Test_Ch3

Due Sep 19, 2018 at 10am   **Points** 30   **Questions** 3

Available Sep 19, 2018 at 8am - Sep 19, 2018 at 10am about 2 hours

Time Limit None

Instructions

Implement the java programs as per given descriptions. Please note the following :

- Write your name in every file that you are submitting for grading.
- You are allowed to:
  - use your textbook
  - look at your assignments on CI Learn
  - look up "Examples by Chapter" on CI Learn
  - look up lecture notes on CI Learn
  - look up java class documentation at: **http://docs.oracle.com/javase/8/docs/api/ (http://docs.oracle.com/javase/8/docs/api/)**
  - other internet searches are **NOT ALLOWED**
  - no other sources can be used
- You must work on the classroom computer. Please note that the instructor will be able to see all students desktops in real time on the instructor's computer through the classroom management tool called "Net Control 2".
- No restroom breaks are allowed.
- When finished, zip the files and submit on CI Learn for grading.

Before you start:

- open **Java API** **(http://docs.oracle.com/javase/8/docs/api/)**
- work out provided examples by hand
- think of how many variables do you need? What should be their data types? Do you need any constants? What should be their names?
- write pseudocode for computing the answers
- declare the variables and constants that you need
- turn pseudocode into java statements
- pay attention to user friendly input and output:
  - before starting computation, we prompt the user for the bill value and item price
  - when the computation is finished, we display the results that are informative and properly formatted

This quiz was locked Sep 19, 2018 at 10am.

Attempt History

	Attempt	Time	Score
LATEST	**Attempt 1**	119 minutes	31.5 out of 30

⚠ Correct answers are hidden.

Score for this quiz: **31.5** out of 30
Submitted Sep 19, 2018 at 10am
This attempt took 119 minutes.

Question 1   **10 / 10 pts**

Write a program that reads a telephone number which is in the format:
areaCode-number-number

For example the following would be valid inputs:

- 805-437-3236
- 12-34-56789

You should output the phone number without the area code and with a space instead of the dash, all enclosed in double quotes. See sample runs:

Run #1:

```
Please enter a phone number (e.i. 12-34-56789) :    12-34-56789
```

```
The phone number 12-34-56789 without area code is: "34 56789"
```

Run #2:

```
Please enter a phone number (e.i. 12-34-56789) :    805-437-3236
```

```
The phone number 805-437-3236 without area code is: "437 3236"
```

Please remember to use <u>appropriate</u> methods like `indexOf`, `lastIndexOf`, `substring` from the <u>String</u> <u>(http://download.oracle.com/javase/8/docs/api/)</u> class to make the program generic.

Make sure that the output is user friendly

   ⤓ **PhoneNumber.java (https://cilearn.csuci.edu/files/596699/download)**

Question 2   9 / 10 pts

Write a program that reads a character from the keyboard and detects if the given character is a letter.

Go to **Java API** (http://docs.oracle.com/javase/8/docs/api/) , look at the `Character` class API , and find a method that can detect if a given character is a letter. Please note that the `Character` class, similarly to the `Math` class, is static.

The following are sample runs:

<u>Run #1:</u>

```
Please enter a single character.

A

The character 'A' is a letter: true
```

<u>Run #2:</u>

```
Please enter a single character.

2

The character '2' is a letter: false
```

   ⤓ **LetterDetection.java.zip (https://cilearn.csuci.edu/files/596771/download)**

Question 3   8 / 10 pts

Write a program that simulates a vending machine. A customer selects an item for purchase and inserts a bill into the vending machine. The vending machine dispenses the purchased item and gives a change in coins. The following coin denominations are supported: **half dollar, quarter, dime, nickel, penny.**

We will assume that the machine accepts **only bills** and gives change back in the **least amount** of coins. We will also assume that the inserted bill **covers the item price**.

Your task is to compute how many coins of each type to return.

Use `Scanner` for the input. The **item price** must be received as `double`; the **bill value** must be received as `int`.

Your output must be formatted with `DecimalFormat`.

The following are sample runs:

Run #1:

```
Enter item price as double (i.e. 1.25): 1.25
```

```

```

```
Enter bill value as int (i.e. 1 for $1 bill, 5 for $5 bill, etc.): 5
```

```
For the item price of $1.25 the change due out of $5.00 is $3.75 :
```

```

```

```
7 half dollar coins; 1 quarter coins; 0 dime coins; 0 nickel coins; 0
 penny coins;
```

Run #2:

```
Enter item price as double (i.e. 1.25): 12.34
```

```

```

Enter bill value as int (i.e. 1 for $1 bill, 5 for $5 bill, etc.): **20**

For the item price of $12.34 the change due out of $20.00 is $7.66 :

15 half dollar coins; 0 quarter coins; 1 dime coins; 1 nickel coins; 1 penny coins;

Run #3:

Enter item price as double (i.e. 1.25): **10**

Enter bill value as int (i.e. 1 for $1 bill, 5 for $5 bill, etc.): **10**

For the item price of $10.00 the change due out of $10.00 is $0.00 :

0 half dollar coins; 0 quarter coins; 0 dime coins; 0 nickel coins; 0 penny coins;

⤓ **VendingMachine.java.zip (https://cilearn.csuci.edu/files/596821/download)**

Quiz Score: **31.5** out of 30

This quiz score has been manually adjusted by +4.5 points.