

Arbeitsblatt: DNET1

Name:

Kurznamen:

NextChange

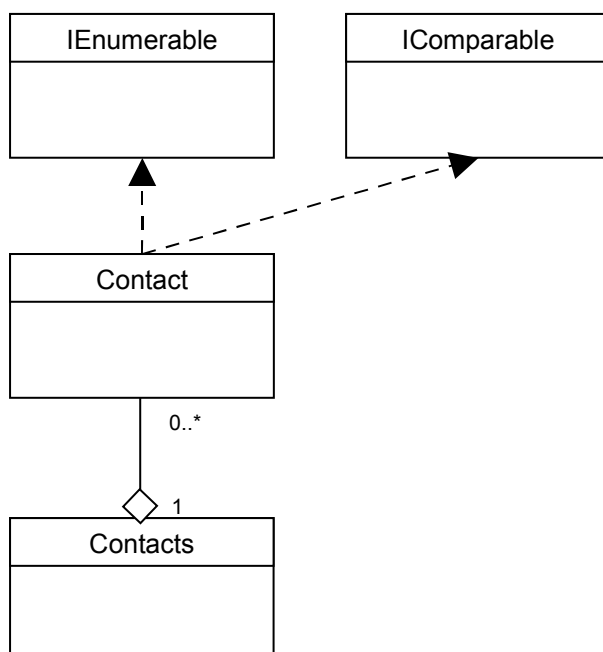
Nach den spannenden Physik fokussierten Aufgaben ein paar mehr "hausbackene" Anwendungsfälle.

In diesem und in den folgenden Praktika erstellen wir ein leichtgewichtiges (verteiltes) Adress- und Kalender-Werkzeug à la Exchange. Aber im Gegensatz zu Exchange werden wir – wo immer möglich – offene Standards verwenden.

Aufgabe 1 - Datenmodell

Im ersten Praktikum soll eine Adresssammlung (Namensliste der ZHAW Mitarbeiter) eingelesen und in eine interne Datenstruktur umgewandelt werden. Die Liste aller Kontakte und aller Felder so ein File geschrieben werden können (wieder als CSV). Weiter soll es möglich sein, einen einzelnen Kontakt als VCF zu erstellen und somit in ein anderes Programm (Outlook, iAddress) zu importieren.

Implementieren Sie für diesen Zweck folgendes Datenmodell



Contact soll das Interface **IComparable** implementieren, so dass nach dem Namen verglichen wird.

`Contact` soll ausserdem die Methode `GetEnumerator` implementieren. Diese Methode kann dann verwendet werden, um durch **die einzelnen Felder** eines `Contacts` zu iterieren (werden wir später noch brauchen).

Die `CompareTo` Methode soll so implementiert werden, dass nach dem Namen sortiert wird.

Hinweis:

- Die bereitgestellte Namensliste enthält nur Namen und Kurzzeichen. Wir werden später sehen, wie wir die Daten anhand via Web Crawling ergänzen.
- Beim Schreiben einfach über alle Felder iterieren i.e. die `ToString` Methode von `Contact` aufrufen.
- Um über alle Felder zu iterieren muss jedes einzelne Feld zurückgegeben werden. Dies geht - ohne die Verwendung von `Reflection` - leider nur durch copy-and-paste Programmierung. Später werden wir sehen, wie es einfacher geht.
- Nach der Rückgabe des Feldes `yield` verwenden

In `Contacts` werden die einzelnen Kontakte gespeichert. `Contacts` selber ist eine `Collection` und kann bzw. soll von einer bestehenden `Collection` der Klassenbibliothek abgeleitet sein. `Contacts` stellt weiter diverse Hilfs-Funktionen für die Adressverwaltung wie `readCSV(string filename)`, `writeCSV(string filename)` zur Verfügung.

Abgabe:

Praktikum: DN6.1

Datei: `Contact.cs`

Aufgabe 2 – VCF Format Converter

Viele Programme bieten das vCard Format für den Datenaustausch an.

- Einlesen der Daten aus der mitgelieferten Liste
- Implementieren Sie eine Methode `ToVcf ()` in `Contact`, die aus einem `Contact` einen vCard String erstellt - dieses kann dann in fast jedes beliebige Adressverwaltungsprogramm importiert werden.
- Verarbeiten Sie auch die Felder zu denen Sie noch keine Daten haben (z.B. Tel Nummer, Adresse); wir werden die Daten im Laufe der nächsten Praktika ergänzen.

Abgabe:

Praktikum: DN6.2

Datei: `Contact.cs`

Hinweise:

- VCard Format
- Zwischen den einzelnen Zeilen ein Zeilenunterbruch

local⁸ einfach finden

Müller Zürich

Müller Hans-Jürg
Zahnarzt

Grat⁸ anrufen Route berechnen Drucken Mehr

Dr. med. dent.

Giesshübelstrasse 69 /Brunaupark
8045 Zürich

Telefon 044 461 00 55

Eintrag ändern
Position des Eintrages aktualisieren
Adresse speichern (vCard)

```
BEGIN:VCARD
VERSION:3.0
N;CHARSET=ISO-8859-1:Müller Hans-Jürg;;;
FN;CHARSET=ISO-8859-1:Müller Hans-Jürg
X-ABShowAs:COMPANY
ADR;TYPE=work,pref;CHARSET=ISO-8859-1:;/Brunaupark;Giesshübelstrasse
69;Zürich;ZH;8045;
TEL;TYPE=work,voice,pref:+41444610055
EMAIL;TYPE=INTERNET:natalie.lynch@pencloud.com
END:VCARD
```