

## Modellierung mit Objekten

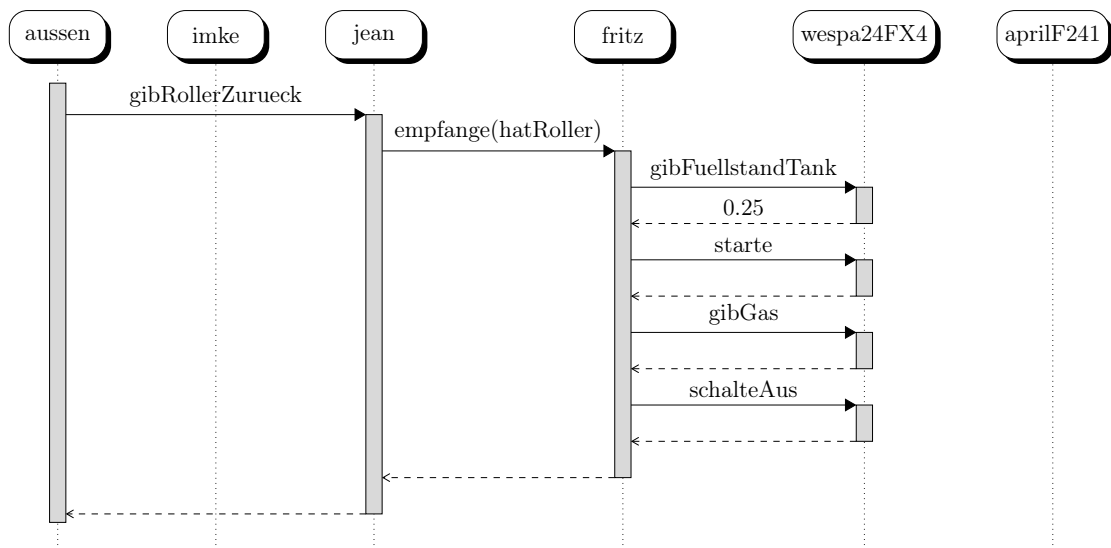
### Aufgabe 1 (25 Punkte)

Der 30-jährige Dieter und seine zwei Jahre jüngere Frau Margret befinden sich im Urlaub und möchten sich jeweils einen Roller mieten. Sie sind an Manni und seinen Kollegen Franz geraten. Dieter bittet Manni, die Formalitäten zu erledigen. Manni überreicht den beiden Urlaubern jeweils ein blaues Formular, das beide in Empfang nehmen und ausfüllen. Nach dem Ausfüllen geben sie die Formulare an Franz, da Manni mittlerweile sein Lieblingslied »[Hoch auf dem gelben Wagen]« singt.

- Lesen Sie die obige Situationsbeschreibung.
- (10) Ermitteln Sie mit Hilfe des *Verfahrens von Abbott* alle in der obigen Situationsbeschreibung vorkommenden Objekte mitsamt ihren Attributen und Methoden.
- (10) Entwerfen Sie für jedes Objekt eine Objektkarte für die Situation, unmittelbar nachdem Manni den beiden Urlaubern jeweils ein Formular übergeben hat.
- (5) Stellen Sie die Beziehungen der Objekte untereinander als Objektdiagramm dar.

## Sequenz- und Klassendiagramme

### Aufgabe 2 (15 Punkte)



- (5) Erweitern Sie das Sequenzdiagramm um die folgende Situationsbeschreibung, die sich im Anschluss abspielen soll:

Imke begutachtet ihren geliehenen Roller »AprilF241«. Sein Zustand ist offenbar, dass er kaputt ist. Sie fragt den Verleiher Fritz nach den Kosten für einen neuen Roller, der ihr »300 €« antwortet.

