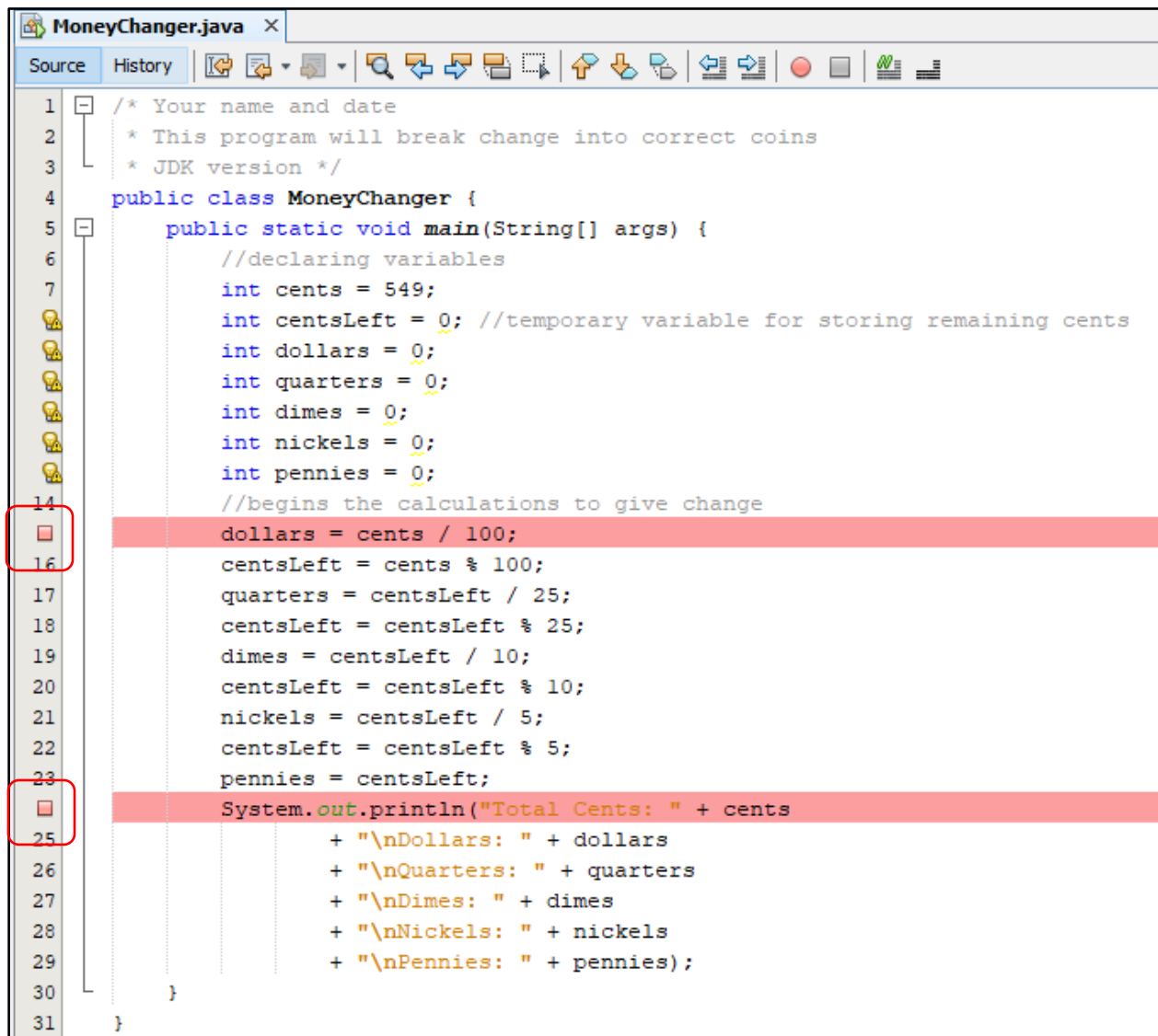


NetBeans Debugger

The NetBeans Debugger allows you to place breakpoints in your source code and step through your code looking for errors. We are able to watch how variables change throughout our program and evaluate expressions. There are many more advanced features of the debugger that we are not going to cover. This section will just give you a brief introduction of debugging.

