Haushaltsbuch

Application Kata 'Haushaltsbuch'

Erstelle ein Programm zur Pflege eines Haushaltsbuches. Das Programm wird per Kommandozeile bedient. Mit folgendem Kommando wird eine Auszahlung getätigt:

```
C:>hb auszahlung 5.99 Kosmetikmatte Schlafzimmer

Kassenbestand: 94,01 EUR
Kassenbestand: 75,95 EUR
```

Der erste Parameter definiert, ob es sich um eine Auszahlung oder Einzahlung handelt. Bei einer Auszahlung wird der Betrag (in Kategorie und optional ein Memoeintrag) in „Schokoladenkeks"-Ausgegeben. Die Ausschlüsse kann auch mit einem anderen Datum versehen werden:

```
C:>hb auszahlung 01.01.2015 700 Miete

Kassenbestand: 125,01 EUR
Miete: 700,00 EUR
```

Zu Beginn, beim erstmaligen Aufruf des Programms, ist der Kassenbestand natürlich Null. Eine Einzahlung kann mit folgendem Befehl vorgenommen werden:

```
C:>hb einzahlung 600

Kassenbestand: 600,01 EUR
```

Der ausgegebene Betrag wird mit dem aktuellen Datum auf die Kasse gebucht. Anschließend wird der aktuelle Kassenbestand ausgegeben. Auch hier kann optional wieder ein anderes Datum angegeben werden:

```
C:>hb einzahlung 01.01.2015 400

Kassenbestand: 400,01 EUR
```

Um eine Übersicht über alle Ein- und Auszahlungen des aktuellen Monats zu erhalten, ruft man das Programm mit dem Befehl „Übersicht" auf:

```
C:>hb übersicht

März 2015

Kassenbestand: 34,20 EUR
Einzahlung: 600 EUR
Lebensmittelkauf: -94,99 EUR
Kosmetikmatte: -5,99 EUR
Einzahlung: 75,00 EUR
```

Um die Übersicht für einen anderen Monat auszuzeigen, können optional zwei weitere Parameter angegeben werden, nämlich Monat und Jahr:

```
C:>hb übersicht 12 2014

Dezember 2014

Kassenbestand: 125,70 EUR
Einzahlung: 200 EUR
Lebensmittelkauf: -123,67 EUR
Kosmetikmatte: -5,99 EUR
Einzahlung: 75,00 EUR
```

Variationen

Die Kategorien werden beim ersten Verwenden automatisch angelegt. Dies führt bei Tippfehlern dazu, dass Kategorien versehentlich angelegt werden. Daher soll kein erstmaliges Verwenden einer Kategorie eine Rückfrage erfolgen:

```
C:>hb auszahlung 7 "Bücher, Buch"

Soll die Kategorie "Bücher, Buch" neu angelegt werden? (j/n)...
```

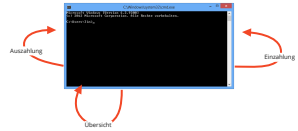
Bei der Eingabe von „j" wird die Kategorie angelegt und die Buchung durchgeführt. Bei Eingabe von „n" wird der Buchungsvorgang abgebrochen.

Nächster Schulungstermin: 20.05.2019

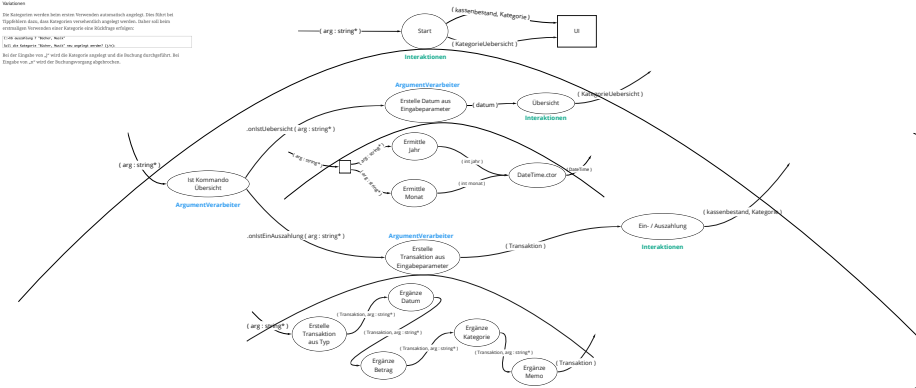
Übungsaufgabe: Haushaltsbuch
https://codeschool.de/coding-dojo/application-katas/haushaltsbuch/

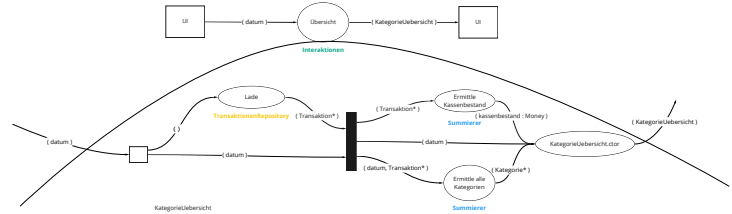
Zu erst einfache Implementation (ohne Variation)

Interaktionen



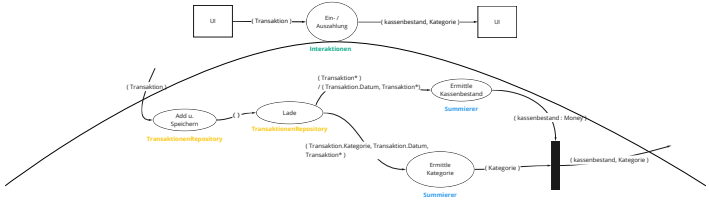

```
C:\>hb auszahlung [datum] betrag kategorie [memo]
C:\>hb einzahlung [datum] betrag
C:\>hb übersicht
```





KategorieUebersicht
==========
Datum : DateTime
Kassenbestand : Money
Kategorien : List<Kategorie>

```
C:\>hb auszahlung [datum] betrag kategorie [memo]
C:\>hb einzahlung [datum] betrag
```



Transaktion
==========
Typ : { Einzahlung, Auszahlung }
Datum : DateTime
Betrag : Money
Kategorie : string
Memotext : string

Kategorie
==========
Bezeichnung : string
Betrag : decimal

Money
==========
>> Library: NodaMoney
http://www.nodamoney.org/

Hinweis Währungen:

https://www.martinfowler.com/eaaCatalog/money.html

| P of EAA Catalog |

Money

Represents a monetary value.

For a full description see P of EAA page **488**

A large proportion of the computers in this world manipulate money, so it's always puzzled me that money isn't actually a first class data type in any mainstream programming language. The lack of a type causes problems, the most obvious surrounding currencies. If all your calculations are done in a single currency, this isn't a huge problem, but once you involve multiple currencies you want to avoid adding your dollars to your yen without taking the currency differences into account. The more subtle problem is with rounding. Monetary calculations are often rounded to the smallest currency unit. When you do this it's easy to lose pennies (or your local equivalent) because of rounding errors.

The good thing about object-oriented programming is that you can fix these problems by creating a Money class that handles them. Of course, it's still surprising that none of the mainstream base class libraries actually do this.