

28.01.2013	<b>Advanced Software Engineering</b> <b>University of Vienna</b> <b>Forschungsgruppe Software Architecture</b>	
<i>Kennzahl</i>	<i>Matrikelnummer</i>	<i>FAMILIENNAME</i> <i>Vorname</i>

12	8	4	15	8	7	6	$\Sigma$ 60
1	2	3	4	5	6	7	

---

Bitte lesen Sie diesen Abschnitt. Sie haben **60** Minuten Zeit die Fragen zu beantworten.

**Hinweise:**

- Füllen Sie Kennzahl, Matrikelnummer, Familienname und Vorname zuerst aus.
  - Während der Prüfung sind keinerlei Unterlagen erlaubt!
  - Technische Hilfsmittel wie Übersetzungscomputer, Taschenrechner, Mobiltelefone, etc. sind nicht erlaubt!
  - Wenn Sie Probleme beim Verstehen einer Frage haben, fragen Sie.
  - Sie können die Fragen sowohl auf Deutsch als auch auf Englisch beantworten.
  - Schreiben Sie mit dokumentenechten Schreibutensilien (kein Bleistift).
  - Sie können den Appendix (letzte Seite) von der Prüfung abtrennen.
  - Viel Erfolg!
-

## Task 1 / Aufgabe 1 (12 Points / Punkte)

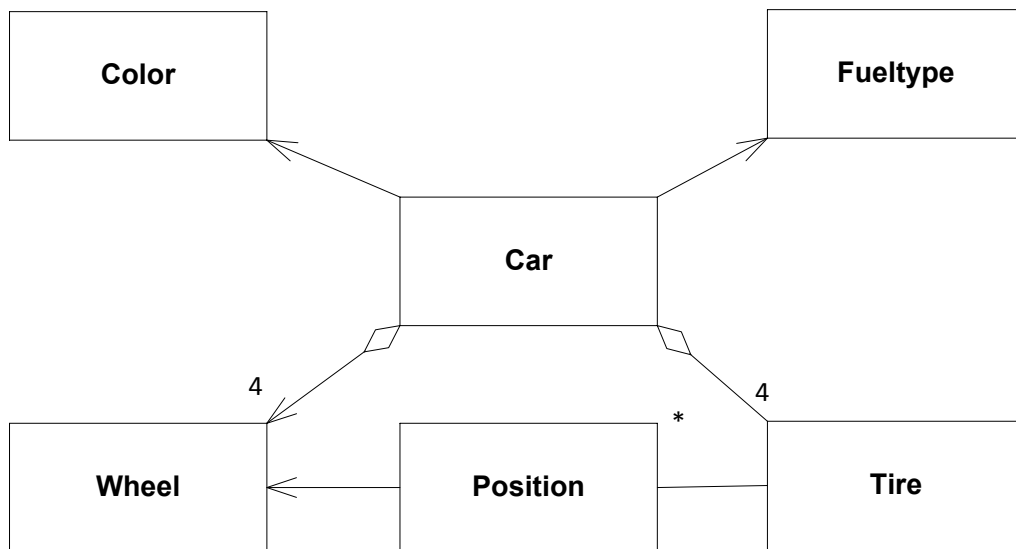
Two important concepts in domain-driven development are Entities and Value Objects. Identify all Entities and Value Objects in the following domain model and explain for **each element** why you identified it as Entity or Value Object.

Mark each Entity with an “E” and each Value Object with a “V”

Zwei wichtige Konzepte im Domain-driven Development sind Entitäten und Value Objects.

Identifizieren Sie im folgenden Domänenmodell alle Entitäten und Value Objects und begründen Sie für **jedes einzelne** Ihre Entscheidungen.

Markieren Sie jede Entity mit einem „E“ und jedes Value Object mit einem „V“



## Task 2 / Aufgabe 2 (8 Points / Punkte)

What does "Ousterhout's Dichotomy" say about the relation of scripting and system programming languages? Explain Ousterhout's claim in detail and answer the following questions:

- What does "gluing" mean?
- Why is "typing" un-/important?

Was besagt "Ousterhouts Dichotomie" über das Verhältnis von Skript- und Systemprogrammiersprachen? Erläutern Sie Ousterhouts Behauptung im Detail und beantworten Sie die folgenden Fragen:

- Was bedeutet „gluing“?
- Warum ist „typing“ un-/wichtig?

### **Task 3 / Aufgabe 3 (4 Points / Punkte)**

Define and explain the term “Ubiquitous Language”? What are the benefits/drawbacks?

Definieren und erklären Sie den Begriff „Ubiquitous Language“. Was sind die Vorteile/Nachteile?

## Task 4 / Aufgabe 4 (15 Points / Punkte)

Fill the Gaps 2-6 in the code on the next page by inserting the correct letters (as you can see for Gap 1) that are given to you in the table below. Be careful, some lines of code may not be used. The expected output of the script is printed below.

For your convenience, you will find the code fragments at the Appendix as well, which can be separated from this exam.

Füllen Sie die leeren Felder 2-6 im Code auf der nächsten Seite mit den richtigen Buchstaben (wie im Feld Nr. 1 zu sehen ist), welche in der Tabelle unten zu finden sind. Beachten Sie: einige Codezeilen werden nicht genutzt. Die erwartete Ausgabe des Scriptes finden Sie unten.

Zur leichteren Handhabung finden Sie die Code-Fragmente im Anhang, den Sie von dieser Prüfung abtrennen können.

