**Aufgaben „FakePersonenGenerator“**

Zielsetzung der Aufgabe

Diese Aufgabe ermöglicht es, sich erneut mit einem Teil der vermittelten Inhalte zu befassen. Die optionalen Teile sind dabei Erweiterungen des Basis-Programms, die zusätzliche "vermittelte Inhalte" umfassen – sie setzen jedoch voraus, dass es ein funktionierendes Basis-Programm gibt.

Das entstehende Programm dient als Daten-Generator für das spätere Kapitel "Datenbankanbindung", daher ist es sinnvoll, dass zu diesem Zeitpunkt mindestens ein Basis-Programm bereitsteht.

Basis-Programm

Sie schreiben einen Datengenerator für die Erzeugung von Testdaten für Datenbanken. In der bereitgestellten Klasse Names finden Sie Arrays mit Vor- und Nachnamen, Städtenamen, Ländern, US-Bundesstaaten, Straßentyp-Suffixen (wie z.B. "-straße", "-weg", "-allee", allerdings auf Englisch), Himmelsrichtungen, lateinischen Worten und Top-Level-Domains. Sie dürfen diese Klasse verändern (sollten Sie auch).

Das Programm soll Personendaten erzeugen und in eine Textdatei schreiben (eine Person pro Zeile). Die Anzahl der zu erzeugenden Zeilen und der Dateiname der Ausgabedatei soll (zunächst) über eine Konstante im Quellcode festgelegt werden.

Dabei sollen die Datenfelder mit '|' getrennt werden. Eine Person hat die folgenden Datenfelder (dahinter stehen die Generierungsregeln):

Vorname: Ein Vorname, 25% der Personen hat zwei mit ' ' getrennte Vornamen

Nachname: Ein Nachname, 30% der Personen hat einen mit '-' verbundenen Doppelnamen

Wohnort: Ein Ortsname (auf eine Postleitzahl wird bewusst verzichtet!)

Strasse: Ein Straßenname, der aus einem Ortsnamen (50%) oder einem Nachnamen (50%) und einem Straßentyp-Suffix besteht. 40% der Straßen haben ein Himmelsrichtungs-Präfix ("*North* Monroe Str."), das mit ' ' abgetrennt ist.

Hausnummer: Eine Zahl mit 2-4 Stellen (als Zeichenkette)

E-Mail: 50% der Personen nutzen Fantasie-Adressen, die aus einem lateinischen Wort bestehen, 40% der Personen nutzen *vorname.nachname* und 10% *nachname*. Achten Sie darauf, dass E-Mail-Adressen keine Leerzeichen enthalten.

Die Domain zu einer E-Mail-Adresse ist immer ein lateinisches Wort, gefolgt von einem '.' und einer Top-Level-Domain.

Für die Erzeugung der Daten schreiben Sie möglichst Methoden (z.B. getRandomNachname() in der Klasse Names). Erforderliche zusätzliche Klassen legen Sie nach eigenem Ermessen an. Nutzen Sie Collections, wo immer es sinnvoll ist!

Erzeugen Sie zunächst alle Datensätze, und schreiben Sie diese erst danach in die Datei.

Erzeugen Sie zunächst alle Datensätze, und schreiben Sie diese erst danach in die Datei.

(Getrennt durch "|")

Optional 1 : GUI

Versehen Sie Ihr Programm mit einer GUI, in der der Name bzw. Pfad der Ausgabedatei (JTextField und/oder JFileChooser) und die Anzahl der zu erzeugenden Datensätze (JTextField, dabei auf Fehleingaben achten, wer mag JFormattedTextField) angegeben werden kann.

Optional 2 : Verschiedene Ausgabeformate

Erweitern Sie Ihr Programm und die GUI so, dass verschiedene Ausgabeformate unterstützt werden:

* in der GUI konfigurierbares Trennzeichen (statt fest einprogrammiertem '|').
* INSERT-Statements (in der GUI einstellbare Tabellen- und Spaltennamen und –Reihenfolge, z.B. mittels JList)
* [JSON](https://de.wikipedia.org/wiki/JavaScript_Object_Notation)