

```

1 package tpe.collections.minions;
2
3 import tpe.collections.minions.domain.Minion;
4 import tpe.collections.minions.domain.MinionFarbe;
5 import tpe.collections.minions.domain.MinionIterable;
6
7 import java.util.HashSet;
8 import java.util.Set;
9
10 public class Main {
11     /**
12      * Gibt einen String zur Ausgabe aller errechneten Werte zurück.
13      *
14      * @param gesamtOD      Minions ohne Dubletten
15      * @param gelbOD        Gelbe Minions ohne Dubletten
16      * @param gelbeODP      Gelbe Minions ohne Dubletten in Prozent
17      * @param violettOD     Violette Minions ohne Dubletten
18      * @param violettODP    Violette Minions ohne Dubletten in Prozent
19      * @param gelbviolettODV Verhältnis Gelbe und Violette Minions
20      * @param gesamtMD      Minions mit Dubletten
21      * @param gelbMD        Gelbe Minions mit Dubletten
22      * @param gelbeMDP     Gelbe Minions mit Dubletten in Prozent
23      * @param violettMD     Violette Minions mit Dubletten
24      * @param violettMDP    Violette Minions mit Dubletten in Prozent
25      * @param gelbviolettMDV Verhältnis Gelbe und Violette Minions
26      * @param fehler        Fehler bei der Zählung
27      *
28      * @return Formatierter String mit allen wichtigen Infos
29     */
30     public static String outputString(
31         int gesamtOD, int gelbOD, double gelbeODP,
32         int violettOD, double violettODP, double gelbviolettODV, int gesamtMD,
33         int gelbMD, double gelbeMDP, int violettMD, double violettMDP,
34         double gelbviolettMDV, double fehler) {
35     /*
36      * Ohne Dubletten: 1128
37      * Minions 717 gelbe (64%) und 411 violett (36%),
38      * gelb/violett 1,74
39      * -----
40      * Mit Dubletten: 1184 Minions 758 gelbe (64%) und 426 violett (36%),
41      * gelb/violett 1,78
42      * -----
43      * Fehler bei der Zählung: 4,7%
44      */
45     return ("Ohne Dubletten: " + gesamtOD + " Minions\n" + gelbOD
46             + " gelbe (" + gelbeODP + "%) und " + violettOD + " violett (" +
47             + violettODP + "%), gelb/violett " + gelbviolettODV + "\n"
48             + "-----\n" +
49             + "Mit Dubletten: " + gesamtMD + " Minions\n" + gelbMD
50             + " gelbe (" + gelbeMDP + "%) und " + violettMD + " violett (" +
51             + violettMDP + "%), gelb/violett " + gelbviolettMDV + "\n"
52             + "-----\n" +
53             + "Fehler bei der Zählung: " + fehler + "\n");
54 }
55
56 /**
57  * ermittelt Gesamtanzahl Minions
58  *
59  * @param factory
60  *
61  * @return integer die Anzahl
62  */
63 public static int getAlleMD(MinionIterable factory) {
64     int anzahl = 0;
65     for (Minion m : factory) {
66         anzahl++;
67     }

