|  |
| --- |
|  |
| Managementsystem für Ferienwohnungen |
|  |
|  |
| **Jony Nchamadi, Nico Fischer, Lukas Wüstenhagen, Daniel Schönke** |
| **24.05.2017** |



|  |
| --- |
|  |

Inhalt

[1.Programmbeschreibung 1](#_Toc483393584)

[2. Quelltextauschnitte der main-Methode 2](#_Toc483393585)

[2.1 Quelltextauschnitte des Moduls Übersicht der Ferienwohnung 3](#_Toc483393586)

[2.2 Quelltextauschnitte des Moduls „ Kundendaten“ 4](#_Toc483393587)

[2.4 Quelltextauschnitt des Moduls BuchungUebersicht 5](#_Toc483393588)

[2.3 Quelltextauschnitte des Moduls Buchung von Ferienwohnungen 5](#_Toc483393589)

[2.5 Auschnitt des Moduls „Update aller Daten" 6](#_Toc483393590)

[2.6 Auschnitt des Moduls „Preisbearbeitung für Ferienwohnung" 8](#_Toc483393591)

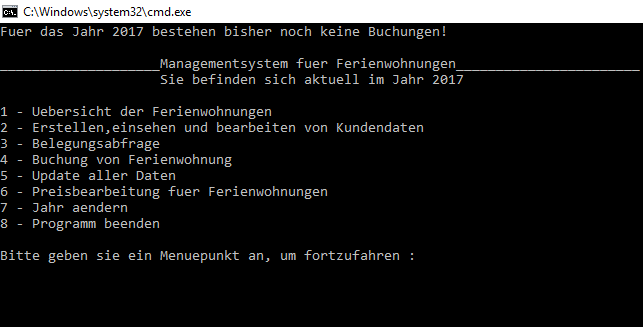
[2.7 Auschnitt des Moduls Jahr ändern (Teilausschnitt der main-Methode) 8](#_Toc483393592)

[3. Befehlsbeschreibung 9](#_Toc483393593)

