**PR1 – Formular für Lesenotizen**

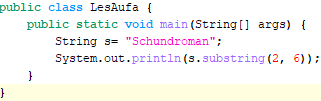
**WS2020/21**

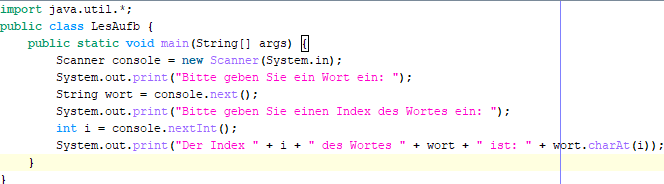
|  |  |  |  |
| --- | --- | --- | --- |
| Nachname  Abdel Kader | Vorname  Schehat | Matrikelnummer  1630110 | Abgabedatum:  28.10.2020 |

**Stringobjekte und interaktive Programme, Pakete und Classpath (L3)**

**Lernzielfragen:**

1. Schreiben Sie einen Ausdruck, der aus dem folgenden String die Zeichenkette "hund“ isoliert:

String s= "Schundroman";

1. Schreiben Sie ein Programm, das vom Benutzer ein Wort und eine Zahl i einliest. Das Programm gibt dann das i-te Zeichen des Wortes aus.
2. In einem Java-Programm können mehrere Klassen gleichen Namens existieren. Wie unterscheidet man die Klassen im Programmtext? Wie unterscheidet man sie auf der Festplatte? Welche Schlüsselwörter benötigt man zur Handhabung mehrerer gleichnamiger Klassen?

* Die Klassen im Programmtext werden durch Packages unterschieden, in denen sich die Klassen befinden. Anschließend wird mit dem CLASSPATH das Programm Klasse für Klasse kompiliert und am Ende ausgeführt

**Notizen:**

**Stringobjekte**

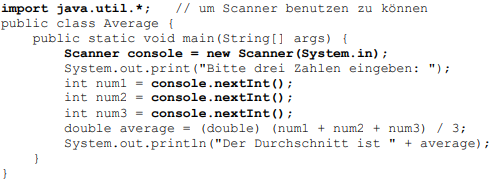
* Strings können wir primitive Datentypen deklariert werden: String <name> = <text>;
* Einzelne Zeichen vom String werden als char repräsentiert mit ‘<char>‘ statt “<String>“
* Methoden verändern nicht den String => müsste eventuell neu zugewiesen werden

|  |  |
| --- | --- |
| **Methodenname** | **Beschreibung** |
| charAt(index) | Zeichen an der gegebenen Indexstelle |
| indexOf(str) | Index, an dem der als Parameter gegebene String str in dem String-Objekt beginnt (-1, wenn er nicht vorkommt) |
| length() | Anzahl der Zeichen im String-Objekt |
| substring(index1, index2) | Die Zeichen von einschließlich index1 bis ausschließlich index2 |
| toLowerCase() | Ein neuer String in Kleinbuchstaben |
| toUpperCase() | Ein neuer String in Großbuchstaben |

**Interaktivität & Hilfsklasse Scanner**

|  |  |
| --- | --- |
| **System.out** | **System.in** |
| Für Ausgaben auf der Console | Für Eingaben auf der Console |
| Hat Methoden println und print | Hat Methoden für das Einlesen von Bytes |
| Kann sinnvoll direkt verwendet werden | Wird in der Regel nicht direkt verwendet |

|  |  |
| --- | --- |
| **Methode** | **Beschreibung** |
| nextInt() | Liest und gibt die Benutzereingabe als int zurück |
| nextDouble() | Liest und gibt die Benutzereingabe als double zurück. Es wird 1,4 erwartet und nicht 1.4 wegen Lokalisierung |
| next() | Liest und gibt die Benutzereingabe als String zurück |
| nextLine() | Liest und gibt die nächste Eingabezeile als String zurück |



* Scanner als Parameter wird so angegeben: public static void methode1(Scanner console)
* Mit return value muss man void mit int ersetzen

**Tokens**

Scanner-Objekt liest Eingaben in Blöcken zwischen Leerraumstellen, diese Blöcke werden Token genannt. Bsp.: 23 John Smith 42.0 "Hello world" => sind 6 Tokens

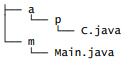
**Pakete**

Jede Java Klasse (java & class-Dateien) ist in einem Paket (package), dieser entspricht der Verzeichnisstruktur im Dateisystem. => 1. Zeile (bzw. nach Kommentar): packaga <paketname>;

**Classpath**

Umgebungsvariable legt Compiler und Interpreter fest, wo im Dateisystem nach Paketen und class-Dateien gesucht wird. CLASSPATH kann mit Paketnamen verknüpft werden (sogar in einem anderen Verzeichnis)

* Bsp.: Verzeichnis /home/sch/prog1/bsp liegt Beispiel.java & .class mit package = prog1/bsp
* Man kann in der Konsole CLASSPATH = /home/sch und danach java /prog1/bsp die Datei ausführen

**Modulepath**

Konzept zur Strukturierung eines Programms, wichtig bei vielen Klasse

