

### 1 Zork 2 – Game Projekt

Thema: Mit Klassen, Eigenschaften, Objekten, Listen, JavaDoc und JUnit arbeiten.

Partnerarbeit: maximal 2 Lernende

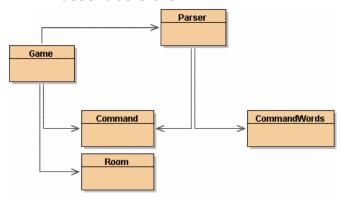
Zeit: ca. 12 Lektionen

Präsentation: Das eigene Spiel wird der Klasse in ca. 5 Minuten Vorträgen präsentiert.

### 1.1 Einleitung

Das Zork 2 ist eine einfache Basis für ein Abenteuer Spiel. Es implementiert Räume, welche man betreten und wieder verlassen kann.

#### 1.2 Klassenübersicht

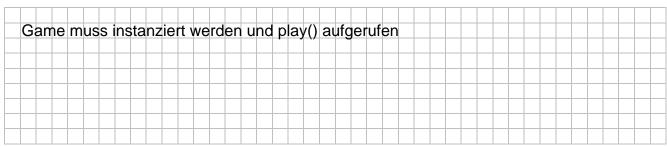


## 1.3 Hauptklasse für Zork

Eine Hauptklasse ist die Klasse, welche den Entry Point des Programmes bildet. Hier fängt also alles an. Sie besitzt eine statische Methode *main*. Dank dieser Methode weiss die intellij Umgebung wo das Programm mit der Ausführung beginnen soll.

Erstellen Sie die Klasse Zork2, welche uns als Hauptklasse dienen soll. Bei der Erstellung dieser Klasse kann die Erstellung der Methode *main* in eclipse direkt angegeben werden.

Welche Klasse muss in der main Methode instanziert werden, und welche Methode dieser Klasse muss aufgerufen werden, damit das Programm startet?





## 1.4 Erzeugung der Räume (aus dem Beispiel)

```
Room outside, lab, tavern, gblock, office;
...

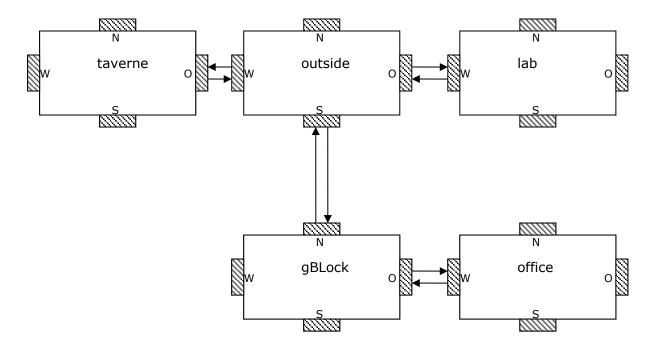
// create the rooms
outside = new Room("outside G block on Peninsula campus");
lab = new Room("a lecture theatre in A block");
tavern = new Room("the Seahorse Tavern (the campus pub)");
gblock = new Room("the G building");
office = new Room("the computing admin office");

// initialise room exits
outside.setExits(null, lab, gblock, tavern);
lab.setExits(null, null, null, outside);
tavern.setExits(null, outside, null, null);
gblock.setExits(outside, office, null, null);
office.setExits(null, null, null, gblock);

currentRoom = outside; // start game outside
```

#### 1.5 Raumsituation

Dies ist eine Skizze der Räume aus dem vorgegebenen Beispiel.





# 2 Aufgabe 1 – Spielidee - Dokumentation:

### 2.1 Eigene Räumlichkeiten definieren.

Definieren Sie eingene Räumlichkeiten eines Gebäudes oder einer Landschaft, ganz nach Ihrer Phantasie. Übertreiben Sie nicht, wählen Sie für den Anfang maximal 10 Räume.

Notizen										
Räume: entrance,	dining	hall, k	itchen	, base	ement,	bath,	libary,	study, f	orge, a	rmory, prison
Skizze siehe Doku										
Skizze Sierie Doko	<b>.</b>									

a) Erstellen Sie eine Skizze und Beschreibung in ihrer Dokumentation und zeigen Sie diese der Lehrperson. siehe Doku

#### 2.2 Gewonnen

Diesem Spiel fehlt eine massgebende Eigenschaft, welche wir jetzt zusätzlich implementieren wollen. Wir wollen implementieren, dass wir auch "gewinnen" können.

a) Beschreiben Sie in ihrer Dokumentation wann Sie gewonnen haben! siehe Doku

Implementieren Sie diese Funktion so, dass man beim Erreichen eines bestimmten Raumes gewinnt. Erweitern Sie die Klasse Raum mit einer Eigenschaft, über welche herausgefunden werden kann ob es sich bei einem bestimmten Raum um den "Gewinner Raum" handelt. Definieren Sie einen solchen Raum (Objekt), bei welchem man gewinnt, sobald man den Raum betritt.

gemacht, siehe Doku



Tip:

Untersuchen Sie wie die Ausgabe der Raumbezeichnung und die möglichen Ausgänge des aktuellen Raumes, programmiert ist. Konkret die Methode *longDescription()* der Klasse Room. Versuchen Sie es zu verstehen und analog zu programmieren.

