



Softwarepraktikum im Grundstudium WS 2004/05

Webtechnologien

Aufgabenblatt 2

Testat am 5.11.2004

29.10.2004

Für die Aufgaben dieses Übungsblatts sollen Sie eine kleine Adressverwaltung implementieren. Die Adressverwaltung bietet die Möglichkeiten, Personen mit deren Adresse und Telefonnummer aufzunehmen, alle Personen aufzulisten und einzelne Personen wieder zu löschen.

Implementieren Sie die Personen in einer Java-Klasse Mensch. Die Attribute der Klasse sollen nur über Methoden (Getter- und Setter) verändert werden können. Implementieren Sie weiterhin eine Java-Klasse AdressDB in der die Personen verwaltet werden. Folgende Kommandos sollen implementiert werden:

1. Person hinzufügen
2. Person löschen
3. Liste aller Personen ausgeben
4. Exit

Verwenden Sie dazu ein Textmenü, so dass das Programm über Zahlen gesteuert werden kann. Dazu können Sie die Java-Klasse IO aus dem Repository.Common benutzen.

Lesen Sie das Aufgabenblatt vollständig durch, bevor Sie mit der Implementierung beginnen!

1. Aufgabe: Eclipse starten und konfigurieren

Wechseln Sie in das Verzeichnis /rbi/eclipse (es ist unter Umständen nicht vorhanden, aber durch den Wechsel wird es vom Automounter nachgeladen) und starten Sie mit

```
./eclipse -data ~/spgs0405/src
```

die Entwicklungsumgebung Eclipse. Mit dieser Entwicklungsumgebung sollen Sie die folgenden Übungen bearbeiten und sie so besser kennenlernen. Durch den oben beschriebenen Befehl richten Sie Ihren Workspace im Verzeichnis ~/spgs0405/src ein.

Erstellen Sie ein Projekt Uebung02. Beim Erstellen ändern Sie unter Configure Defaults den Output Folder von bin auf classes. Eclipse wechselt nun automatisch in die Java-Perspektive. Welche anderen Perspektiven kennt Eclipse, wofür dienen sie?

2. Aufgabe: Kommentare

Erstellen Sie in Eclipse die Klasse Mensch. Ein Kommentar-Header wird von Eclipse automatisch in die neue Datei geschrieben. Ändern Sie in Eclipse die Einstellungen der Dateiheader für neue Dateien, so dass sie die Praktikumsbezeichnung und Ihren Gruppen-Namen enthält. Erstellen Sie nun die Klasse AdressDB und tragen Sie in der Klasse Mensch den neuen Kommentar-Header nach.

3. Aufgabe: CVS in Eclipse

Wählen Sie aus dem Kontextmenü Team / Share Project und geben Sie als

Host: `cvs-i3.informatik.rwth-aachen.de`

Repository path: `/home/projects/spgs0405/Repository.GruppeXY`

Connection Type: `pserver`

sowie Benutzername und Passwort, die Sie am Anfang bekommen haben, ein. Als Modulname verwenden Sie `Uebung02`. Sehen Sie genau nach, welche Dateien zum Repository hinzugefügt werden sollen. Dies darf **nur einmal pro Gruppe** durchgeführt werden. Die anderen Mitglieder der Gruppe wählen unter File / New / Project den Punkt CVS / Checkout Projects from CVS aus. Verwenden Sie dabei die selben Werte wie oben angegeben.

Wechseln Sie in die CVS-Perspektive und sehen Sie sich mit Show Annotation die CVS-Annotationen an. Führen Sie von nun an regelmäßig nach Änderungen updates und commits durch.

4. Aufgabe: Code-Templates

Sehen Sie in Window / Prefences / Java / Editor nach, welche Code-Templates vordefiniert sind. Versuchen Sie mit `<STRG>+<LEERTASTE>` diese zu verwenden. Implementieren Sie nun die beiden Java-Klassen.

5. Aufgabe: Refactoring

Nennen Sie die Datei `Mensch.java` in `Person.java` um. Verwenden Sie dazu die Funktion Refactor / Rename aus dem Kontextmenü der Datei. Welchen Vorteil bietet dieses Feature?

6. Aufgabe: Ausführen

Führen Sie Ihr Programm mit Run / Run as / Java Application aus. Es öffnet sich eine Konsole in Eclipse, in der Sie auch Eingaben machen können. Testen Sie ausführlich, ob Ihr Programm den obigen Anforderungen entspricht.

7. Aufgabe: Debug

Setzen Sie in die Schleife zur Ausgabe aller Personen einen Breakpoint, indem Sie in der Java-Perspektive mit der rechten Maustaste auf den linken Rand neben der Schleife klicken und Toggle Breakpoint auswählen. Starten Sie nun eine Debug-Session mit Run / Debug as / Java Application. Eclipse wechselt automatisch in die Debug-Perspektive. Mit F6 gehen Sie nun schrittweise durch die Schleife. Achten Sie darauf was passiert.

8. Aufgabe: Javadoc

Recherchieren Sie auf der Homepage von SUN in Internet, wozu Javadoc verwendet werden kann und prüfen Sie, wie man den Javadoc-Compiler auf der Kommandozeile aufruft.

Finden Sie heraus, welche Javadoc-Platzhalter es gibt. Fügen Sie Ihrem Programm eine ausführliche Dokumentation mit den gefundenen Javadoc-Platzhaltern hinzu. Beschreiben Sie alle öffentlichen Variablen und Methoden. Alle Methoden, die eine gewissen Komplexität haben, sollen ebenfalls kurz aber verständlich kommentiert werden.

Übersetzen Sie Ihr Projekt mit Javadoc und sehen Sie sich das Ergebnis mit dem Javadoc-View in Eclipse an. Sehen Sie es sich ebenso im Browser an.

9. Aufgabe: Onlinehilfe

Selektieren Sie innerhalb der Datei AdressDB.java eine Variable der Klasse Person. Sehen Sie sich die Onlinehilfe der Variable an und öffnen Sie die Deklaration.

10. Aufgabe: Versionsverwaltung

Stellen in der Versionsverwaltung ein, dass die Dokumentation nicht unter Versionskontrolle gestellt wird. Warum nicht?