

Entdecken Sie Java

Einleitung

Entdecken Sie Java ist ein Übungsbuch in 2 Teilen, das in die Grundlagen der Programmierung mit Java einführt.

Das Übungsbuch wurde nach neurodidaktischen Prinzipien entwickelt um gehirngerechtes Lernen zu unterstützen. Der Schwerpunkt liegt auf der Methode des Entdeckenden Lernens, das einen wichtigen Lernmechanismus des Gehirns nutzt: die automatische Mustererkennung. Sie ermöglicht das selbständige Entdecken und Erarbeiten wichtiger Strukturen, Regeln etc. anhand von Beispielen und Musterlösungen.

Die Aufgaben des Übungsbuchs wurden bereits in Einführungslehrveranstaltungen an der Alpen-Adria-Universität Klagenfurt sowie an verschiedenen Schulen eingesetzt und evaluiert. Beide Teile beinhalten folgende Aufgabentypen mit Musterlösungen:

A Aufgaben zum entdeckenden Lernen

- Lesecken
- Schritt-für-Schritt-Aufgaben

B Miniübungen für kompetenzorientiertes Lernen incl. Musterlösungen, die ebenfalls entdeckendes Lernen fördern

C Programmieraufgaben

D Sprachlernprogramm, ein komplexes Programm, das kapitelweise erweitert wird.

Zusätzlich gibt es eine Website mit Materialien für die Aufgaben sowie die gesammelten Lösungen zu allen Übungen finden Sie auf der Website zum Buch auf <http://entdecken-sie-java.jimdo.com>.

Teil 1 von *Entdecken Sie Java* beinhaltet Grundlagen der Programmierung und ist sowohl in Bachelorkursen zur Einführung in die Programmierung als auch bereits in Schulen der Sekundarstufe II, AHS (Gymnasium), BHS (Berufsbildende Höhere Schulen) und BS (Berufsschulen) einsetzbar. Er beinhaltet u.a. Aufgaben zu Datentypen, ein- und mehrdimensionalen Arrays sowie Strings.

Teil 2 führt in die Objektorientierung ein und ist hauptsächlich zur Einführung in die Programmierung an Universitäten gedacht, kann aber auch bereits an höheren Schulen mit Informatik-Schwerpunkt eingesetzt werden.

Viel Spaß beim Üben wünschen

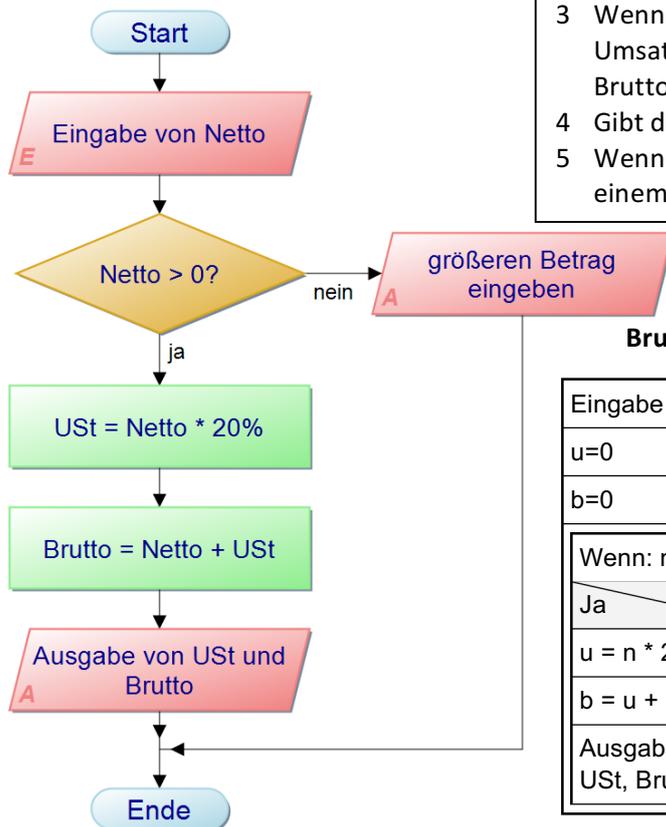
Barbara Sabitzer
Sandra Spieß-Knafl
Stefan Pasterk

