

MERCHANT'S GUIDE TO THE GALAXY

You decided to give up on earth after the latest financial collapse left 99.99% of the earth's population with 0.01% of the wealth. Luckily, with the scant sum of money that is left in your account, you are able to afford to rent a spaceship, leave earth, and fly all over the galaxy to sell common metals and dirt (which apparently is worth a lot).

Buying and selling over the galaxy requires you to convert numbers and units, and you decided to write a program to help you.

The numbers used for intergalactic transactions follows similar convention to the roman numerals and you have painstakingly collected the appropriate translation between them.

Roman numerals are based on seven symbols:

Symbol Value

[2]I [3]1

[4]V [5]5

[6]X [7]10

[8]L [9]50

[10]C [11]100

[12]D [13]500

[14]M [15]1,000

Numbers are formed by combining symbols together and adding the values. For example, MMVI is $1000 + 1000 + 5 + 1 = 2006$. Generally, symbols are placed in order of value, starting with the largest values. When smaller values precede larger values, the smaller values are subtracted from the larger values, and the result is added to the total. For example $\text{MCMXLIV} = 1000 + (1000 - 100) + (50 - 10) + (5 - 1) = 1944$.

The symbols "I", "X", "C", and "M" can be repeated three times in succession, but no more. (They may appear four times if the third and fourth are separated by a smaller value, such as XXXIX.) "D", "L", and "V" can never be repeated. "I" can be subtracted from "V" and "X" only. "X" can be subtracted from "L" and "C" only. "C" can be subtracted from "D" and "M" only. "V", "L", and "D" can never be subtracted. Only one small-value symbol may be subtracted from any large-value symbol. A number written in [16]Arabic numerals can be broken into digits. For example, 1903 is composed of 1, 9, 0, and 3. To write the Roman numeral, each of the non-zero digits should be treated separately. In the above example, 1,000 = M, 900 = CM, and 3 = III. Therefore, 1903 = MCMIII.

(Source: Wikipedia ([17]http://en.wikipedia.org/wiki/Roman_numerals)

Input to your program consists of lines of text detailing your notes on the conversion between intergalactic units and roman numerals.

You are expected to handle invalid queries appropriately.

Test input: glob is I prok is V pish is X tegj is L glob glob Silver is 34 Credits glob prok Gold is 57800 Credits pish pish Iron is 3910 Credits how much is pish tegj glob glob ? how many Credits is glob prok Silver ? how many Credits is glob prok Gold ? how many Credits is glob prok Iron ?

Test Output: pish tegj glob glob is 42 glob prok Silver is 68 Credits glob prok Gold is 57800 Credits glob prok Iron is 782 Credits

WHAT DO I SUPPOSE?

In order to develop the application, I could see that a unit intergalactic could be origin because of exist other previously.

Example 1:

glob is I

prok is V

glob prok SILVER is 28

With these inputs we will have:

| UNIT NAME | UNIT VALUE |
|---------------|----------------|
| Glob | I (1) |
| Prok | V (5) |
| SILVER | VII (7) |

Why?

Glob prok SILVER = 28.

IV SILVER = 28 (Transforming to Numeral roman)

(5 - 1) SILVER = 28, SILVER is a metal, therefore,

4SILVER = 28

SILVER = 28/4

SILVER = 7

Example2

glob is I

prok is V

pish is X

pish glob prok ?

| UNIT NAME | UNIT VALUE |
|-----------------------|------------|
| Glob | I (1) |
| Prok | V (5) |
| pish | X (10) |
| pish glob prok | 14 |

Why?

Pish Glob prok = XIV (Transforming to Numeral roman)

10 + (5 - 1) therefore,

pish glob prok = 14

HOW TO RUN

This is a JAVA challenge project using Eclipse as IDE, if you want to use another please remove the files/folders not supported. This is a MAVEN project so you can test it using any IDE.

Prerequisites

- [JDK 1.8]

- [[Maven](#)]

- [Eclipse]

How to run

- Install Eclipse.
- Import/Open the project on Eclipse.
- In order to verify project's compile please open command prompt in the root project's path and put mvn clean install

(Without tests) mvn clean install -Dmaven.test.skip=true

IMPORTANT!

- There is a file "units.txt" in the project's resources used to test the application, please, if you want to chose another please change the value for the properties in the Constants java file.