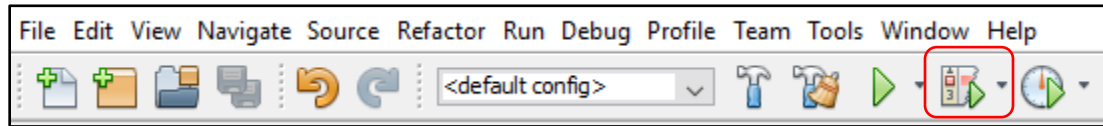
To start debugging, you will need to open the file that contains the code that you want to debug. You will need to create a breakpoint at each line that you want to pause to debug. To create a breakpoint, click on the line code that needs debugging and choose **Debug > Toggle Line Breakpoint** or click in the sidebar to the left of your code where the line number is located. Let's use the MoneyChanger exercise as an example, since it has a few variables that we can monitor. The breakpoints were set before our equations and after so that we can watch the variable values change.



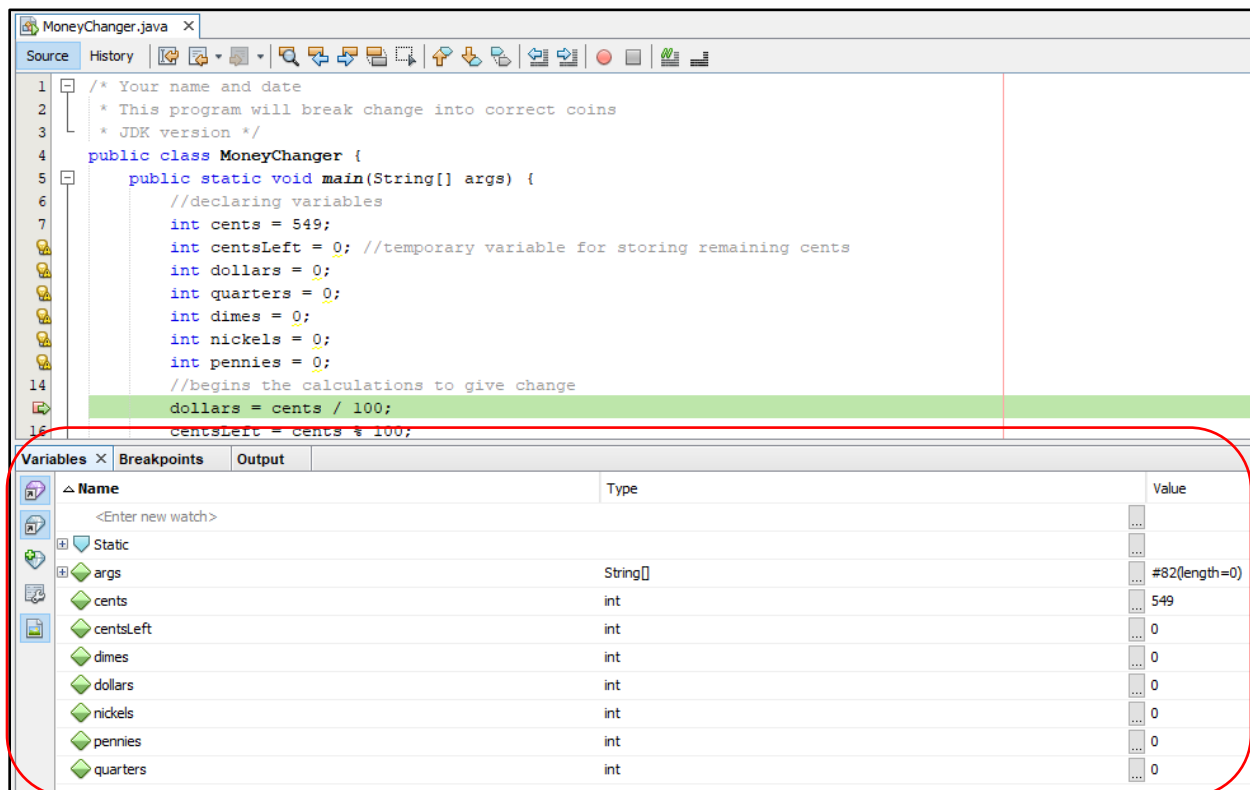
```
1  /* Your name and date
2   * This program will break change into correct coins
3   * JDK version */
4  public class MoneyChanger {
5      public static void main(String[] args) {
6          //declaring variables
7          int cents = 549;
8          int centsLeft = 0; //temporary variable for storing remaining cents
9          int dollars = 0;
10         int quarters = 0;
11         int dimes = 0;
12         int nickels = 0;
13         int pennies = 0;
14         //begins the calculations to give change
15         dollars = cents / 100;
16         centsLeft = cents % 100;
17         quarters = centsLeft / 25;
18         centsLeft = centsLeft % 25;
19         dimes = centsLeft / 10;
20         centsLeft = centsLeft % 10;
21         nickels = centsLeft / 5;
22         centsLeft = centsLeft % 5;
23         pennies = centsLeft;
24         System.out.println("Total Cents: " + cents
25                             + "\nDollars: " + dollars
26                             + "\nQuarters: " + quarters
27                             + "\nDimes: " + dimes
28                             + "\nNickels: " + nickels
29                             + "\nPennies: " + pennies);
30     }
31 }
```

