

Übungen - Woche 8

Vergessen Sie nicht, auch die Programmbeispiele und Aufgaben, die in den verschiedenen Units dieser Woche gestellt wurden, ebenfalls durchzuführen!

Aufgabe 1

Schreiben Sie eine Anwendung die eine beliebige, unsortierte Folge von Zahlen über Tastatur entgegennimmt und diese aufsteigend sortiert und ausgibt. Nutzen Sie dazu verschiedene Typen von Collections, um Erfahrung mit deren Umgang zu sammeln!

Aufgabe 2

In der Woche 7 haben Sie eine Klassenhierarchie Mitarbeiter mit Basisklasse `Mitarbeiter` und den abgeleiteten Klassen `Manager`, `Direktor`, `Engineer` und `Admin` aufgebaut. Vereinfachen Sie diese Hierarchie, indem Sie die beiden abgeleiteten Klassen `Engineer` und `Admin` zu einer Klasse `Angestellte` zusammenfassen.

Um die zugeordneten Angestellten verwalten zu können, erweitern Sie die Klasse `Manager` um die beiden Attribute `staff` sowie `mitarbeiterCount`.

Fügen Sie dann die Methoden `findMitarbeiter()`, `addMitarbeiter()`, `removeMitarbeiter()` ein.

Die Methode `findMitarbeiter()` überprüft, ob der Mitarbeiter bereits zugeordnet ist. Wenn eine Übereinstimmung gefunden wurde, geben Sie die Indexnummer zurück, ansonsten -1.

Schließlich erweitern Sie die Klasse um die Methode `displayStaff()`. Diese Methode gibt den Namen des Managers sowie alle ihm unterstellten Mitarbeiter zurück.

Zur Überprüfung Ihrer Implementierung schreiben Sie ein entsprechendes Testprogramm.

Sie können die Daten für das Definieren weiterer Mitarbeiter frei wählen.

Aufgabe 3

Erstellen Sie eine Anwendung, die nach Eingabe ein Wort in eine andere Sprache übersetzt. Das Wort soll von Deutsch nach Englisch und umgekehrt den gesuchten Begriff übersetzen und auf der Konsole ausgeben.

Falls das Wort sowohl im Englischen als auch im Deutschen existiert, so sollen diese mit ihrer jeweiligen Übersetzung in die andere Sprache erkenntlich ausgegeben werden.

Der Wortschatz der Anwendung soll mindestens 10 deutsche und 10 englische Wörter kennen, darunter mindestens zwei "falsche Freunde" (gleiches Wort, unterschiedliche Bedeutung).

Aufgabe 4

Erstellen Sie eine Datenstruktur, die für verschiedene Fahrzeuge die Fahrzeiten verschiedener Testfahrten erfasst. Dazu sind im Einzelnen das Fahrzeug sowie die erzielten Fahrzeiten zu verwalten.

Nachdem einige Fahrzeuge erfasst wurden, soll das Programm wiederholt für einzelne Fahrzeuge die Fahrzeiten erfassen und entsprechend in die obige Datenstruktur eintragen.

Es soll möglich sein, für einzelne Fahrzeuge sich die Liste der erreichten Fahrzeiten auf den Testfahrten anzeigen zu lassen.