# 

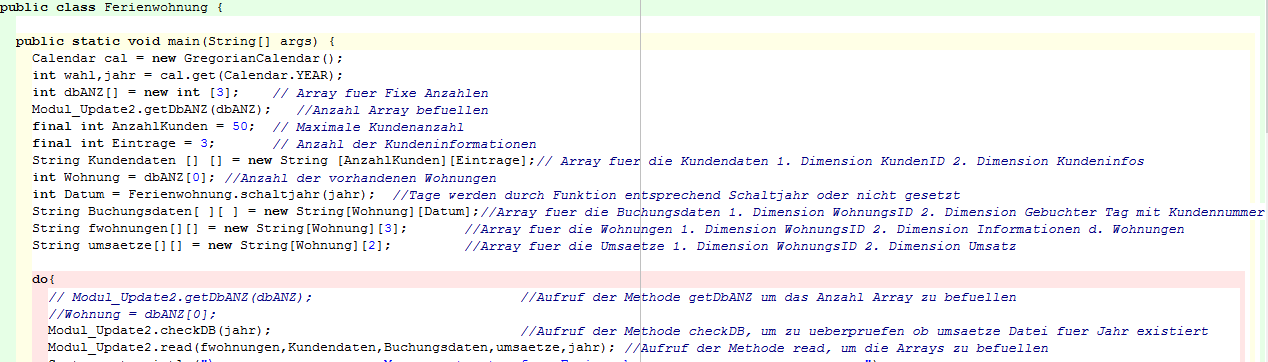
# 1.Programmbeschreibung

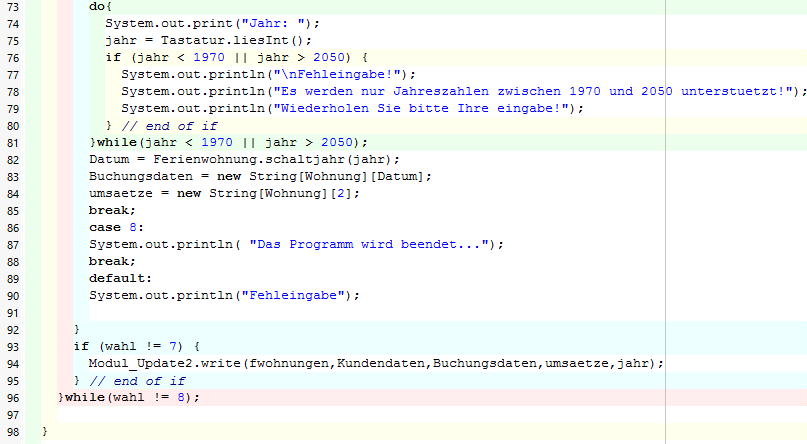
**Mit dem Java-Programm „Ferienwonhung.java“ und den spezifischen Modulen, können wir Ferienwohnungen verwalten. Im folgenden Bildausschnitt ist die Übersicht der Möglichkeitenzu sehen. Es ist mit den Java-Programmen möglich eine Übersicht aller Ferienwohnungen auszugeben, Kundendaten zu erstellen, einzusehen und zu bearbeiten, Ferienwohnungen zu buchen, eine Preisabrechnung zu machen, alle Daten upzudaten, Preisbearbeitungen von Ferienwohnungen durchzuführen und das Programm zu beenden. Jeder einzelne Schritt wird in den Modulen realisiert und verwirklicht und per cmd.exe ausgegeben.**

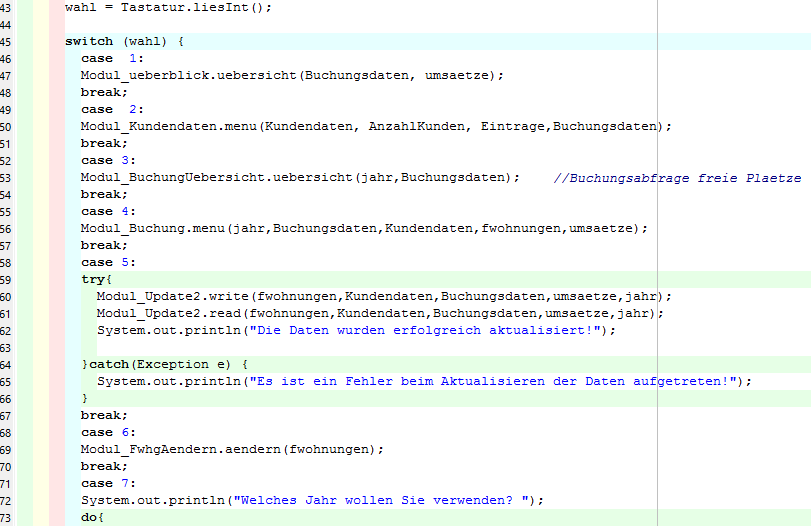


# 

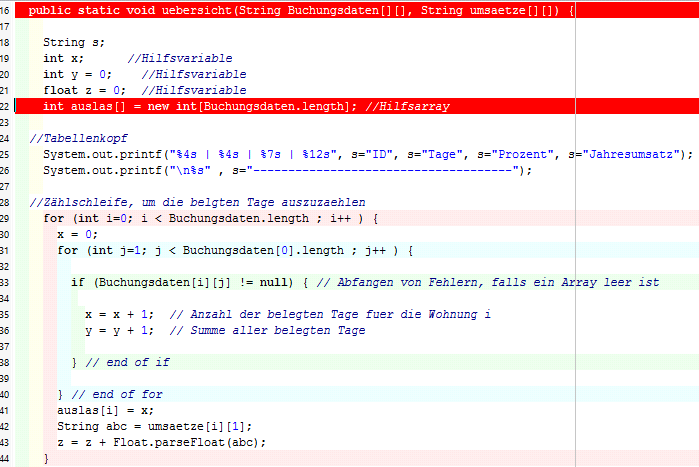
# 2. Quelltextauschnitte der main-Methode



