

## **Hausaufgaben 2**

**16./17.10.2024**

Abgabe der Lösung am 22.10.2024

### **E-Mail-Adressbuch**

Schreiben Sie eine Klasse `E-MailAdressBuch`, die zur Verwaltung von Personennamen und E-Mail-Adressen dient. Arbeiten Sie mit einer `HashMap`.

Die Klasse `E-MailAdressBuch` soll über folgende Methoden verfügen

- **public** `E-MailAdressBuch () {...}`
- **public** `void einfuegen(String name, String email) {...}`
- **public** `String abfrage(String name) {...}`
- **public** `String toString() {...}`

Mittels der Methode `einfuegen` kann ein neuer Eintrag in das E-Mail-Adressbuch eingefügt werden. Falls versucht wird, einen schon vorhandenen Namen einzufügen, soll nur die E-Mail-Adresse aktualisiert werden. Die Methode `abfrage` liefert die E-Mail-Adresse für einen gegebenen Namen zurück. Falls der Name nicht vorhanden ist, wird eine `NoSuchElementException` (unchecked) ausgelöst. Die `toString`-Methode gibt einen String der Form

```
{Paul=paul@worldwideweb.de, Maria=maria@gmy.de, Peter=peter@s-offline.de}
```

zurück.

### **Einlesen aus Datei**

Legen Sie sich eine Testdatei `emailadressbuch.txt` an, in die Sie beliebige Namen und Mail-Adressen in der Form

```
Name;EMail
```

per Hand eintragen. Fügen Sie der Klasse `E-MailAdressBuch` eine weitere Methode

```
public void einlesen(String dateiname)
```

hinzu, mit der Sie Ihre Testdatei einlesen und dem E-Mail-Adressbuch hinzufügen können.

### **Test**

Schreiben Sie einen Testfall, in dem die beiden Dateien `mitarbeiter_matse_intern.txt` und `mitarbeiter_matse_extern.txt` (auf der Veranstaltungsseite zu finden) in einem E-Mail-Adressbuch zusammengefügt und anschließend auf dem Bildschirm ausgegeben werden. Ein weiteres E-Mail-Adressbuch soll die Daten ihrer selbst erzeugten Testdatei enthalten.