

Aufgabe 15.1: Lokale Klasse mit Klassennamen

Implementieren Sie eine Klasse `Email`, die folgende Aufgaben erfüllt:

- Die Klasse soll die Instanzvariablen `betreff`, `text`, `empfaenger` und `absender` vom Typ `String` enthalten.
- Der Konstruktor soll folgende Übergabeparameter besitzen:
 - `String empfaenger`
 - `String absender`
 - `String betreff`
 - `String text`

Der Konstruktor soll die Attribute mit den entsprechenden Übergabeparametern initialisieren.

- Schreiben Sie eine Methode `senden()`, die bei gültigen Mail-Adressen den Empfänger, den Sender, den Betreff und den Text auf dem Bildschirm ausgibt. Ist entweder die Mail-Adresse des Absenders oder die Mail-Adresse des Empfängers ungültig, so soll eine entsprechende Meldung auf dem Bildschirm ausgegeben werden. Um eine Mail-Adresse auf Gültigkeit zu überprüfen, verwendet die Methode `senden()` die Methode `isValid()` der lokalen Klasse `InternetMailAddress`.

Die Klasse `InternetMailAddress` soll als lokale Klasse in der Methode `senden()` der Klasse `Email` realisiert werden. Die lokale Klasse besitzt folgende Methode:

```
private boolean isValid (String address);
```

Diese Methode überprüft, ob eine übergebene Mail-Adresse gültig ist oder nicht. Gültige Adressen

- besitzen genau ein @-Zeichen, das den Namen des Empfängers vom Server trennt (`empfaengername@servername`)
- erlauben folgende Zeichen für Empfängername und Server: a-z, A-Z, 0-9, `_`, `.`
- erfordern mindestens ein Zeichen für Empfängername bzw. Server
- beginnen mit einem Zeichen, das nicht @ oder * ist
- erlauben das Weglassen von `@servername`, falls das letzte Zeichen ein '*' ist

Trifft die letzte Regel zu, so wird von der Methode `isValid()` als Servername für die Mail-Adresse `"it-designers.de"` eingetragen. **Beispiele:**

- | | |
|-----------------------------------|--|
| • <code>Hans.Muster@gmz.de</code> | gültig |
| • <code>Hans.Muster*</code> | gültig (wird zu <code>Hans.Muster@it-designers.de</code>) |
| • <code>@gmz.de</code> | ungültig |
| • <code>Hans.Mustergmz.de</code> | ungültig |

Zum Überprüfen der Gültigkeit einer Email-Adresse können Sie reguläre Ausdrücke in Kombination mit der Methode `matches()` der Klasse `String` verwenden. Konsultieren Sie hierzu die Java-Dokumentation. Testen Sie Ihre Klasse `Email` mit folgendem Programm:

```
// Datei: TestEmail.java
public class TestEmail
{
    public static void main (String[] args)
    {
        Email e1 = new Email ("Klaus.Gross*", "Lotte.Klein@gmz.de",
                               "Hallo", "Hallo Welt");

        e1.senden();
        Email e2 = new Email ("Klaus.Gross*", "Lotte.Kleingmz.de",
                               "Hallo", "Hallo Welt");

        e2.senden();
    }
}
```



Hier ein Auszug der Programmausgabe:

```
-----
Mail von: Lotte.Klein@gmz.de
An: Klaus.Gross@it-designers.de
Betreff: Hallo
Text:
Hallo Welt
-----
```

```
Die Email konnte nicht verschickt werden!
Die Email-Adresse des Absenders ist ungültig.
```