#### 2.3 Befehl "back"

Implementieren Sie den Befehl *back*. Dieser Befehl bringt uns zu diesem Raum, von dem wir gekommen sind.

a) Beschreiben Sie in ihrer Dokumentation wie Sie es realisieren möchten! siehe Doku

#### 2.4 Gegenstand im Raum

Im Raum kann ein Gegenstand sein. Erzeugen Sie eine Klasse *Item* mit den Eigenschaften und den dazugehörenden Methoden. Erweitern Sie die Klasse *Room* so, dass man dem Objekt einen Gegenstand hinzufügen kann.

- a) Beschreiben Sie in ihrer Dokumenation wie Sie vorgehen! gemacht, siehe Doku
- b) Wir müssen auch nach dem Gegenstand im Raum "fragen" können. Dokumentieren Sie wie es realisieren. gemacht, siehe Doku

#### 2.5 Mehrere Gegenstände

Im Raum können mehrere Gegenstände sein. Erweitern Sie die Klasse *Room* so, dass es eine *ArrayList* mit Gegenständen besitzt. Die Ausgabe soll auch entsprechend abgeändert werden.

Notizen						
Nicht mehrere,	, sonaer v	erschieden	e una man mi	uss eine Frag	en beantworte	∍n, um sie
aufzunehmen,	mehr sieł	ne Doku				





#### 2.6 Gewicht

Ein Gegenstand hat ein Gewicht. Erweitern Sie die Klasse Item mit einem Attribut weight.

a) Erweitern sie die Klasse und beschreiben Sie wie sie vorgehen! Kein Gewicht, dafür beschreibung und verschiedene

# 2.7 Befehl "map"

Typen an Gegenständen, siehe Doku.

Ergänzen Sie die Klasse Game mit einer Methode, welche auf Befehl alle Räume mit allen Gegenständen ausgibt. Erweitern Sie dazu die Klasse Game mit einer Arraylist, welche die Räume speichert.

a) Aus d	der Ausgab	e soll auc	h ersichtlich	h sein, in welche	m Raum man sich 🤉	gerade befindet.									
Notizen															
gemacr	it, wird ein	e iviap an	igezeigt, m	ienr siene Doku											



# 3 Schwierigere Aufgaben

# 3.1 Der Spieler kann ein Gegenstand bei sich tragen

a) Dokumentieren Sie und realisieren Sie diese Anforderung!

N	otiz	zen	1																																		
															_				١.																		
	ge	ma	ach	٦t,	Ιtέ	em	າຣ	W	erc	ıeı	וו ח	nε	∍ın	en	n I	≺u	Ck	sa	ICk	cg	eti	raç	gei	n,	me	eh	r s	ıel	ne	D	ok	u					

## 3.2 Der Spieler kann mehrere Gegenstände tragen (mit einem Maximalgewicht)

b) Dokumentieren Sie und realisieren Sie diese Anforderung!

	٧,				٠	. ~		٠.	٠.	. –		۵.	٠		۰			. ~		<b>ч</b> .		٠,		٠.	٠.٠		9	•														
Ν	ot	zei	า																																							
			Т	Т																																						
	C	e	'na	ас	ht	, k	œi	n	M	ax	im	ale	ae	wi	ch'	t, (	da	es	s k	ei	ne	S	ch	we	re	n	Ge	eqe	en	sta	ind	de	si	nd	l, r	ne	hr	si	eh	е	Dς	oku
		+	+	+	+	+						<u> </u>																							_						$\Box$	
		Т	Т	Т																																						

## 3.3 Der Spieler kann Gegenstände auch wieder ablegen

c) Dokumentieren Sie und realisieren Sie diese Anforderung!

د, د	Oitu			C.	C	_		uii	u		4115	,,,	C.	. –	,,,	uı	CJ.	٠,		O.	uc	· u	. 19	•												
Notizen																																				
	Nicht gemacht, dafür benötigt man sie um Räume zu öffnen, mehr siehe Doku																																			
Nich	t ge	ma	ich	ıt,	da	ıfü	r k	eı	ηö	tig	t n	na	n s	sie	u	m	R	äu	me	e z	:u	öff	ne	n,	n	nel	٦r	sie	eh	e I	Dο	ku				
	Ĭ									Ū																										

Notizen																										
Da die	Gege	enstä	inde	kleir	sin	d, ha	ıt e	s ç	jeni	ıg	Pla	atz	im	Rι	ıck	sa	ck,	me	ehr	sie	ehe	D	ok	u		
																						-				-
																						-				-
												_		-								+			+	+