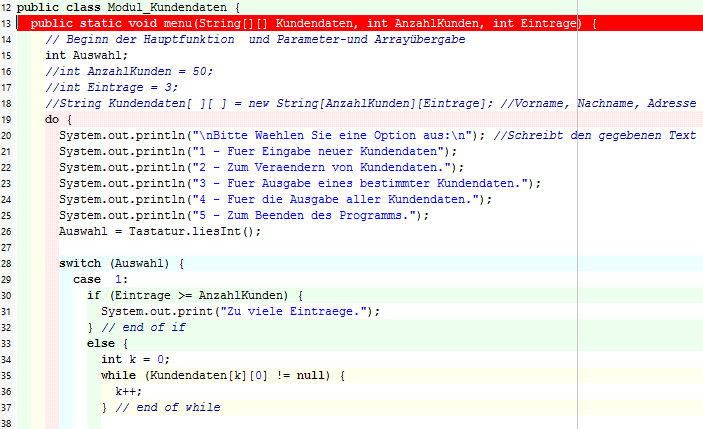


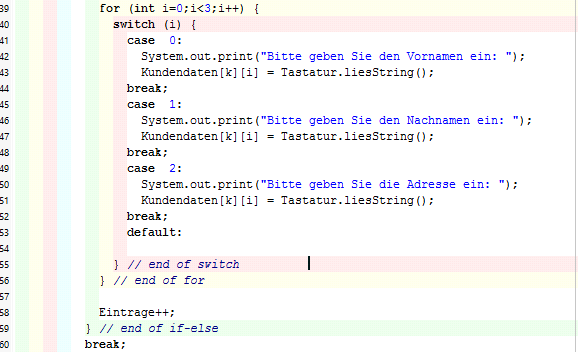


# 2.1 Quelltextauschnitte des Moduls Übersicht der Ferienwohnung

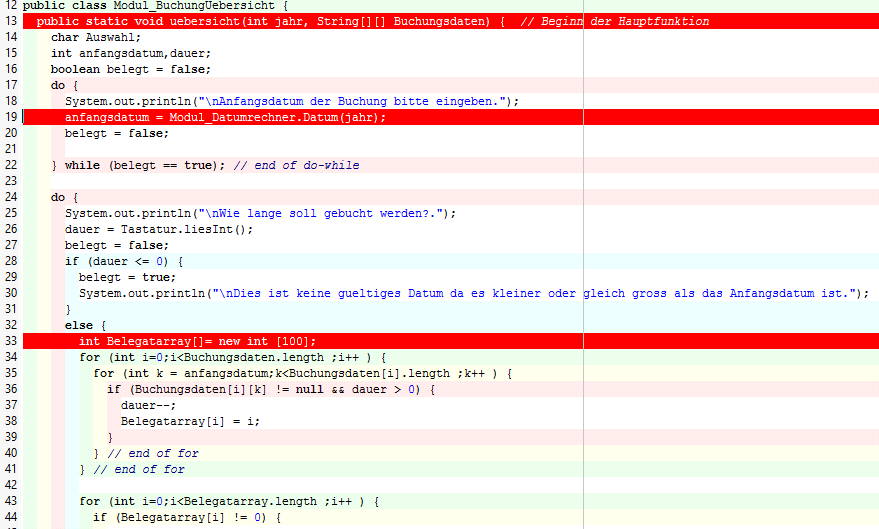


