Test_Ch5

Due Oct 3, 2018 at 10am **Points** 30 **Questions** 2

Available Oct 3, 2018 at 8am - Oct 3, 2018 at 10am about 2 hours **Time Limit** None

Allowed Attempts Unlimited

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.

For each of the exercises:

- before starting implementation write the algorithm in pseudocode in the file as a comment, and after that proceed with the coding.
- use Scanner for input

The test is due by the end of the class.

This quiz was locked Oct 3, 2018 at 10am.

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	111 minutes	26.5 out of 30

ⓘ Correct answers are hidden.

Score for this attempt: **26.5** out of 30
Submitted Oct 3, 2018 at 9:51am
This attempt took 111 minutes.

Question 1

13 / 15 pts

Write a program that asks the user to enter her/his csuci email. To check if the given input represents a valid email define a valid email pattern and use it with the **`matches`** method.

Only **one pattern** should be defined.

The entered email is a **valid csuci email** if it:

1. starts with at least one letter in lower case
2. followed by a dot
3. followed by at least one letter in lower case
4. followed by none or exactly three digits
5. followed by `@`
6. followed by none, or at least one letter in lower case and a dot
7. followed by `csuci.edu`

If the input is not a valid csuci email, your program should output that the email is not valid, otherwise the program should output that the email is valid.

If the email is valid, its domain (sequence of characters between the @ and the last dot) should be extracted and used as a `switch` variable to compute the result: student, staff, or unknown. The result must be printed **outside** of the `switch` block.

For example:

- `jane.flower@csuci.edu` - would be a **valid staff** email

- `jane.flower123@myci.csuci.edu` - would be a **valid student** email

- `jane.flower123@mydomain.csuci.edu` - would be a **valid** email from **unknown** domain

- `jane.j.flower@csuci.edu` - would be an **invalid** email

- `jane.flower@csuci` - would be an **invalid** email, and so on

The following shows sample runs of the program:

Run #1

```
Please enter your school email
anna.bieszczad@csuci.edu
The entered email "anna.bieszczad@csuci.edu" is valid
Domain "csuci" indicates staff email
```

Run #2

```
Please enter your school email
jane.flower123@myci.csuci.edu
The entered email "jane.flower123@myci.csuci.edu" is
valid
Domain "myci.csuci" indicates student email
```

Run #3

```
Please enter your school email
jane.flower@mydomain.csuci.edu
The entered email "jane.flower@mydomain.csuci.edu" is
valid
Domain "mydomain.csuci" indicates unknown email
```

Run #4

```
Please enter your school email
anna.t.bieszczad@csuci.edu
The entered input "anna.t.bieszczad@csuci.edu" is NOT
a valid email
```

Run #5

```
Please enter your school email
jane.flower@mydomain.edu
The entered input "jane.flower@mydomain.edu" is NOT a
valid email
```

Run #6

```
Please enter your school email
annabieszczad@csuci.edu
The entered input "annabieszczad@csuci.edu" is NOT a
valid email
```

Run #7

```
Please enter your school email
anna.bieszczad@csuci.gov
```

```
The entered input "anna.bieszczad@csuci.gov" is NOT a
valid email
```

Run #8

```
Please enter your school email
anna.bieszczad.csuci.edu
The entered input "anna.bieszczad.csuci.edu" is NOT a
valid email
```

[Email.java.zip (https://cilearn.csuci.edu/files/623042/download)](https://cilearn.csuci.edu/files/623042/download)

Question 2

11.5 / 15 pts

Write a program that calculates and prints the bill for a cellular phone company. The company offers two *types of service*: **regular** or **premium**. The rates vary depending on the type of service and are computed as follows:

- **Regular service**: $10.00 plus first 50 minutes free. Charges for over 50 minutes are $0.20 per minute.
- **Premium service**: $25.00 plus:

  o For calls made during the day, the first 75 minutes are free; charges for over 75 minutes are $0.10 per minute

  o For calls made during the night, the first 100 minutes are free; charges for over 100 minutes are $0.05 per minute.

Your program should prompt the user to enter an *account number* (**int**), a *type of service* (**char**), and the *number of minutes the service was used* (**int**).

A *type of service* of **R** means regular service; a *type of service* of **P** means premium service. Treat any other character as an error.

For the **premium service**, the customer may be using service both during the day and at night. Therefore, to calculate the bill, the program must ask the user to input the *number of minutes the service was used* during the day, and calculate the number of minutes used during the night.

Use **switch** statement for charge calculations with the *type of service* as the **switch variable**.

Your program should output the *account number*, *type of service*, *number of minutes the telephone service was used*, and the *amount due*. Use **DecimalFormat** for the output formatting. The code to print the results should be outside of the switch block, and it should not be run in case of error.

See sample runs below:

Run#1

```
Enter account number:
123
Enter service code:
b
Enter number of minutes:
100

Invalid service code: b
```

Run#2

```
Enter account number:
123
Enter service code:
r
Enter number of minutes:
40

Account number: 123
Service code: REGULAR
Minutes used : 40
Service charge : $10.00
```

Run#3

```
Enter account number:
123
Enter service code:
r
Enter number of minutes:
100

The number of minutes exceeded 50; extra charge applied.
Account number: 123
Service code: REGULAR
Minutes used: 100
Service charge : $20.00
```

Run#4

```
Enter account number:
123
Enter service code:
p
Enter number of minutes:
75
How many of the minutes are daytime minutes?
75

Account number: 123
Service code: PREMIUM
Minutes used: 75
Service charge : $25.00
```

Run#5

```
Enter account number:
123
Enter service code:
p
Enter number of minutes:
100
How many of the minutes are daytime minutes?
75

Account number: 123
Service code: PREMIUM
Minutes used: 100
Service charge : $25.00
```

Run#6

```
Enter account number:
123
Enter service code:
p
Enter number of minutes:
200
How many of the minutes are daytime minutes?
100

The number of day minutes exceeded 75; extra charge applied.
Account number: 123
Service code: PREMIUM
Minutes used: 200
Service charge : $27.50
```

Run#7

```
Enter account number:
123
Enter service code:
p
Enter number of minutes:
300
How many of the minutes are daytime minutes?
100

The number of day minutes exceeded 75; extra charge applied.
The number of night minutes exceeded 100; extra charge applied.
Account number: 123
Service code: PREMIUM
Minutes used: 300
Service charge : $32.50
```

[Bills.java.zip (https://cilearn.csuci.edu/files/623039/download)](https://cilearn.csuci.edu/files/623039/download)

Quiz Score: **26.5** out of 30

This quiz score has been manually adjusted by +2.0 points.