

Contents

Provided Money and Currency API	2
Introduce.....	2
Setup	2
Diagram Class and Interface of JSR354.....	2
CurrencyUnit and MonetaryAmount.....	3
Demo.....	4

Provided Money and Currency API

Introduce

JSR354 targets to support all general applications: eCommerce, Banking, Finance...

The specification supports java se platform version 9 and above. The API is backward compatible with java 7,8. The reference implementation "**Moneta**" is modular and supports the Java Platform Module System. We will work it.

Setup

Link download: <https://mvnrepository.com/artifact/org.javamoney/moneta>

Or use Maven

```
<!-- https://mvnrepository.com/artifact/org.javamoney/moneta -->
```

```
<dependency>
```

```
    <groupId>org.javamoney</groupId>
```

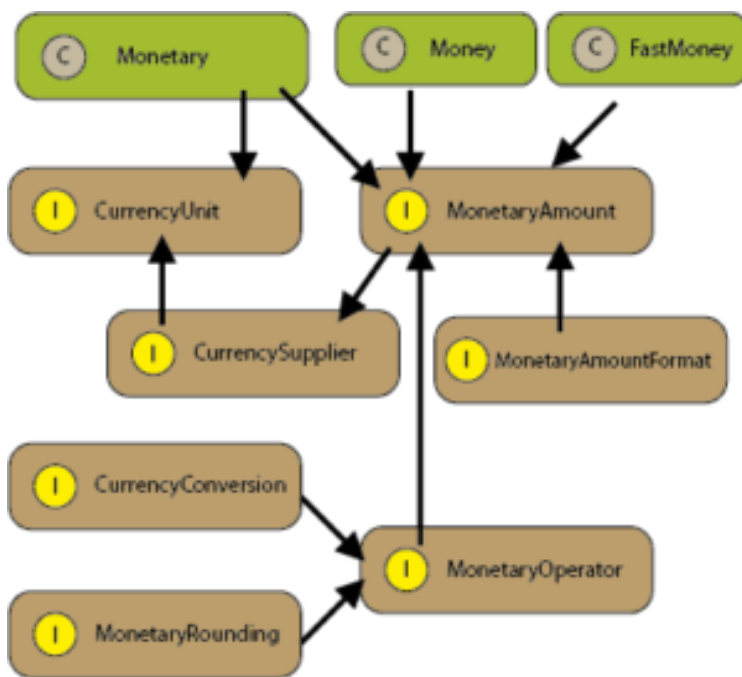
```
    <artifactId>moneta</artifactId>
```

```
    <version>1.4.2</version>
```

```
    <type>pom</type>
```

```
</dependency>
```

Diagram Class and Interface of JSR354



CurrencyUnit and MonetaryAmount

CurrencyUnit models a currency. It is very similar to the existing `java.util.Currency` class, except it allows custom implementations. That get instance **CurrencyUnit** by **MonetaryCurrencies** factory.

MonetaryAmount represents a concrete numeric representation of a monetary amount. A **MonetaryAmount** is always bound to a **CurrencyUnit**.

Money and **FastMoney** are two **MonetaryAmount** implementations of **JavaMoney**. **Monet** is default implementation that stores number values using **BigDecimal**. **FastMoney** is an alternative implementation which stores amounts in long fields.

According to documentation operation on **FastMoney** are 10-15 times faster compared to **Money**. However, **FastMoney** is limited by the size number and precision of the **Long** type. You can refer here about data type of java :

<https://docs.oracle.com/javase/tutorial/java/nutsandbolts/datatypes.html>

Demo

```
CurrencyUnitAndMonetaryAmount.java MonetaryFunctions.class Collectors.class
24 import org.javamoney.moneta.function.GroupMonetarySummaryStatistics;
25 import org.javamoney.moneta.function.MonetaryFunctions;
26 import org.javamoney.moneta.function.MonetarySummaryStatistics;
27
28 public class CurrencyUnitAndMonetaryAmount {
29
30
31     /**
32      * Create MonetaryAmount and CurrencyUnit
33      */
34     void createCurrencyUnitAndMonetaryAmount() {}
50
51     /**
52      * features: add, subtract, multiply, divide, round
53      */
54     void calculatorBasicMonetary() {}
87
88     /**
89      * Working with stream Check non null, Sorted, Filter, Group by CurrencyUnit
90      */
91     void calculatorAdvanced() {}
122
123     /**
124      * Convert Currencies
125      */
126     void exchangeRates() {}
127 }
```

GitHub:

<https://github.com/nguyenthinhit996/sharefullcode/blob/java/Learn%20Java/NewFeatureSE9/Money%20and%20Currency/DemoSource/src/main/java/com/sharefullcode/blogspot/CurrencyUnitAndMonetaryAmount.java>

This section is non commercial mainly sharing and advance knowlage for java.This tutorials has referenced document from the list below if you has complain for license, i will remove all from internet. Thank you all everything.

https://download.oracle.com/otn-pub/jcp/money_currency-1_1-mrel-eval-spec/JavaMoneySpecification.pdf

<https://www.baeldung.com/java-money-and-currency>

<https://dzone.com/articles/looking-java-9-money-and>