Java Chico's Change Counter

Time required: 90 minutes

Please read the directions carefully before beginning the assignment.

- Comment your code as shown in the tutorials and other code examples.
- Follow all directions carefully and accurately.
- Think of the directions as minimum requirements.

Pseudocode or TODO

1. Write pseudocode or TODO for the exercise.
2. Comment your code to show evidence of understanding.

Here's Why I Want You to Do It

Demonstrate understanding of:

Variables, Constants, Calculations, Decisions

Here's What I Want You to Do

You work at Bank Asgard as a software engineer. Many of the new employees do not know how to count change in dollars. Their last jobs were at banks that only used pesos. You have been tasked with creating a program to help them until they acquire the skill.

Create a Java application named **ChangeCounter.java**

Write a class that calculates and displays the conversion of an entered number of dollars into currency denominations—20's, 10's, 5's, and 1's.

- Use named constants
- Use the if decision statement

TODO Outline

You can copy and paste this TODO outline to get started.

```
/*
 * Filename: ChangeCounter.java
 * Written by:
 * Written on:
 * Convert dollars into change
 */
import java.util.Scanner;
class ChangeCounter {
    public static void main(String[] args) {
        // TODO: Declare currency value constants needed for counting change

        // TODO: Declare variables needed for conversion

        // TODO: Print creative title

        // TODO: Get dollars input from user

        // TODO: Calculate change in 20's, 10's, 5's, and 1's

        // TODO: Display result

        // Close scanner input object
        input.close();
   }
}
```

Example run:

```
----------------------------------------
|     Chico's Change Counter           |
----------------------------------------
Enter dollars: 116
$116 is:
5-$20's
1-$10's
1-$5's
1-$1's
```

Assignment Submission

1. Use pseudocode or TODO.

2. Comment your code to show understanding.
3. Attach the program files.
4. Attach screenshots showing the successful operation of the program.
5. Submit in Blackboard.