1 Algorithmen und Datentypen

A Entdecken

A 1 Lesecke 1: Darstellung von Algorithmen

Bruttobetrag - Flussdiagramm



Bruttobetrag – Algorithmus in natürlicher Sprache

- 1 Frage nach dem Nettobetrag.
- 2 Gib einen Nettobetrag ein.
- 3 Wenn der Nettobetrag größer 0 ist, berechne 20% Umsatzsteuer vom Nettobetrag und den Bruttobetrag (Summe von Netto und USt).
- 4 Gib die USt und den Bruttopreis am Bildschirm aus.
- 5 Wenn der Nettobetrag nicht größer 0 ist, frage nach einem größeren Betrag.

Bruttobetrag - Nassi-Shneiderman-Diagramm

Eingabe n	
u=0	
b=0	
Wenn: n > 0	
Ja	Nein
u = n * 20%	
b = u + n	
Ausgabe: USt, Brutto	Ausgabe: Größeren Betrag eingeben!

Bruttobetrag – Java Code

```

public class Brutto2 {
    static Scanner in = new Scanner(System.in);

    public static void main(String[] args) {
        System.out.println("Bitte geben Sie den Nettobetrag ein: ");
        double n = in.nextDouble();
        double u=0, b=0;
        if (n <= 0)
            System.out.println("Geben Sie einen größeren Betrag ein!");
        else {
            u = n*20/100;
            b = n + u;
            System.out.println("Die USt beträgt: " + u);
            System.out.println("Der Bruttopreis beträgt: " + b);
        }
    }
}
  
```

Bitte geben Sie den Nettobetrag ein: 10
Die USt beträgt: 2.0
Der Bruttopreis beträgt: 12.0

Bitte geben Sie den Nettobetrag ein: 0
Geben Sie einen größeren Betrag ein!

Aufgabe 1. Lesen und Verstehen

Die obige Lesecke zeigt anhand des einfachen Beispiels der Berechnung von Umsatzsteuer und Bruttobetrag den Weg von der Problemstellung zum Programm. Lesen Sie erst den Algorithmus in natürlicher Sprache und ordnen Sie die einzelnen Schritte den jeweiligen Entsprechungen im Flussdiagramm, im Nassi-Shneidermann-Diagramm und im Programmcode zu (z.B. durch Nummerieren oder farbliches Hervorheben).

A 2 Lesecke 2: Programmcode verstehen

```
import java.util.Scanner;
public class Noten {
    public static void main(String[] args) {
        System.out.println("Geben Sie Ihre Note ein: ");
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        switch (n) {
            case 5:
                System.out.println("Leider nicht bestanden");
                break;
            case 4:
                System.out.println("Genügend");
                break;
            case 3:
                System.out.println("Befriedigend");
                break;
            case 2:
                System.out.println("Gut");
                break;
            case 1:
                System.out.println("Sehr gut");
                break;
        }
    }
}
```

```
import java.util.Scanner;
public class Eingabe {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Bitte Zahl a eingeben:");
        int a = sc.nextInt();
        System.out.println("Bitte Zahl b eingeben:");
        int b = sc.nextInt();
        int c = a + b;
        System.out.println("Die Summe ist: " + c);
    }
}
```

Aufgabe 2. Programmcode verstehen

Was leisten die beiden Programme der Lesecke 2? Formulieren Sie passende Algorithmen in natürlicher Sprache oder Pseudocode und stellen Sie sie anschließend in Form von Fluss- und Nassi-Shneiderman-Diagrammen dar.

Aufgabe 3. Merken

Gestalten Sie einen „Schwindelzettel“ für ProgrammieranfängerInnen, der die in den obigen Beispielen ersichtlichen Gesetzmäßigkeiten und Regeln in übersichtlicher Form erfasst. Beachten Sie dabei

- a) Aufbau und Elemente von Nassi-Shneiderman- und Flussdiagrammen,
- b) Grundstruktur und Komponenten eines Java-Programms,
- c) formale und sprachliche Besonderheiten in Java,
- d) Datentypen,
- e) Variablendeklaration,
- f) Syntaxregeln etc.