The screenshot shows the NetBeans IDE with the file `MoneyChanger.java` open. The code is a Java program that calculates the number of coins (dollars, quarters, dimes, nickels, and pennies) for a given amount in cents (549). Two breakpoints are set: one at line 15 (before the calculation of dollars) and another at line 23 (after the calculation of pennies). The breakpoints are indicated by red squares in the left margin next to the line numbers.

After setting the breakpoints, we will click on the **Debug Project** button.



The debugger should display a tab with your variables as shown below. The variable type and value will be indicated. You should see that the cents variable is initialized to 549 and the other variables are set to 0.




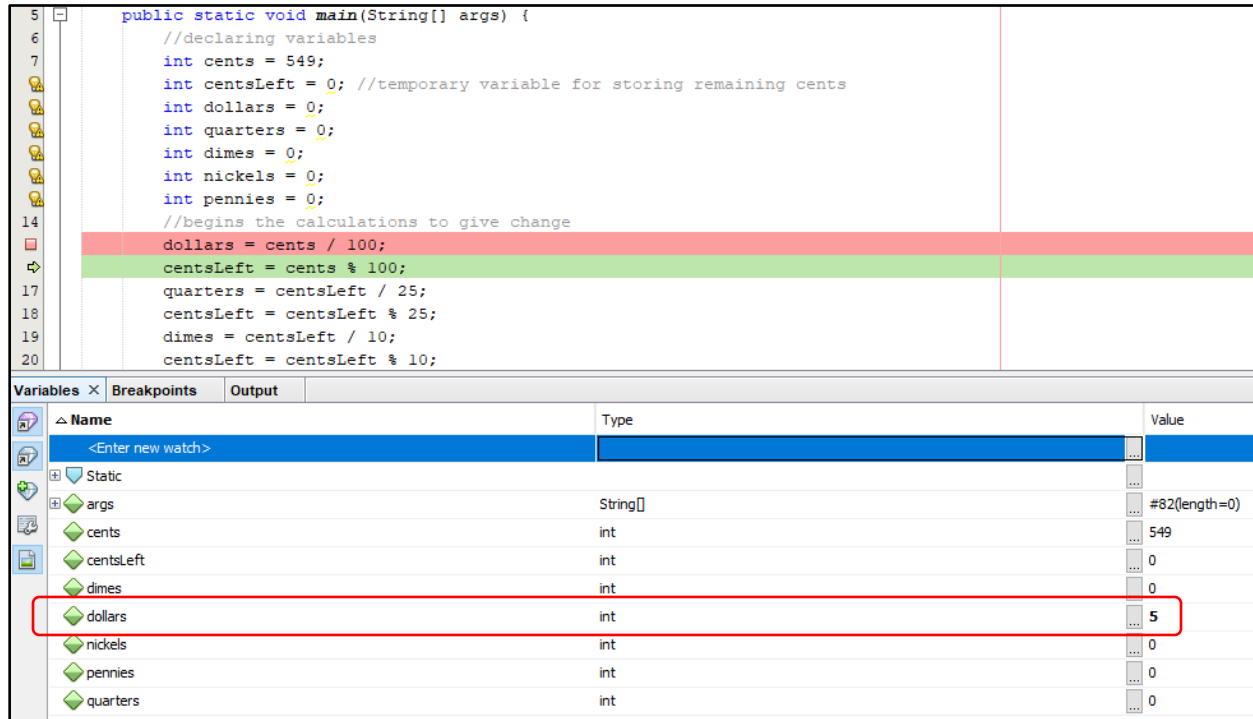
You should notice a new section of the toolbar for debugging.



The basic choices are to finish debugging the program, pause the debugging, continue to the next breakpoint, or to step over your program.