- b) (10) Modellieren Sie auf der Basis der Information, die Sie dem Sequenzdiagramm (inkl. Ihrer Erweiterung aus Aufgabenteil a) entnehmen können, ein entsprechendes Klassendiagramm mit Beziehungen und Kardinalitäten. Konstruktoren bzw. Destuktoren müssen nicht mit angegeben werden. Ergänzen Sie die Klassen aber um geeignete Attribute.

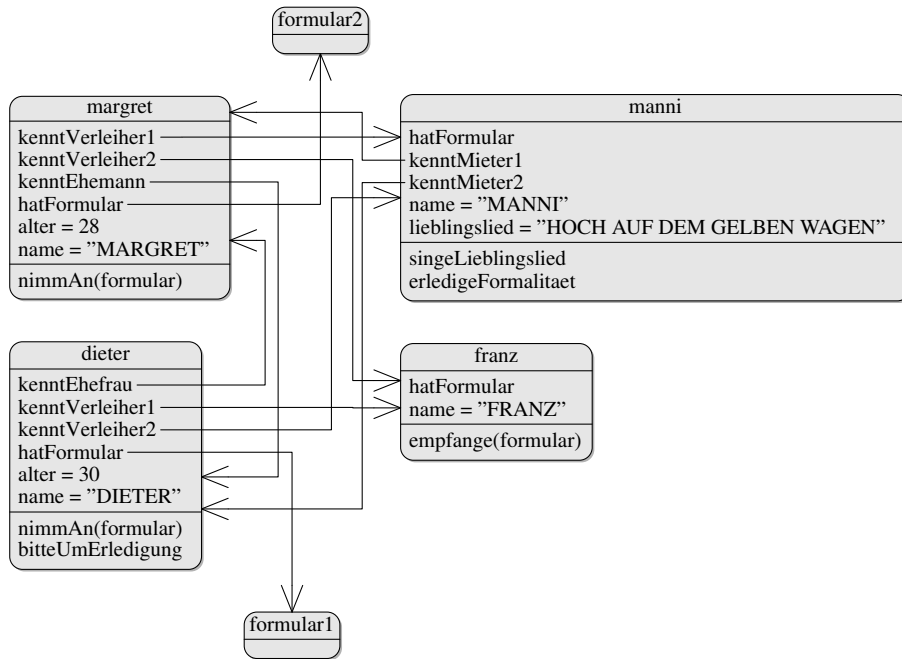
### Aufgabe 3

Implementieren Sie die Methode `empfange(roller)`. Der Methodenkopf ist vorgegeben, Sie müssen Anfragen/Aufträge einfügen.

```
# Methode empfange  
def empfange (self , roller):
```