### Expected output / Erwartete Ausgabe:

```
failed: property 'age' not in range '18..65'.
failed: property 'name' is null.
Validation complete.
```

### Code fragments

A	<code>o = \$("age"   "name").checkConstraints()</code>
B	<code>o.metaClass.constraints = checkConstraints(Closure constraints)</code>
C	<code>o.constraints.setDelegate(this)</code>
D	<code>o.constraints = new Closure()</code>
E	<code>void checkConstraints(String name, Object args) {</code>
F	<code>Object invokeMethod(String name, Object args) {</code>
G	<code>Object checkConstraints(Closure constraints) { args -&gt;</code>
H	<code>for (subject.constraints) { args[0] = it</code>
I	<code>it, delegate -&gt;</code>
J	<code>// validate each constraint</code>
K	<code>switch(val?.class) {</code>
L	<code>if (val != null) {</code>
M	<code>switch( it.value ) {</code>
N	<code>for ( args[0] ) {</code>
O	<code>switch( name ) {</code>
P	<code>if (it.key == "size" &amp;&amp; !it.value?.contains(val.length()))</code>
Q	<code>if (it.key == "size" &amp;&amp; it.value[0] &lt; val &lt; it.value[1])</code>
R	<code>if (it.key == "blank" &amp;&amp; it.value != null)</code>
S	<code>'\${it}'.</code>
T	<code>int age</code>
U	<code>'\${it.value[0]}..\${it.value[1]}'.</code>
V	<code>'\${it.value.inspect()}'.</code>
W	<code>'\${it.value[size]}'.</code>
X	<code>'\${it.value.call()}'.</code>
Y	<code>format "C:"</code>

```
class Customer {
```

1 T

```
String name
```

```
static constraints = {
```

```
age(size:18..65)
```

```
name(size:3..20, blank:false)
```

```
}
```

```
}
```

```
class Validator {
```

```
def subject
```

```
public void validate(def o) {
```

```
subject = o
```

2

```
o.constraints.call()
```

```
println "Validation complete."
```

```
}
```

3

```
def val = subject.getProperty(name)
```

```
args[0].each {
```

4

```
case null: if (it.key == "blank" && !val)
```

```
println "failed: property '${name}' is null."
```

```
break
```

```
case Integer :
```

```
if (it.key == "size" && !(it.value.contains(val)))
```

```
println "failed: property '${name}' not in range
```

5

```
break
```

```
case String:
```

6

```
println "failed: property '${name}' is too long or too short."
```

```
break
```

```
default: break;
```

```
}
```

```
}
```

```
}
```

```
}
```

```
customer = new Customer(age: 2, name:null)
```

```
validator = new Validator()
```

```
validator.validate(customer)
```

### **Task 5 / Aufgabe 5 (8 Points / Punkte)**

- a) Explain the term “model” in general!
  - b) Which kind of models are used in Software Development? Give at least 3 examples for models used in Software Development and describe how they are used.
  - c) What are the necessities for automatically processing models?
- 
- a) Erklären Sie den Begriff „Modell“ im Allgemeinen.
  - b) Welche Arten von Modelle werden in der Softwareentwicklung eingesetzt? Nennen und erklären Sie 3 Beispiele für Modelle, die in der Softwareentwicklung verwendet werden.
  - c) Was sind die Voraussetzungen für die automatische Verarbeitung eines Modells?

## **Task 6 / Aufgabe 6 (7 Points / Punkte)**

Describe a model-driven tool chain in software development and explain each of its items in a short sentence.

Beschreiben Sie eine „Model-driven Tool Chain“ in der Softwareentwicklung. Beschreiben Sie jedes Element der Tool Chain mit einem kurzen Satz.



### **Task 7 / Aufgabe 7 (6 Points / Punkte)**

Explain the difference between external and embedded DSLs! Describe their advantages and disadvantages. Which alternatives or variants are possible?

Erklären Sie den Unterschied zwischen externen und eingebetteten DSLs! Beschreiben Sie ihre Vor- und Nachteile. Welche Alternativen oder Varianten sind möglich?

## Appendix

### Task 4

#### Code fragments

A	<code>o = \$("age"   "name").checkConstraints()</code>
B	<code>o.metaClass.constraints = checkConstraints(Closure constraints)</code>
C	<code>o.constraints.setDelegate(this)</code>
D	<code>o.constraints = new Closure()</code>
E	<code>void checkConstraints(String name, Object args) {</code>
F	<code>Object invokeMethod(String name, Object args) {</code>
G	<code>Object checkConstraints(Closure constraints) { args -&gt;</code>
H	<code>for (subject.constraints) { args[0] = it</code>
I	<code>it, delegate -&gt;</code>
J	<code>// validate each constraint</code>
K	<code>switch(val?.class) {</code>
L	<code>if (val != null) {</code>
M	<code>switch( it.value ) {</code>
N	<code>for ( args[0] ) {</code>
O	<code>switch( name ) {</code>
P	<code>if (it.key == "size" &amp;&amp; !it.value?.contains(val.length()))</code>
Q	<code>if (it.key == "size" &amp;&amp; it.value[0] &lt; val &lt; it.value[1])</code>
R	<code>if (it.key == "blank" &amp;&amp; it.value != null)</code>
S	<code>'\${it}'.</code>
T	<code>int age</code>
U	<code>'\${it.value[0]}..\${it.value[1]}'.</code>
V	<code>'\${it.value.inspect()}'.</code>
W	<code>'\${it.value.size}'.</code>
X	<code>'\${it.value.call()}'.</code>
Y	<code>format "C:"</code>

#### Expected output / Erwartete Ausgabe:

failed: property 'age' not in range '18..65'.  
failed: property 'name' is null.  
Validation complete.