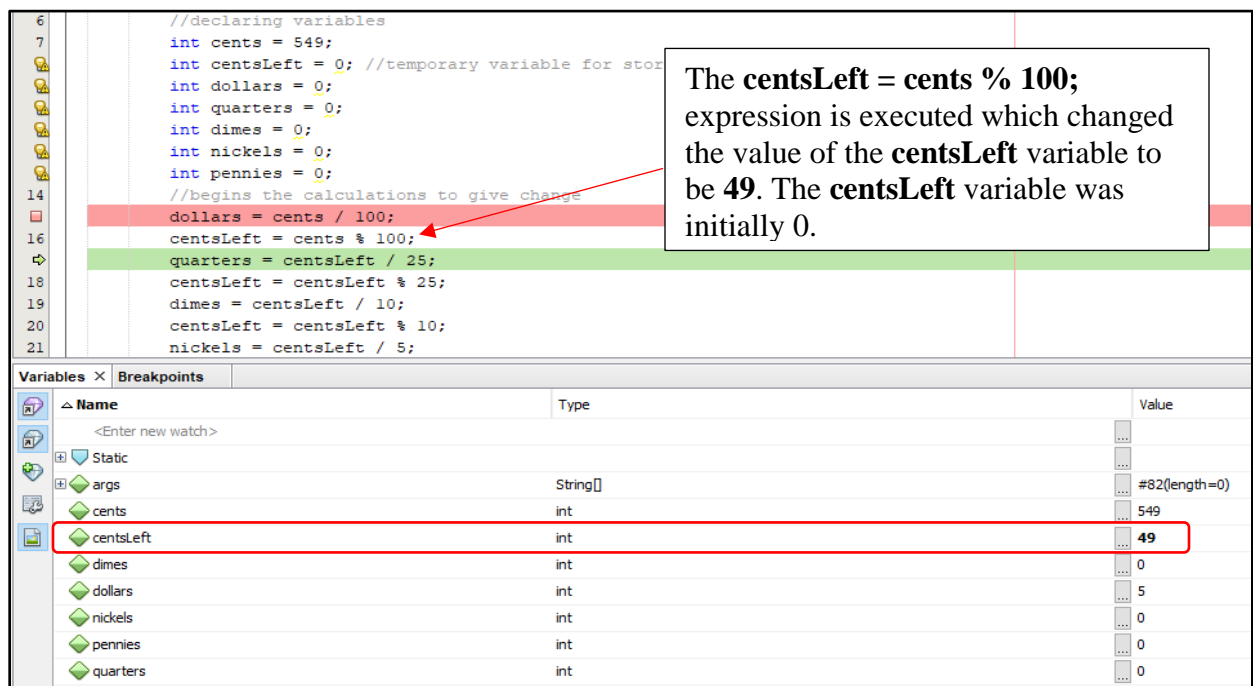
Let's use the step over button  so that we can see how each expression changes the values of our variables. This causes the **dollars = cents / 100;** expression to be executed which changed the value of the **dollars** variable to be **5**. The **dollars** variable was initially 0.



```
5 public static void main(String[] args) {
6     //declaring variables
7     int cents = 549;
8     int centsLeft = 0; //temporary variable for storing remaining cents
9     int dollars = 0;
10    int quarters = 0;
11    int dimes = 0;
12    int nickels = 0;
13    int pennies = 0;
14    //begins the calculations to give change
15    dollars = cents / 100;
16    centsLeft = cents % 100;
17    quarters = centsLeft / 25;
18    centsLeft = centsLeft % 25;
19    dimes = centsLeft / 10;
20    centsLeft = centsLeft % 10;
```

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	0
dimes	int	0
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	0


Let's use the step over button  again.



```
6 //declaring variables
7 int cents = 549;
8 int centsLeft = 0; //temporary variable for storing remaining cents
9 int dollars = 0;
10 int quarters = 0;
11 int dimes = 0;
12 int nickels = 0;
13 int pennies = 0;
14 //begins the calculations to give change
15 dollars = cents / 100;
16 centsLeft = cents % 100;
17 quarters = centsLeft / 25;
18 centsLeft = centsLeft % 25;
19 dimes = centsLeft / 10;
20 centsLeft = centsLeft % 10;
21 nickels = centsLeft / 5;
```

The **centsLeft = cents % 100;** expression is executed which changed the value of the **centsLeft** variable to be **49**. The **centsLeft** variable was initially 0.