# 2.2 Quelltextauschnitte des Moduls „ Kundendaten“

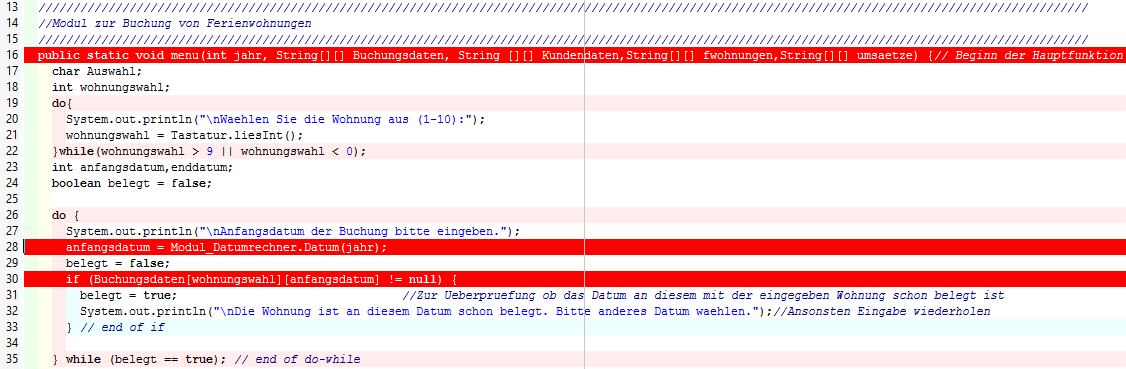


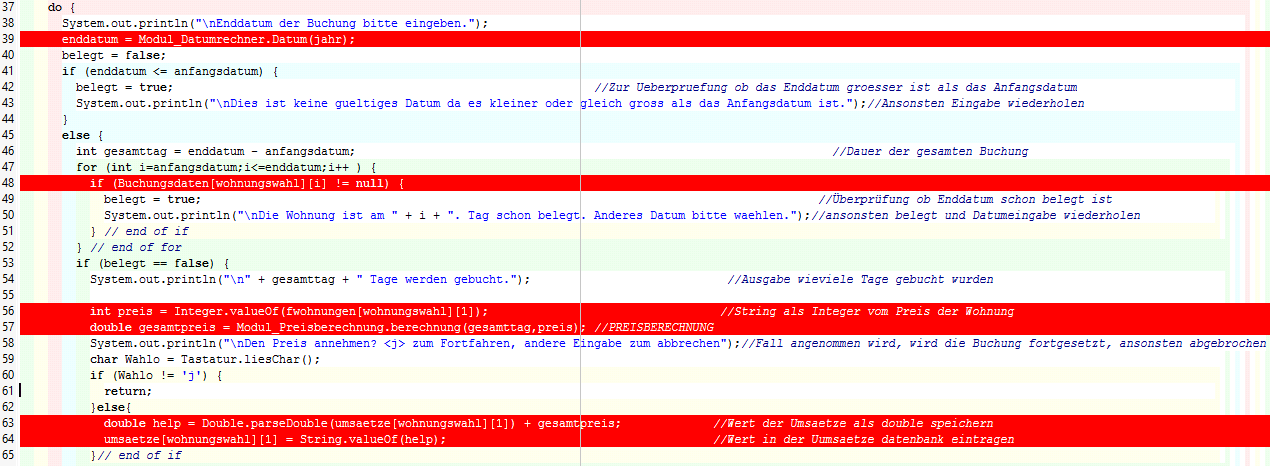