By default Constant.java contains

- PATH_FILE:/home/marcelo/eclipse-workspace/merchant-java/src/main/resources/
- FILENAME:units.txt

- If you want to change the content of units.txt, please dont forget to use next conditions

- INPUT STRUCTURE: This application has been developed ALWAYS with the next structure
"UNIT_NAME" IS "UNIT_VALUE"

- Do not put ?/CREDITS/Credits/credits like the last word
- If we have the word "HOW/How/how" inside a "UNIT_NAME" the applications will take as an input line request and the value to convert will be since "IS" word to the end.

Examples:

glob is I

prok is V

pish is X

tegi is L

glob glob Silver is 34

glob prok Gold is 57800

pish pish Iron is 3910

how much is pish tegj glob glob

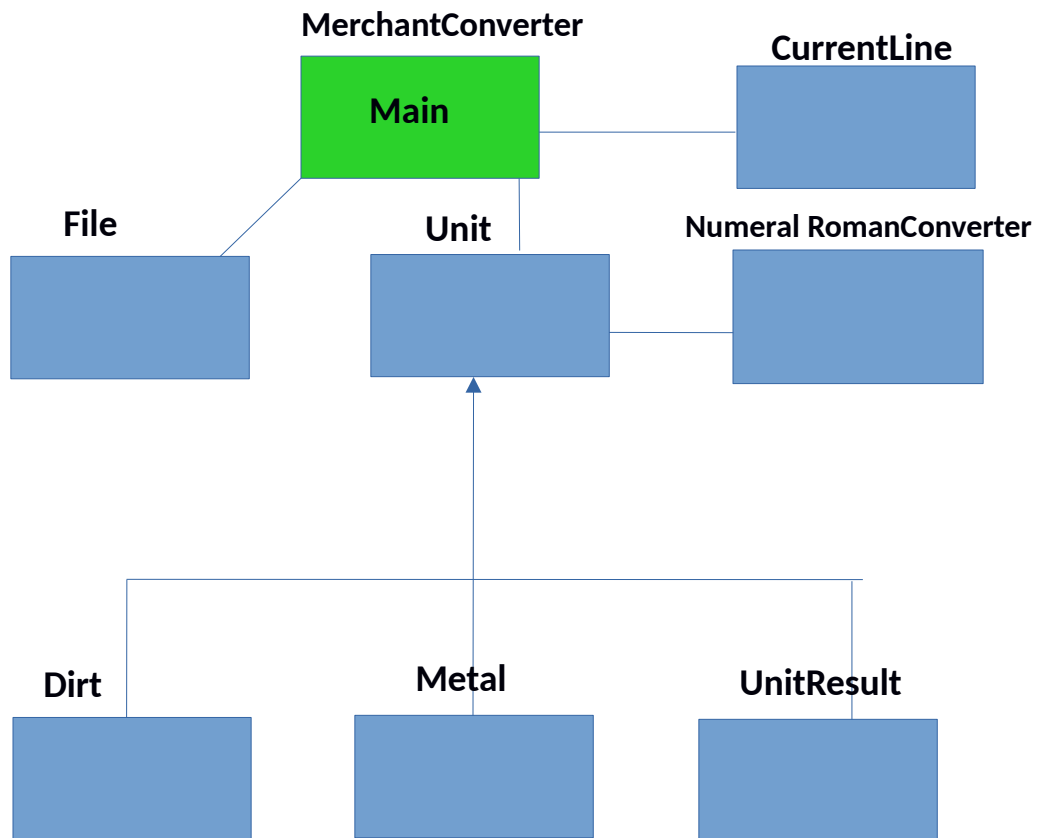
how many Credits is glob prok Silver

how many Credits is glob prok Gold

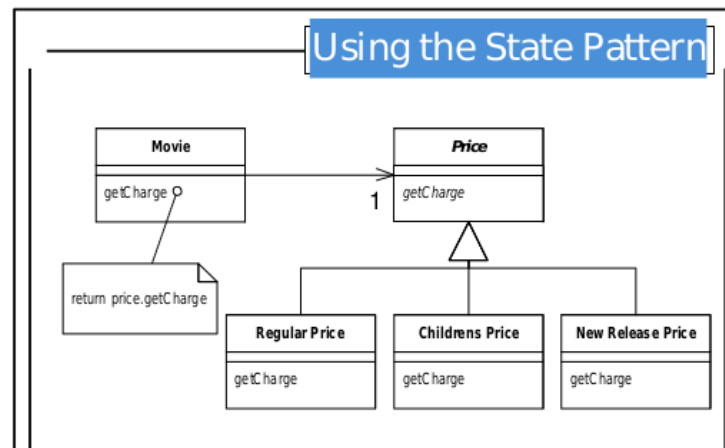
how many Credits is glob prok Iron

how much wood could a woodchuck chuck if a woodchuck could chuck wood

Project's design























As you can see, I have used a State pattern.