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	49
dimes	int	0
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	0

Let's use the step over button  again.

```
14 //begins the calculations to give change
15 dollars = cents / 100;
16 centsLeft = cents % 100;
17 quarters = centsLeft / 25;
18 centsLeft = centsLeft % 25;
19 dimes = centsLeft / 10;
20 centsLeft = centsLeft % 10;
21 nickels = centsLeft / 5;
22 centsLeft = centsLeft % 5;
```

The **quarters = centsLeft / 25;** expression is executed which changed the value of the **quarters** variable to be **1**. The **quarters** variable was initially 0.


Variables		Breakpoints
Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	49
dimes	int	0
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	1

Let's use the step over button  again.

```
14 //begins the calculations to give change
15 dollars = cents / 100;
16 centsLeft = cents % 100;
17 quarters = centsLeft / 25;
18 centsLeft = centsLeft % 25;
19 dimes = centsLeft / 10;
20 centsLeft = centsLeft % 10;
21 nickels = centsLeft / 5;
22 centsLeft = centsLeft % 5;
23 pennies = centsLeft;
24 System.out.println("Total Cents: " + cents
25 + "\nDollars: " + dollars
26 + "\nQuarters: " + quarters
27 + "\nDimes: " + dimes
28 + "\nNickels: " + nickels
29 + "\nPennies: " + pennies);
```

The **centsLeft = centsLeft % 25;** expression is executed which changed the value of the **centsLeft** variable to be **24**. The **centsLeft** was 49 before.

Variables		Breakpoints
Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	24
dimes	int	0
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	1

Let's use the step over button  again.

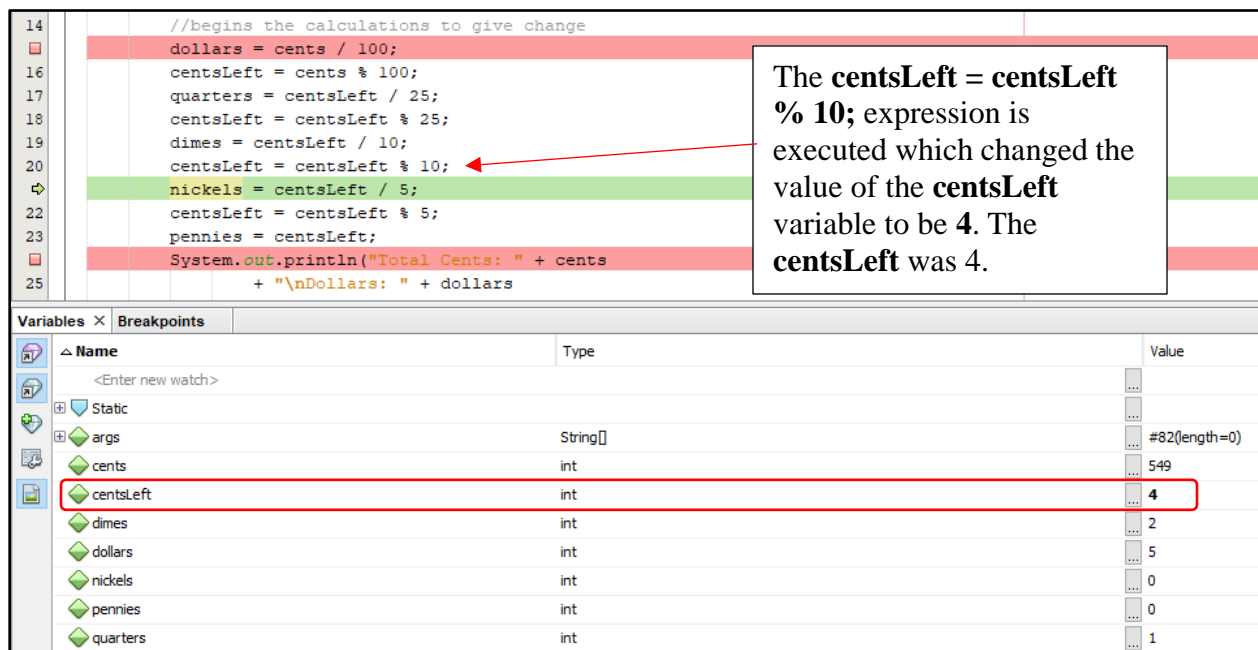


```
14 //begins the calculations to give change
15 dollars = cents / 100;
16 centsLeft = cents % 100;
17 quarters = centsLeft / 25;
18 centsLeft = centsLeft % 25;
19 dimes = centsLeft / 10;
20 centsLeft = centsLeft % 10;
21 nickels = centsLeft / 5;
22 centsLeft = centsLeft % 5;
23 pennies = centsLeft;
24 System.out.println("Total Cents: " + cents
25                      + "\nDollars: " + dollars
26                      + "\nQuarters: " + quarters
27                      + "\nDimes: " + dimes
28                      + "\nNickels: " + nickels
```

The **dimes = centsLeft / 10;** expression is executed which changed the value of the **dimes** variable to be 2. The **dimes** variable was 0 initially.