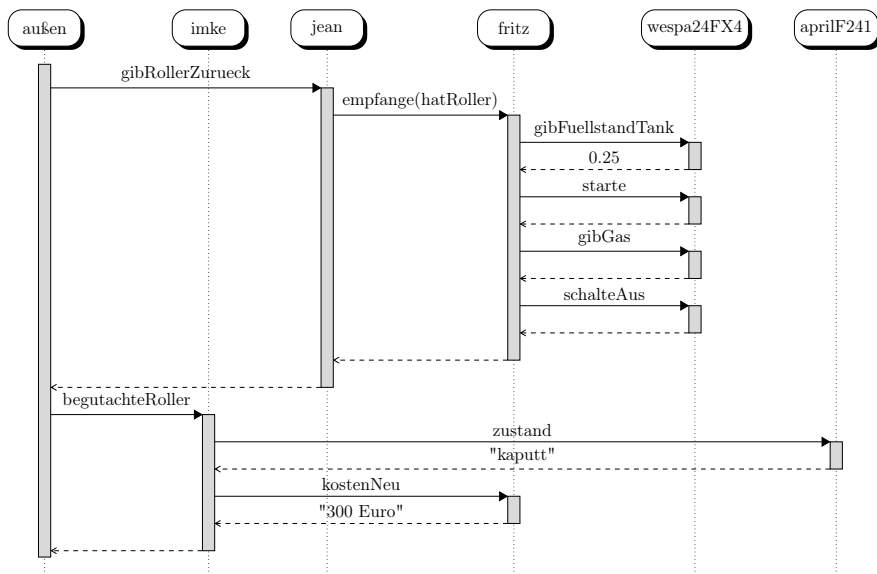
## Lösungen zu den Aufgaben

### Lösung 1 (25 Punkte)

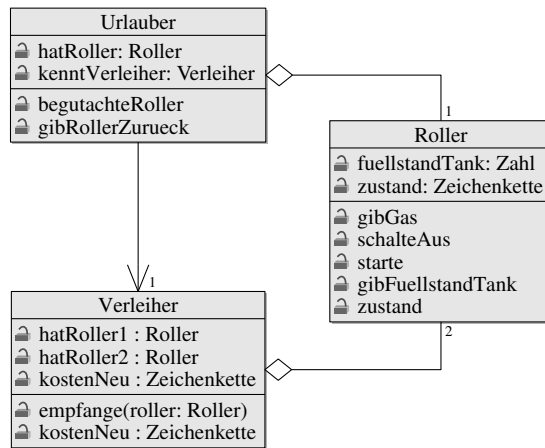


### Lösung 2 (15 Punkte)

#### a) Erweitertes Sequenzdiagramm:



#### b) Klassendiagramm:



### Lösung 3

*# Methode empfange*

```
def empfange (self , roller):
    self.hatRoller= roller
    print(roller.gibFuellstandTank())
    roller.starte()
    roller.gibGas()
    roller.schalteAus()
```

## Erwartungshorizont

Die Schülerin/der Schüler...	max. Punkte	erreicht
<b>Aufgabe 1</b>	<b>25</b>	
ermittelt die Objekte <code>margret</code> , <code>dieter</code> , <code>franz</code> , <code>manni</code> , <code>formular1</code> und <code>formular2</code> .	3	
ermittelt die Attribute <code>alter</code> , <code>hatFormular</code> , <code>lieblingslied</code> und ordnet sie korrekt zu.	3	
ermittelt die Methoden <code>nimmAn(formular)</code> , <code>erledigeFormalitaet</code> und <code>singeLied</code> .	4	
entwirft ein, der Situation entsprechendes, Objektdiagramm.	5	
stellt das Objektdiagramm den Konventionen entsprechend dar: <ul style="list-style-type: none"> <li>• Objektnamen klein</li> <li>• Attributwerte groß</li> <li>• Aufträge im Imperativ</li> <li>• Präfixe bei den Attributen</li> <li>• Anführungszeichen bei Zeichenketten</li> </ul>	5	
stellt die Beziehungen zwischen den Objekten vollständig und korrekt dar.	5	
<b>Aufgabe 2</b>	<b>15</b>	
gibt die Aufträge <code>begutachteRoller</code> , <code>zustand</code> und <code>kostenNeu</code> , sowie die zugehörigen Antworten an und erweitert das Diagramm entsprechend.	5	
modelliert ein zu den vorherigen Aufgaben konsistentes Klassendiagramm.	8	
stellt das Klassendiagramm unter Berücksichtigung der geltenden Vereinbarungen korrekt dar.	2	
<b>Aufgabe 3</b>	<b>10</b>	
implementiert eine dem Klassendiagramm entsprechende Methode unter Verwendung der notwendigen Anfragen und Aufträge.	8	
wendet beim Implementieren die korrekte Syntax an.	2	
<b>Gesamt</b>	<b>50</b>	

## Notenverteilung

Note	$\geq P.$	Note	$\geq P.$	Note	$\geq P.$	Note	$\geq P.$
sehr gut plus	38	gut	30	befriedigend minus	22	mangelhaft plus	13
sehr gut	36	gut minus	28	ausreichend plus	20	mangelhaft	11
sehr gut minus	34	befriedigend plus	26	ausreichend	18	mangelhaft minus	8
gut plus	32	befriedigend	24	ausreichend minus	16	ungenügend	0