

Übung 4 [24]: Textadventure (2)

In dieser Übung sollen einige Zusätze programmiert werden, um eine bessere Textausgabe und einen "verständigeren" Parser zu erreichen. Sie müssen dazu wieder das Paket vom Server laden, da einige Änderungen und Erweiterungen im Programm vorgenommen wurden (Alles bis auf Ihren Source-Code zu Raum und den Gegenständen ist zur Sicherheit zu ersetzen). Beachten Sie, daß auch in Ihren Klassen Änderungen notwendig sein werden: Es sind neue Methoden zu programmieren oder alte mit den neuen Möglichkeiten abzuändern.

Erweitern Sie Ihr Textadventure um folgende Punkte:

- Abenteuer.Daten.Orte.Raum: In der Methode `betreten(...)` ist der Text so zu verändern, daß er für beliebige Personen richtig ist. Dazu ist die Anrede und die Verb-Form richtigzustellen. Test: Sowohl Sie als auch der Bettler (befehlen Sie ihm!) soll versuchen, einen dunklen Raum zu betreten.
- Abenteuer.Daten.Objekte.Gegenstand: Der Klasse sind einige Methoden hinzuzufügen, um den richtigen Artikel für den Gegenstand mitspeichern und ausgeben zu können. Erstellen Sie dazu eine statische Liste der verschiedenen Artikel und implementieren Sie die Methode `artikelName` nach der Javadoc. Test: Allgemein ausprobieren. Diese Methode wird vielfach verwendet (z. B. aufnehmen/ablegen von Gegenständen).
- Abenteuer.Programm.Parser.Parser: Implementieren Sie in der Klasse Parser die Methoden `nextToken`, `hasMoreTokens` sowie den Konstruktor. Die Eingabe wird von `System.in` gelesen. Sie dürfen die Klasse `StringTokenizer` dazu nicht verwenden. Bedenken Sie, daß sie beim Einlesen immer ein Wort im voraus lesen müssen, um feststellen zu können, ob das Zeilenende schon erreicht wurde (z. B. wenn das letzte Wort in der `killListe` enthalten ist)!
- Versuchen Sie herauszufinden, wo die Anpassung des Textes an die handelnde Person, den Gegenstand, ... noch nicht richtig funktioniert. Schreiben Sie dazu ein paar kurze Notizen auf einem Extra-Zettel (auch Handschrift, sofern lesbar).

Weitere Informationen über die Klassen und die Änderungen finden Sie in den Javadoc-Dateien in HTML-Form, die ebenfalls in der ZIP-Datei enthalten sind.

Hinweis: Die Klasse Parser ist teilweise vorgegeben. Sie müssen nur die fehlenden Methoden implementieren (Sowie etwaige private Hilfsmethoden).