Upload Code as File

Test against custom input

Run Code

Submit Code

Congratulations!

You have passed the sample test cases. Click the submit button to run your code against all the test cases.

Sample Test case 0

Input (stdin)

1 12324.134

Your Output (stdout)

1 US: \$12,324.13

2 India: Rs.12,324.13

3 China: ¥12,324.13

4 France: 12 324,13 €

Expected Output

1 US: \$12,324.13

2 India: Rs.12,324.13

88°F Sunny

Search

ENG IN

1:53 PM 2/13/2023