

Aufgabe 5: Listen (Check-Up)

Erstellen Sie ein Deutsch-Englisch Wörterbuch. Verwenden Sie dazu eine einfach verkettete Liste mit Kompositum. Identifizieren Sie die benötigten Klassen, legen Sie das Wörterbuch an und implementieren Sie anschließend die geforderten Methoden.

- Ein Listenelement, welches immer jeweils auf seinen Nachfolger verweisen kann, enthält jeweils einen Eintrag des Wörterbuchs. Ein Eintrag besteht aus dem deutschen und dem zugehörigen englischen Wort. Diese können natürlich jeweils zurückgegeben werden.
- Mit der Methode `einfuegen (String deutsch, String englisch)` soll ein neuer Eintrag in das Wörterbuch eingefügt werden können. Wie in jedem Wörterbuch müssen die (deutschen) Einträge jedoch alphabetisch sortiert sein, sodass nicht an einer beliebigen Stelle eingefügt werden kann. Um die korrekte Einfügeposition zu finden, ist das Vergleichen von Strings notwendig. Recherchieren Sie dazu, wie die Methode `compareTo()` in Java funktioniert!
- Der Aufruf der Methode `uebersetze(String deutsch)` auf der Liste soll nun für ein übergebenes deutsches Wort die englische Übersetzung ausgeben.

Lösung

Klasse WörterbuchEintrag

Die abstrakte Klasse im Kompositumentwurfsmuster von der sowohl die primitive Klasse als auch die Behälterklasse erben.

```
3 public abstract class WoerterbuchEintrag {
4     protected WortPaar nächstes;
5
6     protected WortPaar gibNächstes () {
7         return nächstes;
8     }
9
10    protected void setzeNächstes (WortPaar wortPaar) {
11        nächstes = wortPaar;
12    }
13 }
```

Klasse WortPaar

Das Listenelement (die primitive Klasse im Kompositumentwurfsmuster).

```
3 public class WortPaar extends WoerterbuchEintrag {
4     private final String deutsch;
5
6     private final String englisch;
7 }
```

```

8     public WortPaar(String deutsch, String englisch) {
9         this.deutsch = deutsch;
10        this.englisch = englisch;
11    }
12
13    public String gibDeutschesWort() {
14        return deutsch;
15    }
16
17    public String gibEnglischesWort() {
18        return englisch;
19    }
20 }

```

Klasse Wörterbuch

Die Behälterklasse im Kompositumentwurfsmuster.

```

6     public class Woerterbuch extends WoerterbuchEintrag {
7
8         public void einfügen(String deutsch, String englisch) {
9             WortPaar wort = new WortPaar(deutsch, englisch);
10
11             // Spezialbehandlung, wenn vor das erste Wortpaar des Wörterbuchs
12             ↪ eingefügt werden muss.
13             WortPaar kopf = gibNächstes();
14             if (kopf == null ||
15                 ↪ kopf.gibDeutschesWort().compareTo(wort.gibDeutschesWort()) >=
16                 ↪ 0) {
17                 wort.setzeNächstes(kopf);
18                 setzeNächstes(wort);
19                 return;
20             }
21             WortPaar vergleichsWort = gibNächstes();
22             while (
23                 vergleichsWort.gibNächstes() != null &&
24                 vergleichsWort.gibNächstes().gibDeutschesWort().compareTo(
25                     wort.gibDeutschesWort()
26                 ) < 0
27             ) {
28                 vergleichsWort = vergleichsWort.gibNächstes();
29             }
30             wort.setzeNächstes(vergleichsWort.gibNächstes());
31             vergleichsWort.setzeNächstes(wort);
32         }
33
34         public String übersetze(String deutsch) {
35             if (gibNächstes() == null) {
36                 return "Noch keine Wörter im Wörterbuch.";
37             }
38             WortPaar wort = gibNächstes();
39             while (wort != null) {
40                 if (wort.gibDeutschesWort().equals(deutsch)) {
41                     return wort.gibEnglischesWort();
42                 }
43                 wort = wort.gibNächstes();
44             }
45             return "Es konnte keine passende Übersetzung gefunden werden";
46         }
47     }

```

```

44
45 // Fleißaufgabe.
46 public void auflisten() {
47     WortPaar wort = gibNächstes();
48     while (wort != null) {
49         System.out.println(wort.gibDeutschesWort() + " <> " +
50             ↪ wort.gibEnglischesWort());
51         wort = wort.gibNächstes();
52     }
53 }
54
55 // Schauen, ob alles funktioniert.
56 public static void main(String[] args) {
57     Woerterbuch wörterbuch = new Woerterbuch();
58     System.out.println(wörterbuch.übersetze("Wassermelone"));
59
60     wörterbuch.einfügen("Wassermelone", "Watermelon");
61     System.out.println(wörterbuch.übersetze("Wassermelone"));
62
63     wörterbuch.einfügen("Apfel", "Apple");
64     wörterbuch.einfügen("Zitrone", "Lemon");
65     wörterbuch.einfügen("Birne", "Pear");
66     wörterbuch.einfügen("Klementine", "Clementine");
67
68     wörterbuch.auflisten();
69 }

```