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	24
dimes	int	2
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	1


Let's use the step over button  again.

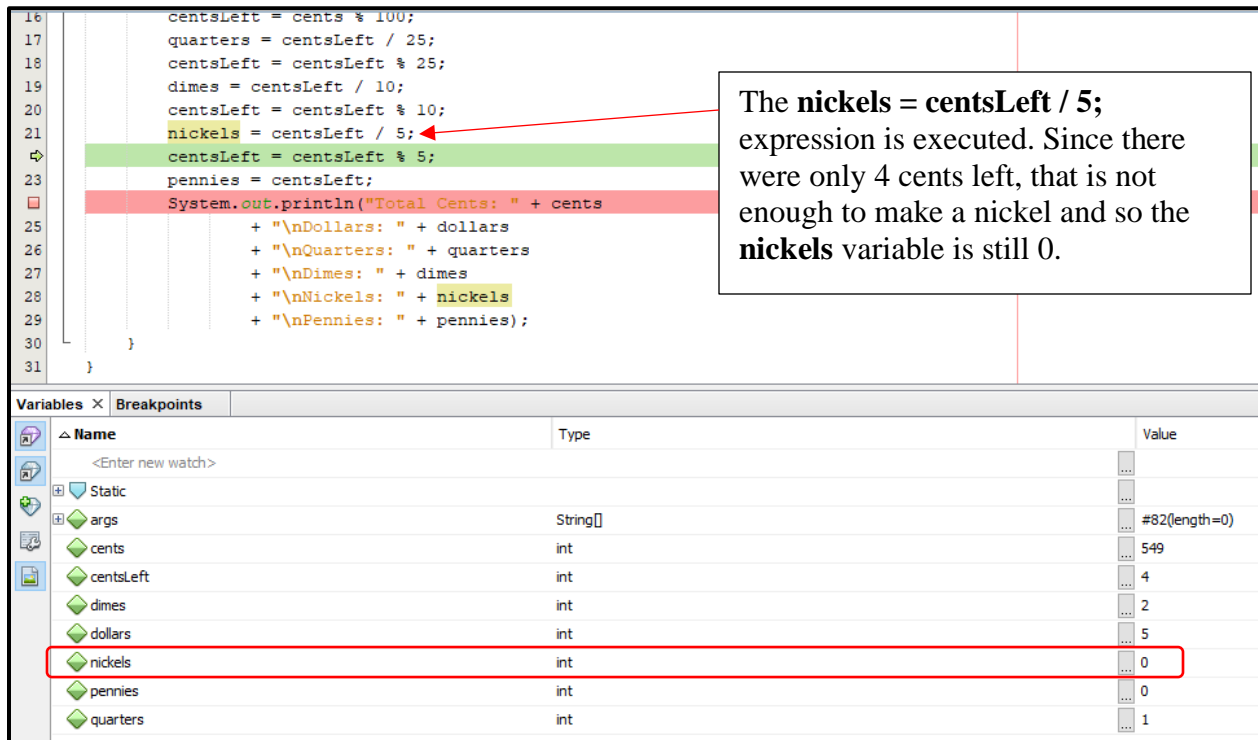


```
14 //begins the calculations to give change
15 dollars = cents / 100;
16 centsLeft = cents % 100;
17 quarters = centsLeft / 25;
18 centsLeft = centsLeft % 25;
19 dimes = centsLeft / 10;
20 centsLeft = centsLeft % 10;
21 nickels = centsLeft / 5;
22 centsLeft = centsLeft % 5;
23 pennies = centsLeft;
24 System.out.println("Total Cents: " + cents
25                      + "\nDollars: " + dollars
```

The **centsLeft = centsLeft % 10;** expression is executed which changed the value of the **centsLeft** variable to be 4. The **centsLeft** was 4.

