Money and Currency API

Pavel Rozenblioum, Jayway
@javatomte

Money and Currency API

- JSR 354
- JEP in the works
- RI Moneta http://javamoney.github.io/
- Main contributors: Anatole Tresch and Werner Keil

Why do we need it

- Money is important in many applications
- No standard type for money or support for conversions
- Formatting is limited

Overview of the API

- javax.money - MonetaryAmount, CurrencyUnit, MonetaryOperator, factory classes, exceptions…
- javax.money.conversion - CurrencyConversion, ExchangeRate
- javax.money.format - formatting
- javax.money.spi - service provider interfaces etc

```
<dependencies>
    <dependency>
        <groupId>org.javamoney</groupId>
        <artifactId>moneta</artifactId>
        <version>1.0-RC1</version>
    </dependency>
</dependencies>
```

Currency

```
CurrencyUnit sek = MonetaryCurrencies.getCurrency("SEK");

//SEK
```

Money

```
MonetaryAmount hundredSek = MonetaryAmounts
        .getDefaultAmountFactory()
        .setNumber(100.03523452435)
        .setCurrency(sek)
        .create();

// SEK 100.03523452435
```

Money

```
Money alsoHundredSek = Money
        .of(100.03523452435, sek);

// SEK 100.03523452435
```

FastMoney

```
FastMoney fastHundredSek = FastMoney
        .of(100.03523452435, sek);
```

```
// java.lang.ArithmeticException:
100.03523452435 can not be represented by this
class, scale > 5
```

FastMoney

```
FastMoney fastHundredSek = FastMoney
        .of(100.03523, sek);
```

```
//SEK 100.03523
```

Money + FastMoney

```
MonetaryAmount twoHundredSek =
        hundredSek.add(fastHundredSek);

// SEK 200.07046452435
// class Money
```

FastMoney + Money

```
MonetaryAmount twoHundredSek =
        fastHundredSek.add(hundredSek);
```

```
//java.lang.ArithmeticException: Parameter
exceeds maximal scale: 5
```

FastMoney + Money

```
Money roundedMoney = Money.of(100.03523, sek);
MonetaryAmount twoHundredSek =
        fastHundredSek.add(roundedMoney);

//SEK 200.07046
```

Mixing Currencies

```
CurrencyUnit dkk = MonetaryCurrencies
        .getCurrency("DKK");
Money hundredDKK = Money.of(100, dkk);
hundredSek.add(hundredDKK);

//javax.money.MonetaryException: Currency
mismatch: SEK/DKK
```

Converting Currencies

```
CurrencyConversion dkkConversion =
        MonetaryConversions.getConversion("DKK");
MonetaryAmount hundredSekInDKK =
        hundredSek.with(dkkConversion);
```

```
//DKK 79.54773808905498770439319685108
```

Formatting support

```
String britishHundredSek = MonetaryFormats
        .getAmountFormat(Locale.UK)
        .format(hundredSek);

//SEK100.04

String swedishHundredSek = MonetaryFormats
        .getAmountFormat(new Locale("sv"))
        .format(hundredSek);

//SEK 100,04
```

Custom formats

```
AmountFormatQuery formatQuery = AmountFormatQueryBuilder
        .of(new Locale("sv"))
        .set(CurrencyStyle.NAME)
        .set("pattern", "Money! ##,##.00 ¤")
        .build();
MonetaryAmountFormat customFormat =
        MonetaryFormats.getAmountFormat(
                formatQuery);
customFormat.format(Money.of(1_000_000, sek))
```

```
//Money! 1 00 00 00,00 svensk krona
```

Play around!

Thank you!

If you have any questions tweet @javatomte