

# CITP 190 – Introduction to Java

## Project 3

This project requires you to create two projects. One project will calculate interest. The other project will calculate coins for change.

To receive full credit for this project you must submit the following:

- A detailed design diagram for each program.
- The source code for each program (the .java file) following the coding standards.
- The bytecode for each program (the.class file)
- Proof of the correctness of your output using the test data provided. Please note that you must provide proof for all test data. You may provide additional test data.
- A capture of the output of each program. The output must show all data from your proof.

Submit all files as one (1) ZIP file to the Project 3 Drop Box in the course site.

### Grading:

Program design (5 points for each program)	10 points
Following coding standards	5 points
Output is presented as shown (including spacing and spelling)	2 points
Code	3 points

### *Important notes:*

1. Incorrect calculations will result in a 0 grade for this project.
2. Uploading more than one zip file will result in a 0 grade for this project.
3. Not using all the test data provided will result in a 0 grade for this project.

## Project 3-1: Calculate interest

### Console

```
Welcome to the Interest Calculator
```

```
Enter loan amount:  520000
Enter interest rate: .05375
```

```
Loan amount:      $520,000.00
Interest rate:     5.375%
Interest:          $27,950.00
```

```
Continue? (y/n) : y
```

```
Enter loan amount:  4944.5
Enter interest rate: .01
```

```
Loan amount:      $4,944.50
Interest rate:     1%
Interest:          $49.45
```

```
Continue? (y/n) : n
```

### Operation

- The application prompts the user to enter a loan amount and an interest rate. The interest rate should be entered as a decimal number, not the percentage. For example a 5.5% interest rate would be entered as .055.
- The application calculates the interest amount (the loan amount multiplied by the interest rate) and formats the loan amount, interest rate, and interest amount. Then, it displays the formatted results to the user.
- The application prompts the user to continue.

### Specifications

- This application should use the BigDecimal class to make sure that all calculations are accurate. It should round the interest that's calculated to two decimal places, rounding up if the third decimal place is five or greater.
- The value for the formatted interest rate should allow for up to 3 decimal places.
- Assume that the user will enter valid double values for the loan amount and interest rate.
- The application should continue only if the user enters "y" or "Y" to continue.
- Use spaces to pad the prompts so the input is left aligned as shown above.

### Test Data

Loan Amount	Interest Rate
\$10,000	6.2%
\$15,875	3.475%
\$500	12.5%

## Project 3-2: Calculate coins for change

### Console

```
Welcome to the Change Calculator
```

```
Enter number of cents (0-99): 99
```

```
Quarters: 3
```

```
Dimes: 2
```

```
Nickels: 0
```

```
Pennies: 4
```

```
Continue? (y/n): y
```

```
Enter number of cents (0-99): 55
```

```
Quarters: 2
```

```
Dimes: 0
```

```
Nickels: 1
```

```
Pennies: 0
```

```
Continue? (y/n): n
```

### Operation

- The application prompts the user to enter a number of cents from 0 to 99.
- The application displays the minimum number of quarters, dimes, nickels, and pennies that represent the coins that make up the specified number of cents.
- The application prompts the user to continue.

### Specifications

- Assume that the user will enter a valid integer value for the number of cents.
- The application should continue only if the user enters “y” or “Y” to continue.
- Use spaces to pad the prompts so the input is left aligned as shown above.

### Test Data

Use the following amounts:

1	50	14	24
3	51	15	25
4	75	16	26
5	79	19	49
6	83	20	
9	99	21	
10	11	22	

Enter Y for continue after each amount except the last amount. Note that you are testing inside, outside, and at boundaries!