Project's estruture

```
└─ merchant-java
   └─ src/main/java
      └─ src/main/resources
      └─ src/test/java
         └─ src/test/resources
      └─ JRE System Library [J2SE-1.5]
      └─ Maven Dependencies
      └─ src
      └─ target
      └─ pom.xml
```

- ▼  merchant-java
 - ▼  src/main/java
 - ▶  com.merchant
 - ▶  com.merchant.enumerations
 - ▶  com.merchant.enumerations.util
 - ▶  com.merchant.util
 - ▼  src/main/resources
 -  LEEME.docx
 -  README.md
 -  units.txt

- ▼  src/test/java
 - ▶  com.merchant.enumerations.util.test
 - ▶  com.merchant.test
 - ▶  com.merchant.util.test
 -  src/test/resources
 - ▶  JRE System Library [J2SE-1.5]
 - ▶  Maven Dependencies
 - ▶  src
 - ▶  target
 -  pom.xml

src/test/java - merchant-java

TEST OF APPLICATION (mvn clean install)

Generate JAR

```
Problems Javadoc Declaration Search Console Call Hierarchy Debug
<terminated> merchant [Maven Build] /usr/local/java/jdk1.8.0_151/bin/java (Feb 6, 2018, 11:48:07 AM)
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building Merchant galactic converter 0.0.1-SNAPSHOT
[INFO] -----
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ merchant-java ---
[INFO] Deleting /home/marcelo/eclipse-workspace/merchant-java/target
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ merchant-java ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 1 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ merchant-java ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 14 source files to /home/marcelo/eclipse-workspace/merchant-java/target/classes
[INFO]
[INFO] --- maven-resources-plugin:2.6:testResources (default-testResources) @ merchant-java ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 0 resource
[INFO]
[INFO] --- maven-compiler-plugin:3.1:testCompile (default-testCompile) @ merchant-java ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 8 source files to /home/marcelo/eclipse-workspace/merchant-java/target/test-classes
[INFO]
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ merchant-java ---
[INFO] Surefire report directory: /home/marcelo/eclipse-workspace/merchant-java/target/surefire-reports

-----
T E S T S
-----
Running com.merchant.util.test.NumericalRomanConverterTest
```

```
<terminated> merchant [Maven Build] /usr/local/java/jdk1.8.0_151/bin/java (Feb 6, 2018, 11:48:07 AM)

T E S T S
-----
Running com.merchant.util.test.NumericalRomanConverterTest
Tests run: 7, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.065 sec
Running com.merchant.util.test.UnitUtilTest
Tests run: 6, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.02 sec
Running com.merchant.util.test.NumberUtilTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.01 sec
Running com.merchant.util.test.CurrentLineTest
Tests run: 4, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.006 sec
Running com.merchant.test.ConvertMerchantTest
***** RESULTS *****
pish tegj glob glob is 42.0
glob prok Silver is 68.0
glob prok Gold is 57800.0
glob prok Iron is 782.0
I have no idea what you are talking about
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.009 sec
Running com.merchant.test.FileTest
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.001 sec
Running com.merchant.test.UnitTest
Tests run: 13, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.003 sec
Running com.merchant.enumerations.util.test.MetalEnumUtilTest
Tests run: 2, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0 sec

Results :

Tests run: 39, Failures: 0, Errors: 0, Skipped: 0

[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ merchant-java ---
[INFO] Building jar: /home/marcelo/eclipse-workspace/merchant-java/target/merchant-java-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- maven-install-plugin:2.4:install (default-install) @ merchant-iava ---
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ merchant-java ---
[INFO] Building jar: /home/marcelo/eclipse-workspace/merchant-java/target/merchant-java-0.0.1-SNAPSHOT.jar
[INFO]
[INFO] --- maven-install-plugin:2.4:install (default-install) @ merchant-java ---
[INFO] Installing /home/marcelo/eclipse-workspace/merchant-java/target/merchant-java-0.0.1-SNAPSHOT.jar to /home/marcelo/.m2/repository/com/test/merchant-java/
[INFO] Installing /home/marcelo/eclipse-workspace/merchant-java/pom.xml to /home/marcelo/.m2/repository/com/test/merchant-java/0.0.1-SNAPSHOT/merchant-java-0.0.1-SNAPSHOT.pom
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 5.398 s
[INFO] Finished at: 2018-02-06T11:48:15-05:00
[INFO] Final Memory: 17M/200M
[INFO]
```


EXAMPLE UNIT TEST CASES

This is the example with just one test class with JUnit.