# 2.4 Quelltextauschnitt des Moduls BuchungUebersicht

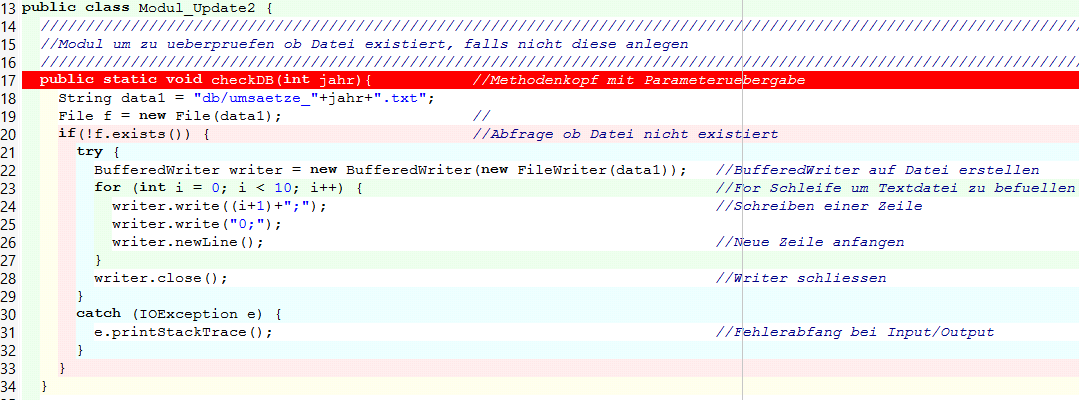


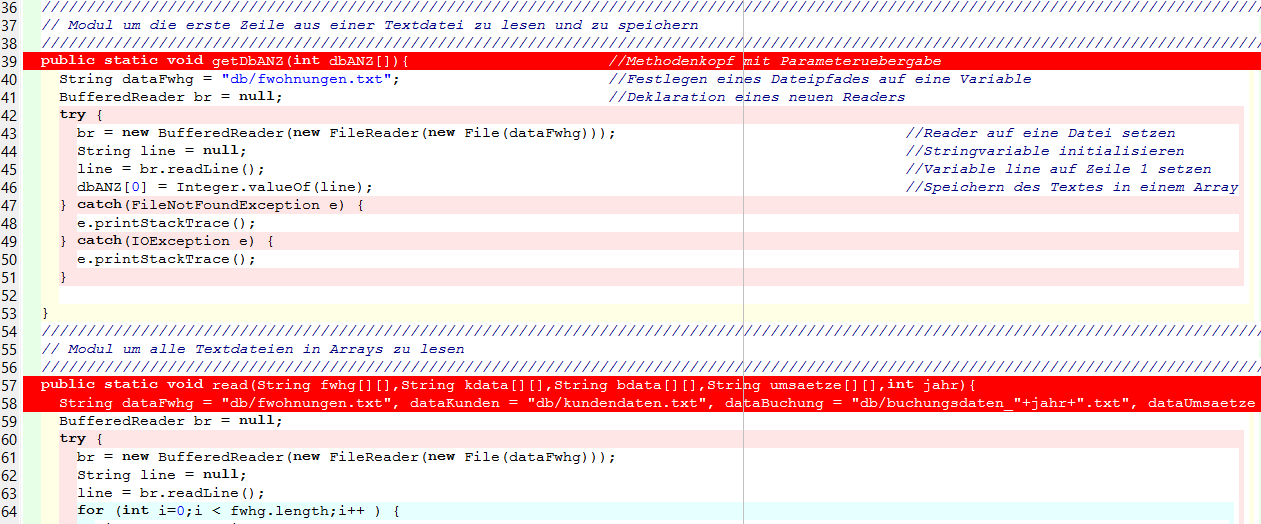
# 2.3 Quelltextauschnitte des Moduls Buchung von Ferienwohnungen

