Challenge 1- Change Calculator

Create a function calculate change for a given number (number represents cents). The function should return list, vector or collection of numbers representing the number of quarters (25), dimes (10), nickels (5), and pennies (1) that would yield the correct change.

Solution (Java)
For example, I'm considering the given number as 83cents (number = 83).

To solve this problem-
To get exact change and least number of coins, first I'm considering if I can include quarters, then I'm checking for possible dimes, then nickels and then pennies.

To do that I'm choosing the below steps:

1. Check if the number is greater than 25, if it is, then dividing the number by 25 will give the possible number of quarters that can be present in the change. Now add the count to the result list. Next, we should find the remaining change by just performing- number % 25.

   If the number is less than 25, just add 0 to the result list (quarters count is 0). And, the remaining change is just the same number.

   For above example,
   - 83 > 25;   83 / 25 = 3. Therefore add 3 to result list (quarters count = 3).
   - 83 % 25 = 8. Remaining change = 8

2. Now, to find the number of possible dimes present in the remaining change, check if the number is greater than 10, if it is, then dividing the number by 10 will give the possible number of dimes that can be present in the change. Now add the count to the result list. Next, we should find the remaining change by just performing- number % 10.

   If the number is less than 10, just add 0 to the result list (dimes count is 0). And, the remaining change is just the same number.

   - 8 < 10;   Therefore add 0 to result list (dimes count = 0).
   - Remaining change for next step = 8

3. Repeat the above step by changing '10' to '5' to get possible nickels

   - 8 > 5;   8 / 5 = 1. Therefore add '1' to result list (Nickels count = 1).
   - 8 % 5 = 3. Remaining change for next step = 3

4. Repeat step 2 by changing ‘10’ to ‘1’ to get possible pennies

    - 3 > 1;   3 / 1 = 3. Therefore add ‘3’ to result list (Pennies count = 3).
    - At this point no change will be left

5. Hence for 83cents, we get **3-Quarters, 0-Dimes, 1-Nickels, 3- Pennies**.

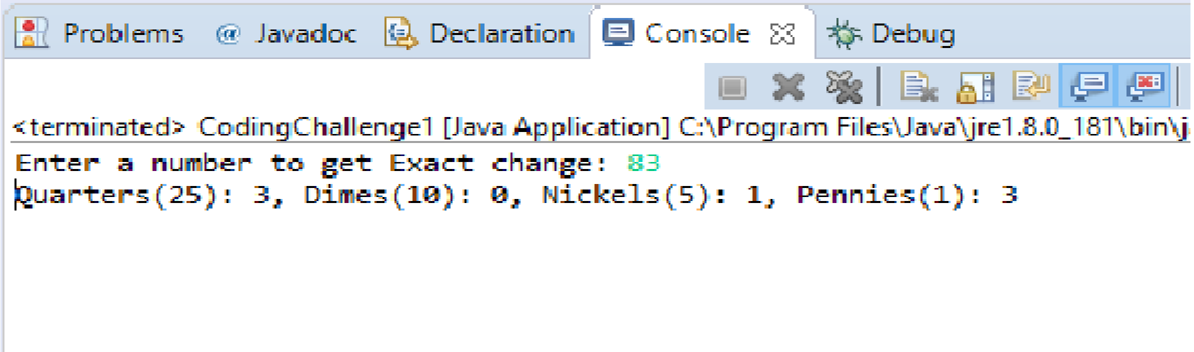
Code

ChangeCalulator function:

```
public static List<Integer> changeCalculator(int givenNum){

            int temp;
            List<Integer> coinsCount = new ArrayList<Integer>();

            if(givenNum >= 25) {
                  temp = givenNum / 25;
                  givenNum = givenNum % 25;
                  coinsCount.add (temp);
            } else { coinsCount.add(0); }

            if(givenNum >= 10) {
                  temp = givenNum / 10;
                  givenNum = givenNum % 10;
                  coinsCount.add(temp);
            } else { coinsCount.add(0); }


            if(givenNum >= 5) {
                  temp = givenNum / 5;
                  givenNum = givenNum % 5;
                  coinsCount.add(temp);
            } else { coinsCount.add(0); }


            if(givenNum >= 1) {
                  temp = givenNum / 1;
                  givenNum = givenNum % 1;
                  coinsCount.add(temp);
            } else { coinsCount.add(0); }


            return coinsCount;
      }
```

To download complete code, please find **CodingChallenge1.java** class in the same repository.

Sample Output



```
<terminated> CodingChallenge1 [Java Application] C:\Program Files\Java\jre1.8.0_181\bin\j
Enter a number to get Exact change: 83
Quarters(25): 3, Dimes(10): 0, Nickels(5): 1, Pennies(1): 3
```

Test Cases & Instructions to execute the code

In the console as shown in the “Sample Output” above, please enter the number to such we should get the change and hit “Enter”, it will show the exact change!

Enter a number to get Exact change: 83
Quarters(25) : 3, Dimes(10) : 0, Nickels(5) : 1, Pennies(1) : 3

Enter a number to get Exact change: 96
Quarters(25) : 3, Dimes(10) : 2, Nickels(5) : 0, Pennies(1) : 1

Enter a number to get Exact change: 0
Quarters(25) : 0, Dimes(10) : 0, Nickels(5) : 0, Pennies(1) : 0

Even if user enters negative number, it will give 0-Quarters, 0-Dimes, 0-Nickels, 0- Pennies
Enter a number to get Exact change: -1
Quarters(25) : 0, Dimes(10) : 0, Nickels(5) : 0, Pennies(1) : 0