

U13 Rollen-basierte Programmierung mit ObjectTeams/Java

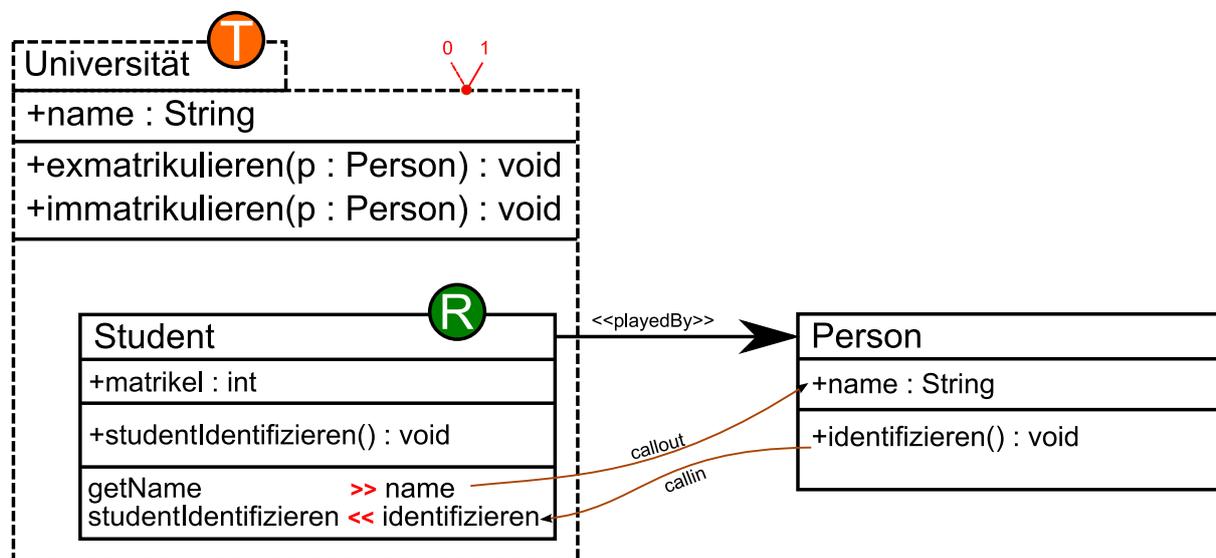
Inhalt der Übung

- Einführung in die Konzepte von ObjectTeams/Java

Übungsaufgaben

Personen können innerhalb der Universität durch Immatrikulation die Rolle Student aufnehmen und sie durch Exmatrikulation wieder ablegen. Eine Person wird mit ihrem Namen identifiziert. Ein Student hingegen über seine Matrikelnummer.

Das folgende modifizierte Klassendiagramm stellt diesen Sachverhalt in der Sprache ObjectTeams/Java dar.



Laden Sie sich die Beispielanwendung aus der Einführung herunter und analysieren Sie sie.

Aufgabe 1 (Professor)

Erweitern Sie die Beispielanwendung um die Rolle Professor. Ein Professor wird über seinen Lehrstuhl (String) identifiziert. Hans und Anja bleiben weiterhin Studenten, Peter jedoch wird Professor. Die Ausgabe des Testprogramms soll ungefähr so aussehen:

```
---Uni aktiv-----
Matrikelnummer: 123
Hans
Matrikelnummer: 345
Anja
Peter ist Professor am Lehrstuhl Softwaretechnologie.
```

Überlegen Sie wie Sie innerhalb der Rolle Professor an den Namen der Person kommen.

Hinweis: Wenn Sie bei einem callin „replace“ nutzen, muss die Rollenmethode statt der Sichtbarkeit das Schlüsselwort „callin“ verwenden.

Aufgabe 2 (Lehrveranstaltungen)

Erweitern Sie das Beispiel um *Lehrveranstaltungen*, die von Studenten besucht und von Professoren gehalten werden. Beschränken Sie sich dabei auf *Vorlesungen* und lassen Sie Übungen außen vor.

- Das Team `Universität` soll um die Methoden `vorlesungAnbieten(...)` und `einschreiben(...)` erweitert werden.
- Die Rolle `Student` soll eine Methode `alleBesuchtenVorlesungen()` enthalten, welche sämtliche von dem jeweiligen Studenten besuchten Lehrveranstaltungen zusammen mit dem Namen des zuständigen Professors ausgibt.
- Die Rolle `Professor` soll eine Methode `alleStudentenMeinerVorlesungen()` enthalten, die die Namen sämtlicher Studenten die eine (oder mehrere!) seiner Lehrveranstaltungen besuchen ausgibt.
- Erweitern Sie das Team `Universität` um die Methoden `besuchteVorlesungenVonStudent(Person p)` und `alleStudentenEinesProfessors(Person p)` um über das Team vom Testprogramm aus, auf die vorher angelegten Rollenmethoden zugreifen zu können.
- Erweitern Sie das Testprogramm um die neu hinzugefügten Methoden zu überprüfen.

Hinweise:

- Betrachten Sie den Typ `Vorlesung` als Rollentyp, mit dem Kern `Lehrveranstaltung`.
- Verwenden Sie die reflektive Methode `hasRole(Basisobjekt, Rollentyp)` der Klasse `Team` um in `Team`-Methoden zu überprüfen, ob ein Basisobjekt eine Rolle spielt.
- Um die Erzeugung von Rolleninstanzen zu verhindern, können die Parameter von `Team`-Methoden mit Basisklassen getypt sein und innerhalb der Methode mit Hilfe der reflektiven `Team`-Methode `getRole(Parameter, Rollentyp)` die entsprechenden Rolleninstanzen geholt werden.

Literaturempfehlung:

Stephan Herrmann, Christine Hundt, Marco Mosconi: *ObjectTeams/Java Language Definition Version 1.2*. TU Berlin. 2008 (<http://objectteams.org/def/1.2/OTJLDv1.2-current.pdf>)

Stephan Herrmann: *A Precise Model for Contextual Roles: The Programming Language ObjectTeams/Java*. Applied Ontology, Volume 2, Number 2 / 2007, pp. 181-207, IOS Press. 2007 (<http://objectteams.org/publications/JAO07.pdf>)

Links:

Eclipse-Download: <http://eclipse.org/downloads>
ObjectTeams: <http://www.objectteams.org>