

Praktische Informatik I

Programmieren in Java

Übung 5

Objekte und Klassen

Aufgabe 1

Implementieren Sie die Klasse `Radio`, die über folgende Attribute verfügt:
`band` kann eine der Zeichenketten „UKW“, „MW“ oder „LW“ enthalten.
`lautstaerke` ist eine ganze Zahl zwischen 0 und 10.
`kanal` das Radio hat 5 Kanäle (von 1 bis 5).

Es existieren vier Konstruktoren:

- Der allgemeine Konstruktor übernimmt Werte für alle drei Attribute. Es wird geprüft, ob die Attributwerte gültig sind. Ist das Band nicht gültig, wird es mit „UKW“ belegt. Ist die Lautstärke nicht im gültigen Bereich, wird sie mit „1“ belegt. Der Kanal genauso.
- Der Standardkonstruktor erzeugt ein UKW-Radio, welches auf Lautstärke 1 und auf Kanal 1 eingestellt ist.
- Der Konstruktor mit einem String-Argument überprüft, ob der String eines der zugelassenen Bänder ist. Ist das nicht der Fall, wird ein UKW-Radio erzeugt. Die anderen Argumente werden wie beim Standardkonstruktor belegt.
- Der Konstruktor mit einem int-Argument bestimmt den Kanal, der bei Auslieferung gewählt ist. Lautstärke und Band werden wie beim Standardkonstruktor gewählt. Es wird überprüft, ob es sich um eine Zahl zwischen 1 und 5 handelt. Ist dies nicht der Fall, werden die Argumente wie beim Standardkonstruktor gewählt.

Führen Sie die Gültigkeitsüberprüfung möglichst nur einmal durch!

Der Methode `lautstaerkeErhoehen(...)` wird ein (positives) Argument übergeben, welches angibt um wie viele Einheiten die Lautstärke erhöht werden soll (Hinweis: nach 10 kommt 0). Es gibt keinen Rückgabewert.

Der Methode `lautstaerkeVermindern(...)` wird ein (positives) Argument übergeben, welches angibt um wie viele Einheiten die Lautstärke vermindert werden soll (Hinweis: nach 0 kommt 10). Es gibt keinen Rückgabewert.

Die Methode `kanalWechseln()` dient zum Hochschalten des Kanals. Nach Kanal 5 kommt wieder der Kanal 1. Es gibt keinen Rückgabewert.

Die Methode `ausgabe()` gibt die Attribute aus. Es gibt keinen Rückgabewert.

- 1) **Erstellen Sie ein UML Diagramm zu dieser Klasse.**
- 2) **Erzeugen Sie mehrere Objekte der Klasse `Radio` in der `main()` Methode dieser Klasse und testen die Klasse, z.B. wie folgt:**

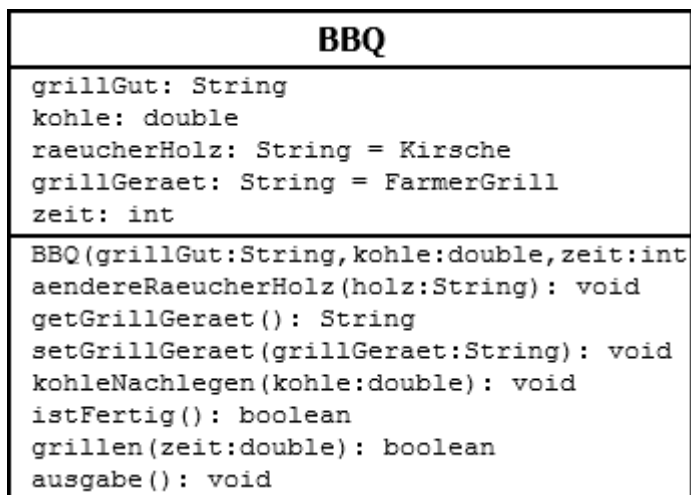
```

Radio kofferRadio = new Radio();
kofferRadio.ausgabe();
kofferRadio.lautstaerkeErhoehen(25);
kofferRadio.ausgabe();
Radio boomBox = new Radio("LKW");
boomBox.lautstaerkeVermindern(12);
boomBox.ausgabe();
boomBox = new Radio("MW");
boomBox.ausgabe();
Radio ghettoBlaster = new Radio (4);
Radio autoRadio = new Radio("UKW", 5, 5);
ghettoBlaster.ausgabe();
autoRadio.ausgabe();

```

Aufgabe 2

Sie stehen vor Ihrem Smoker und wollen ein Barbecue veranstalten. Zu Ihrer Unterstützung bei der Durchführung des Vorgangs schreiben Sie die folgende Klasse BBQ. Implementieren Sie die Klasse gemäß UML Diagramm:



Der allgemeine Konstruktor nimmt für jedes Attribut, außer `raeucherHolz` und `grillGeraet` ein Argument entgegen. `kohle` wird in Kilogramm und `zeit` in Minuten gemessen.

Die Methode `aendereRaeucherHolz()` ändert den Wert des entsprechenden Attributes.

Die Methoden `getGrillGeraet()` und `setGrillGeraet()` ändern den Wert des entsprechenden Attributes oder lesen ihn aus.

`kohleNachlegen()` erhöht die Kohlemenge um den entsprechenden Betrag.

`istFertig()` testet, ob die voreingestellte Zeit abgelaufen ist.

`grillen()` grillt die angegeben Zeit. Pro Stunde wird 1 Kilogramm Kohle verbraucht. Die Methode passt den Wert der Attribute `kohle` und `zeit` an und liefert `true` zurück, falls genügend Kohle zum Grillen da ist. Ansonsten wird `false` zurückgeliefert und es wird nicht gegrillt.

Die Methode `ausgabe()` gibt die Werte der Attribute aus. Ist das Essen fertig, soll dies vermeldet werden.

Achten Sie bei den Methoden darauf, dass sinnvolle Werte eingegeben werden!

Testen Sie die Klasse mit einer geeigneten `main()` Methode.