# Übung: Schulung Implementierung

Hinweis: Beachten Sie zur Bearbeitung der nachfolgenden Aufgabenstellungen das Informationsmaterial

*L3\_3\_2 Info Assoziation Implementierung.docx*

**(I) Problemstellung**

Die fit-for-future GmbH ist ein Unternehmen, das unternehmensunabhängige Schulungen im Bereich Unternehmensführung anbietet. Zur Verwaltung der Schulungen soll eine objektorientierte Software verwendet werden, wofür bereits ein UML-Klassendiagramm erstellt wurde.

1. Implementieren Sie die Klassen *Raum* und *Teilnehmer*. Speichern Sie die Klassen im Ordner L3\_3\_2 Loesung\_Schulung wie folgt: raum.py und teilnehmer.py
2. Implementieren Sie anhand des UML-Klassendiagramms die Klasse *Referent*. Speichern Sie anschließend die Klasse in einer Datei referent.py im Ordner L3\_3\_2 Loesung\_Schulung.

Die Methode *schulungen\_ausgeben()* soll die Bezeichnung aller Schulungen des Referenten ausgeben.

|  |
| --- |
| Formdesign: *schulungen\_ausgeben()* |
| Schulungen von Kaya Aktas:  Mitarbeitergespräch |

1. Implementieren Sie anhand des UML-Klassendiagramms die Klasse *Schulung*. Speichern Sie anschließend die Klasse in einer Datei schulung.py im Ordner L3\_3\_2 Loesung\_Schulung.

Die Methode *mindestteilnehmer\_pruefen()* soll bei einer Teilnehmeranzahl von unter fünf Teilnehmern ausgeben, dass die Mindestteilnehmeranzahl noch nicht erfüllt ist.

|  |
| --- |
| Formdesign: *mindestteilnehmer\_pruefen()* |
| Mindestteilnehmerzahl noch nicht erfüllt. |

Die Methode *teilnehmer\_hinzufuegen()* soll einen Teilnehmer der Schulung zuordnen. Dabei ist zu beachten, dass die Räumlichkeiten der fit-for-future GmbH maximal für 16 Teilnehmer ausgelegt sind. Außerdem soll überprüft werden, ob die Mindestteilnehmerzahl von fünf Personen noch unterschritten ist. Ist die Schulung ausgebucht (Teilnehmerzahl 16), soll dies auf der Konsole ausgegeben werden.

|  |
| --- |
| Formdesign: *teilnehmer\_hinzufuegen()* |
| Max Mustermann wurde hinzugefügt.  Mindestteilnehmerzahl noch nicht erfüllt.  oder  Max Mustermann wurde hinzugefügt.  oder  Die Schulung ist voll! |

Die Methode *abrechnung\_erstellen()* soll die Schulungsdaten (Referent, Bezeichnung und Teilnehmeranzahl) und den Gewinn der Schulung ausgeben.

|  |
| --- |
| Formdesign: *abrechnung\_erstellen()* |
| Referent: Aktas Kaya  Schulung: Mitarbeitergespräch  Teilnehmeranzahl: 6  ---------------------------------------------  Abrechnung  Umsatz: 1500.0 EUR  Honorar: 1000.0 EUR  Gewinn: 500.0 EUR |

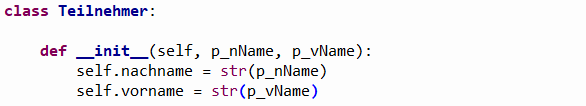
1. Testen Sie ihren Programmcode, indem Sie das Hauptprogramm ausführen:

* Erstellen Sie einen Raum „R233“
* Erstellen Sie einen Referenten Kaya Aktas mit einem Honorar von 1000 EUR.
* Erstellen Sie eine Schulung „Mitarbeitergespräch“ mit einer Gebühr von 250 EUR in Raum R233 mit dem Referenten Kaya Aktas.
* Fügen Sie die Teilnehmer „Simon Amsel“, „Laura Ackermann“, „Can Yasin“, „Inga Yefremova“, „Josephin Thompson“ und „Marie McEwen“ der Schulung hinzu.
* Führen Sie die Methoden *abrechnung\_erstellen()* und *schulungen\_ausgeben()* für die entsprechenden Objekte aus.

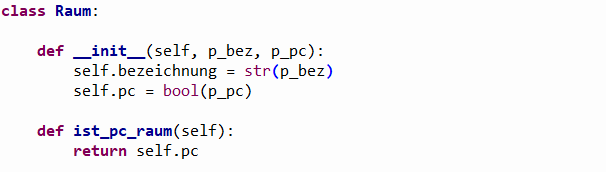
**(II) UML-Diagramm**



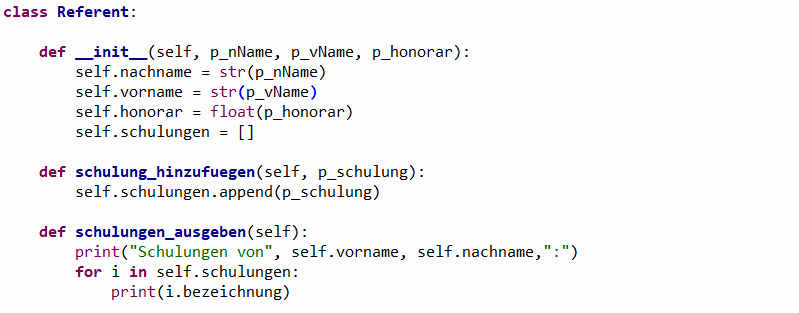
**(III) Programmcode (Python-Code)**



**Datei:** teilnehmer.py



**Datei:** raum.py



**Datei:** referent.py



**Datei:** schulung.py