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	4
dimes	int	2
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	1

Let's use the step over button  again.

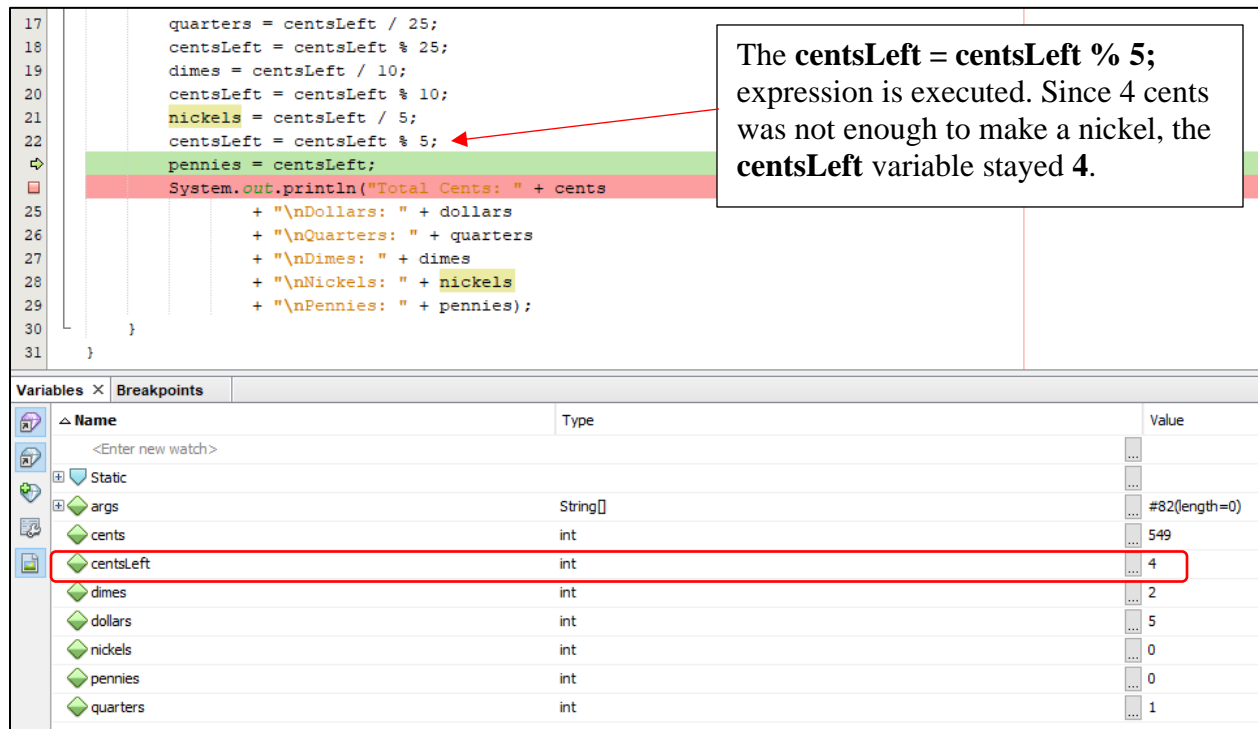


```
16      centsLeft = cents % 100;
17      quarters = centsLeft / 25;
18      centsLeft = centsLeft % 25;
19      dimes = centsLeft / 10;
20      centsLeft = centsLeft % 10;
21      nickels = centsLeft / 5;
22      centsLeft = centsLeft % 5;
23      pennies = centsLeft;
24      System.out.println("Total Cents: " + cents
25                          + "\nDollars: " + dollars
26                          + "\nQuarters: " + quarters
27                          + "\nDimes: " + dimes
28                          + "\nNickels: " + nickels
29                          + "\nPennies: " + pennies);
30  }
31 }
```

The **nickels = centsLeft / 5;** expression is executed. Since there were only 4 cents left, that is not enough to make a nickel and so the **nickels** variable is still 0.

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	4
dimes	int	2
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	1


Let's use the step over button  again.