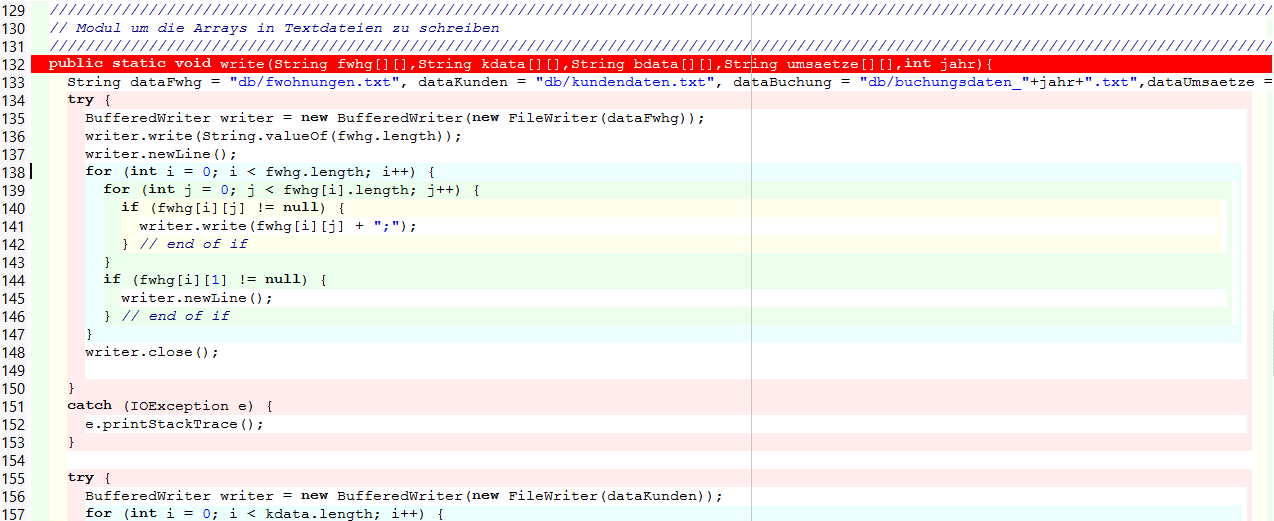




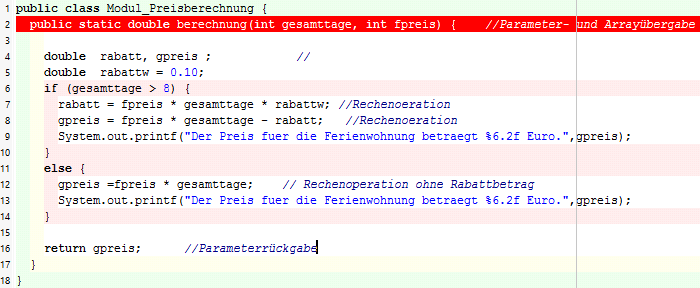
# 2.5 Auschnitt des Moduls „Update aller Daten"