```

```

68     return anzahl;
69 }
70
71 /**
72 * Zähle gelbe Minions mit Dupletten
73 *
74 * @param factory
75 *
76 * @return integer die Anzahl
77 */
78 public static int getGMD(MinionIterable factory) {
79     int anzahl = 0;
80     for (Minion n : factory) {
81         if (n.getFarbe() == (MinionFarbe.GELB)) {
82             anzahl++;
83         }
84     }
85     return anzahl;
86 }
87
88 /**
89 * Zähle violette Minions mit Dupletten
90 *
91 * @param factory
92 *
93 * @return integer die Anzahl
94 */
95 public static int getVMD(MinionIterable factory) {
96     int anzahl = 0;
97     for (Minion n : factory) {
98         if (n.getFarbe() == (MinionFarbe.VIOLETT)) {
99             anzahl++;
100        }
101    }
102    return anzahl;
103 }
104
105 /**
106 * Zähle alle Minions ohne Dupletten
107 *
108 * @param factory
109 *
110 * @return integer die Anzahl
111 */
112 public static int getAlleOD(MinionIterable factory) {
113
114     Set<Minion> set = new HashSet<Minion>();
115
116     for (Minion n : factory) {
117         set.add(n);
118     }
119
120     return set.size();
121 }
122
123 /**
124 * Zähle alle gelbe Minions ohne Dupletten
125 *
126 * @param factory
127 *
128 * @return
129 */
130 public static int getGOD(MinionIterable factory) {
131     Set<Minion> set = new HashSet<Minion>();
132
133     for (Minion n : factory) {
134         if (n.getFarbe() == MinionFarbe.GELB)

```

```

135             set.add(n);
136         }
137     }
138     return set.size();
139 }
140 
141 /**
142 * Zähle alle violetten Minions ohne Dupletten
143 *
144 * @param factory
145 *
146 * @return
147 */
148 public static int getVOD(MinionIterable factory) {
149     Set<Minion> set = new HashSet<Minion>();
150 
151     for (Minion n : factory) {
152         if (n.getFarbe() == MinionFarbe.VIOLETT)
153             set.add(n);
154     }
155 
156     return set.size();
157 }
158 
159 /**
160 * Berechnet Prozentwert der Eingabe
161 *
162 * @param gesamt 100 %
163 * @param wert wieviel prozent sind das?
164 *
165 * @return integer den Prozentsatz
166 */
167 public static double berechneProzent(double gesamt, double wert) {
168     return Math.round((100.0 * wert) / gesamt);
169 }
170 
171 /**
172 * Berechnet Differenz zweier Werte in Prozent
173 *
174 * @param gesamt alle zusammen
175 * @param ohneDupletten keine Dupletten enthalten
176 *
177 * @return integer die Anzahl ohne Dupletten
178 */
179 public static double berechneFehlerQuote(double gesamt, double ohneDupletten) {
180 
181     double tmp = gesamt - ohneDupletten;
182 
183     return Math.round((100.0 * tmp) / gesamt);
184 }
185 
186 /**
187 * Ermittelt das Verhältnis von a zu b
188 *
189 * @param a erster Parameter
190 * @param b zweiter Parameter
191 *
192 * @return double das Verhältnis von a zu b
193 */
194 public static double verhaeltniss(double a, double b) {
195 
196     return (double) Math.round(a / b * 100) / 100;
197 }
198 
199 public static void main(String[] args) {
200     MinionIterable factory = new MinionIterable();

```

```
202      // OHNE DUBLETEN
203      int gesamtOD = getAlleOD(factory);
204      int gelbOD = getGOD(factory);
205      double gelbODP = berechneProzent(gesamtOD, gelbOD);
206      int violettOD = getVOD(factory);
207      double violettODP = berechneProzent(gesamtOD, violettOD);
208      double gelbviolettODV = verhaeltniss(gelbOD, violettOD);
209
210      // MIT DUBLETEN
211      int gesamtMD = getAlleMD(factory);
212      int gelbMD = getGMD(factory);
213      double gelbMDP = berechneProzent(gesamtMD, gelbMD);
214      int violettMD = getVMD(factory);
215      double violettMDP = berechneProzent(gesamtMD, violettMD);
216      double gelbviolettMDV = verhaeltniss(gelbMD, violettMD);
217
218      //FEHLER
219      double fehler = berechneFehlerQuote(gesamtMD, gesamtOD);
220
221      System.out.println(outputString(
222          gesamtOD, gelbOD, gelbODP, violettOD, violettODP, gelbviolettODV,
223          gesamtMD, gelbMD, gelbMDP, violettMD, violettMDP, gelbviolettMDV,
224          fehler
225      ));
226
227
228  }
229 }
```