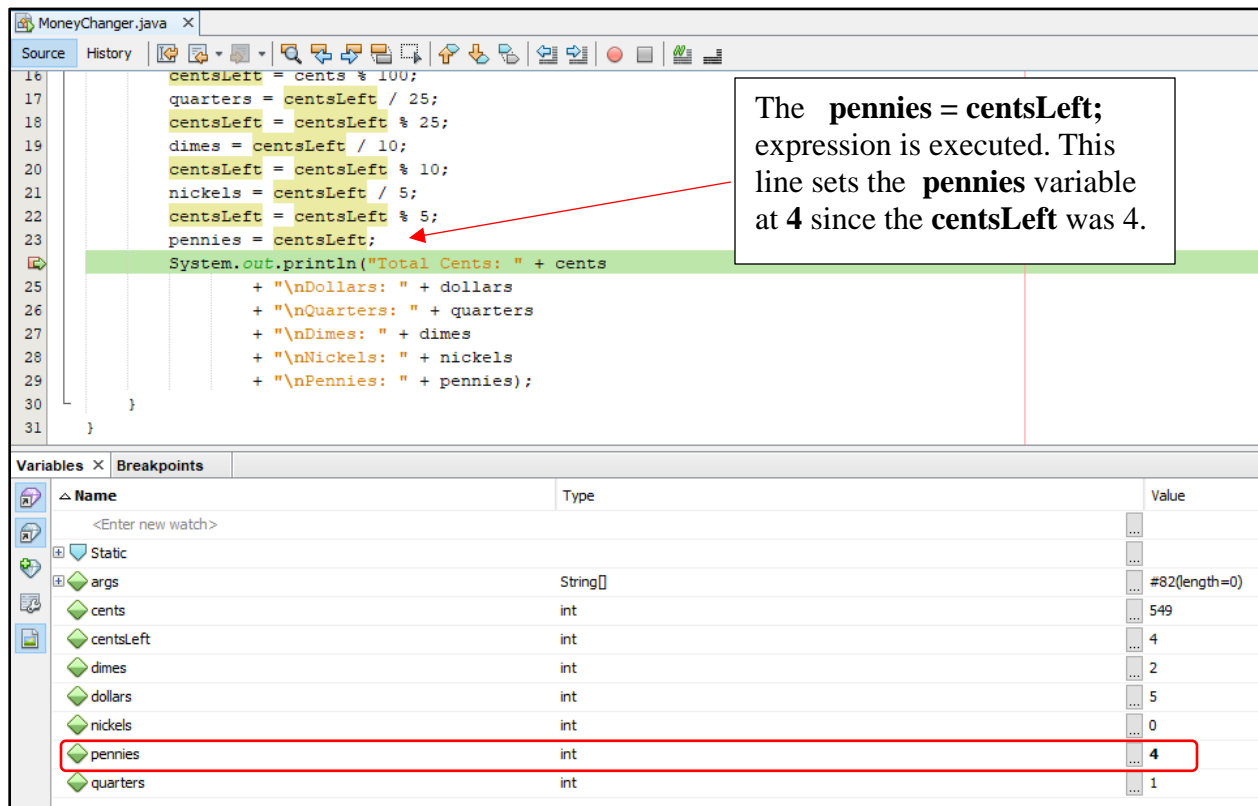


```
17      quarters = centsLeft / 25;
18      centsLeft = centsLeft % 25;
19      dimes = centsLeft / 10;
20      centsLeft = centsLeft % 10;
21      nickels = centsLeft / 5;
22      centsLeft = centsLeft % 5;
23      pennies = centsLeft;
24      System.out.println("Total Cents: " + cents
25                          + "\nDollars: " + dollars
26                          + "\nQuarters: " + quarters
27                          + "\nDimes: " + dimes
28                          + "\nNickels: " + nickels
29                          + "\nPennies: " + pennies);
30  }
31 }
```

The **centsLeft = centsLeft % 5;** expression is executed. Since 4 cents was not enough to make a nickel, the **centsLeft** variable stayed 4.


Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	4
dimes	int	2
dollars	int	5
nickels	int	0
pennies	int	0
quarters	int	1

Let's use the step over button  again.




The **pennies = centsLeft;** expression is executed. This line sets the **pennies** variable at **4** since the **centsLeft** was 4.

Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	4
dimes	int	2
dollars	int	5
nickels	int	0
pennies	int	4
quarters	int	1

If you would have used the continue button  from the first breakpoint instead of the step over button, then your program would have executed all of the expressions and the variables would be the final results.

Variables × Breakpoints		
Name	Type	Value
<Enter new watch>		
Static		
args	String[]	#82(length=0)
cents	int	549
centsLeft	int	4
dimes	int	2
dollars	int	5
nickels	int	0
pennies	int	4
quarters	int	1

To end the debugging, you will click on the finish debugging button .