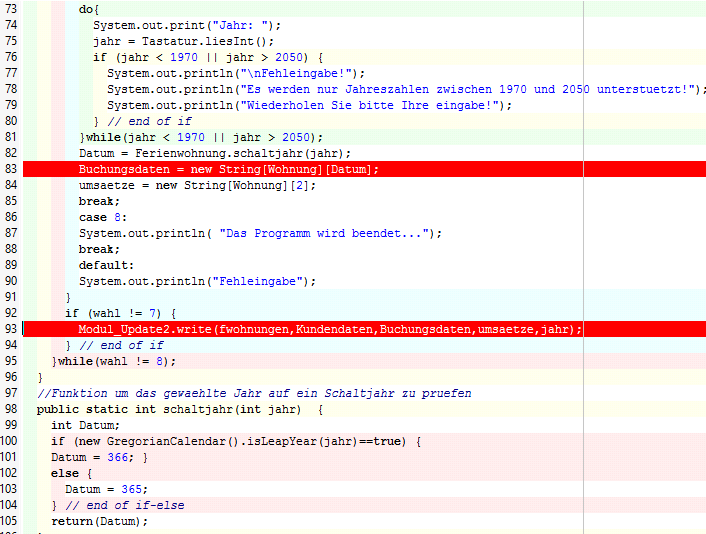




# 2.6 Auschnitt des Moduls „Preisbearbeitung für Ferienwohnung"



# 2.7 Auschnitt des Moduls Jahr ändern (Teilausschnitt der main-Methode)



# 3. Befehlsbeschreibung

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Nr. | **Thema** | **Code allgemein / Methode** | | | **Beispiel** |
| 1 | Bezeichner | Variable, Klasse, Methode, Pakete | | | int Zahl, class Liferant |
| 2 | Kommentare | //einzeilige Kommentar /\*mehrzeiliger Kommentar \*/ | | | Int i = 0 // Variablendeklaration int a,b,c,d,e,f /\* Variablendeklarationen \*/ |
| 3 | Arithmetische Operationen | (+,-,\*,/,++,--)  (Addition, Subtraktion Multiplikation, Divison, Inkrement, Dekrement) | | | Summe = Summe + Betrag  I++ |
| 4 | Logische Operationen | && (und), || (oder), ! (nicht) | | | Switch (1 && 2) |
| 5 | Vergleichoperationen für Zhalen | <= (kleiner oder gleich), >= (größer oder gleich), == (gleich), != (ungleich) | | | If (Zahl1 == Zahl2).. |
| 6 | Variable einfache Datentypen | Int var; double var1,var2,var3; short, flot, byte, long, char, | | | int Betrag = 100; float Betrag = 3.2134 |
| **7** | **Array** | Datentyp Arrayname [] = new datentyp [AnzE] | | | Int Zahlen [] = new int [5] |
| **Kontrollstrukturen** | | | | | |
| 7 | Folge “lineare Struktur” | Anweisung | | | public void action Zahlen (…) |
| 8 | Parameterübergabe | Anweisung | | | public static void menu (int jahr, String [] [] Buchungsdaten) |
|  | Modularverweisung | Anweisung | | | Case1:  Modul\_Kundendaten.menu (Kundendaten, AnzahlKunden, Einträge) |
| **Wichtige Ereignisse (Funktion)** | | | | | |
| 9 | **Zweck des Ereignis** | | | **Ereignis** | |
|  | Standardereignis, wird ausgeführt wenn die Entertaste gedrückt wird | | | public static void main (String [] args) | |
| **Wichtige Methoden** | | | | | |
|  | **Zweck der Methode** | | **Methode** | | **Beispiel** |
|  | Daten an einer bestimmte Stelle lesen | | Tastatur.liesInt(); | | system.out.print („Geben sie eine Zahl ein!“); zahl 1 = Tastatur.liesInt(); |
|  | Text abfragen | | GetText(); | | System.out.print („Wie heißen Sie?“); name = getText(); |
|  | Ausgabe auf der Konsole | | system.out.println | | system.out.println (“1“+“2 = 3); |
|  | Den Wert einer Variable um den Wert 1 erhöhen | | Variable ++; | | for ( i = 0, i < 10, i++) |
|  | Zusammenfügen von zwei Strings | | “+“- Zeichen | | system.out.println (“1“+“2 = 3); |
|  | Reservieren von Platz (Zeichenplätze) | | %5,2f (5Zeichen und 2 Kommazahlen) | | system.out.printf (“Der Geldbetrag ist %5,2f hoch“, zahl); |
|  | Zeilenumbruch | | “\n“ | | System.out.println (“\n Hallo“); |
|  | Fallauswahl = Überprüfung eines Wertes und entsprechende Ausführung | | Switch () Case:(Fall), break (Pause/Stop) default (Standardausgabe) | | Switch (wahl) { case1: Modul\_Überblick.übersicht (Buchungsdaten,umsätze); break; default: system.out.println (“Fehler“); |
|  | Verzweigung = Überprüft ein bestimmte Bedingung -> Anweisung | | If (Bedingung-> Anweisung) else(Anweisung) | | System.out.println (“Zahl eingeben“); x = Tastatur.liesInt(); if (x <10) { system.out.println (“Die Zahl ist kleiner als 10“);} else { system.out.println (“Die Zahl ist größer als 0“);} |
|  | Zähler-gesteuerte Schleife = Ausgabe von Wiederholungen nach bestimmten Bedingen | | for ( Bedingungen) Anweisungen | | int i; for(i=0; i<5; i++) { system.out.printf (“ Zahl %d\n", i+1)} |
|  | Kofgesteuerte Schleife = Ausgabe von Wiederholungen nach bestimmten Bedingen | | Do (Anweisung) while (Bedingung) | | int alter;  do {  printf("\nBitte geben sie ihr Alter ein: ");  scanf("%d", &alter);  } while(alter < 5 || alter > 100);  printf("Danke.\n"); |
|  | Methode die benutzt wird um mögliche Fehler abzufangen und darauf zu reagieren | | Try { Code der gesichert läuft} catch Exception Aushnamen{} | | Try Modul\_Update2.write(Kundendaten); system.out.println(„Aktualiserung erfolgreich“) } catch (Exception e ) { system.out.println (“Aktualisierung fehlgeschlagen“) } |
|  | Parsing =Umwandlung eines Datentyps in einen anderen | | Datentyp Datenname = Datenwert; Datentyp Datename.parseDatentyp (Datenname); | | String tenString = "10“; Integer result = 20 + tenString; Integer result = 20 + Integer.parseInt(tenString); |