

## 1 Zork 2 – Game Projekt

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

Partnerarbeit: maximal 2 Lernende

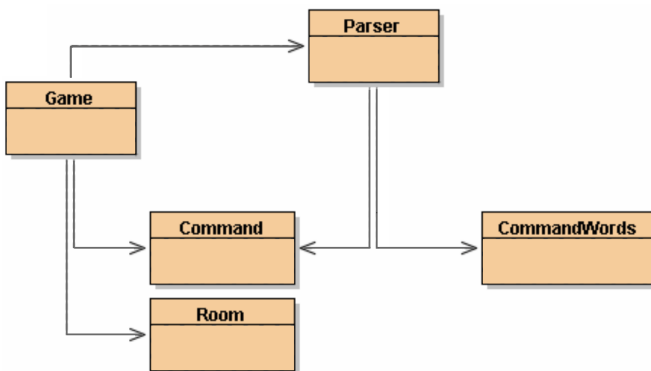
Zeit: ca. 12 Lektionen

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.

```

public class Zork2Game {

    public static void main(String[] args) {
        ...
    }
}
  
```

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 instanziiert werden, und welche Methode dieser Klasse muss aufgerufen werden, damit das Programm startet?

Game muss instanziiert werden und play() aufgerufen

#### 1.4 Erzeugung der Räume (aus dem Beispiel)

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

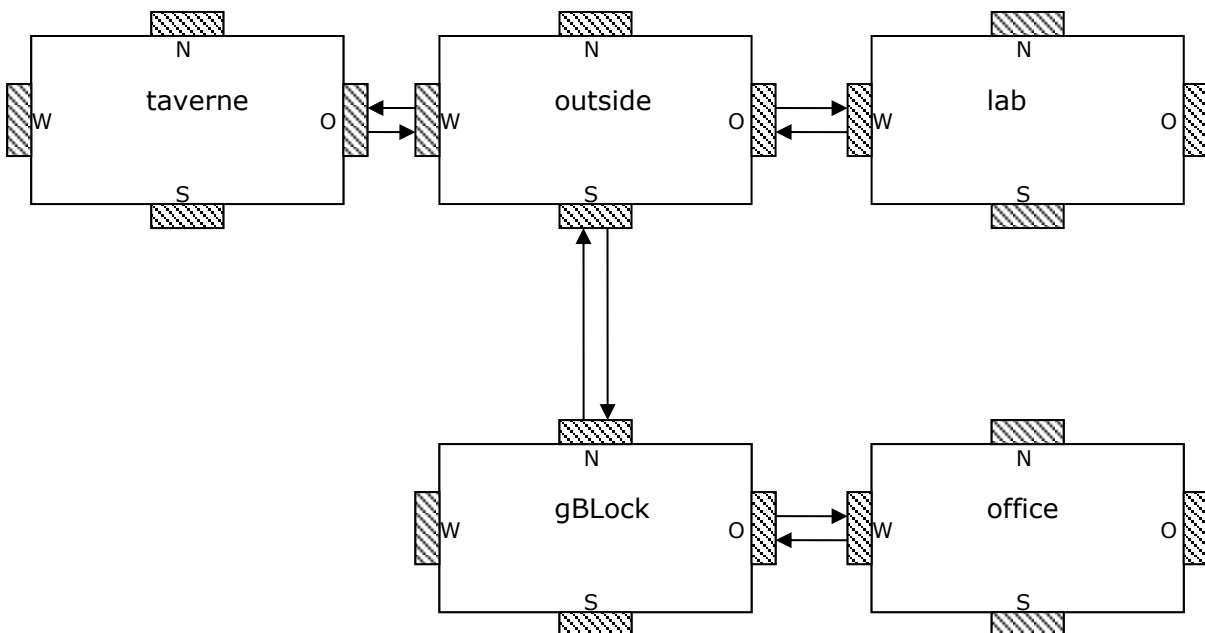
// create the rooms
outside = new Room("outside G block on Peninsula campus");
lab = new Room("a lecture theatre in A block");
tavern = new Room("the Seahorse Tavern (the campus pub)");
gblock = new Room("the G building");
office = new Room("the computing admin office");

// 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.







### 3 Schwierigere Aufgaben

### 3.1 Der Spieler kann ein Gegenstand bei sich tragen

a) Dokumentieren Sie und realisieren Sie diese Anforderung!

[illegible]

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

b) Dokumentieren Sie und realisieren Sie diese Anforderung!

[illegible]

### 3.3 Der Spieler kann Gegenstände auch wieder ablegen

c) Dokumentieren Sie und realisieren Sie diese Anforderung!

[illegible][illegible]