Aufgabe 4. Nachmachen

Versuchen Sie nun, obige Programme selbst zu erstellen und eventuell leicht abzuändern (verbessern, ergänzen etc.)

Aufgabe 5. Puzzle Datentypen

Ordnen Sie die folgenden Werte den richtigen Datentypen in der Tabelle zu. Als Hilfestellung ist bereits je ein Wert für jeden Datentyp vorhanden.

'\u0061'	"Hallo"	-25874	true		3598,27	'\f'
-28			'\n'	16	false	-9
5.64f	-18,6	19	39541	'\t'	'\u00ae'	3,7
-114	'\u20AC'	"Guten Morgen!"	-3.3f	1.9865f	1.3f	127
double	int	float	char	String	byte	boolean
4,98745	-28	5.64f	'\t'	"Hallo"	127	true

B Üben

Vom Problem zum Algorithmus: Problemstellungen „verstehen“ und „übersetzen“

Problemstellungen zu verstehen ist nicht immer einfach, das muss man lernen. Das hat nicht unbedingt damit zu tun, dass Angaben auf Übungszetteln oder bei Klausuren manchmal schwer zu verstehen sind, weil sie Fachbegriffe enthalten, die Ihnen noch nicht so vertraut sind. Im Gegenteil, in der Praxis sind Sie eher mit einer Situation wie der folgenden konfrontiert:

Ein(e) Nicht-InformatikerIn hat ein bestimmtes Problem, das sich durch ein einfaches Programm leicht lösen ließe. Sie sollen das Problem als ProgrammiererIn lösen, haben jedoch in diesem Bereich noch nicht gearbeitet und dadurch keine Erfahrung gesammelt. Der/Die AuftraggeberIn erklärt Ihnen das Problem und die Anforderungen an das Programm. Auch wenn Sie auf Anhieb verstehen, was dieses neue Programm leisten und wie das Ergebnis aussehen soll, müssen Sie die Problemstellung doch genau analysieren und in „Ihre Sprache“ (Algorithmen etc.) übersetzen, um nicht nur eine lauffähige, sondern auch eine wirklich gute Anwendung zu programmieren.

Dies gelingt am besten, wenn Sie sich viele Fragen stellen:

- Wie soll das Ergebnis aussehen?
- Welche Aufgaben hat das Programm?
- In welcher Reihenfolge sollen sie durchgeführt werden?
- Welche Einzelschritte sind notwendig?
- Welche Daten sollen eingegeben und welche ausgegeben werden?
- Welche Variablen und Konstanten, welche Datentypen werden benötigt?
- Wo, wann und wie werden die Daten eingegeben oder aus einer externen Datei eingelesen?
- Welchen Daten gehören zusammen?
- Wo verwende ich einfache Datentypen, wo Arrays, wo brauche ich Klassen?
- Wo sind Verzweigungen nötig?
- Wo sind Schleifen nötig? Welche sind wofür geeignet?

Berücksichtigen Sie diese Fragen auch bei den nächsten Programmieraufgaben!

Aufgabe 6. Einfache Rechnungen

Versuchen Sie nun, das Erlernte anzuwenden und erstellen Sie eigene Algorithmen und einfache Programme nach dem Muster der Lesecke, z.B.

- a) das Produkt zweier Zahlen,
- b) eine Subtraktion zweier Zahlen,
- c) eine Division zweier Zahlen,
- d) einen Brutto-Netto-Rechner, der aus einem eingegebenen Bruttobetrag die Umsatzsteuer (10% oder 20%) und den Nettobetrag berechnet.

Beachten Sie dabei auch die Hinweise „Vom Problem zum Algorithmus“, die Ihnen dabei helfen sollen, Problemstellungen zu verstehen, zu analysieren und so aufzubereiten, dass Sie in Programmcode umgesetzt werden können.

Aufgabe 7. Datentypen

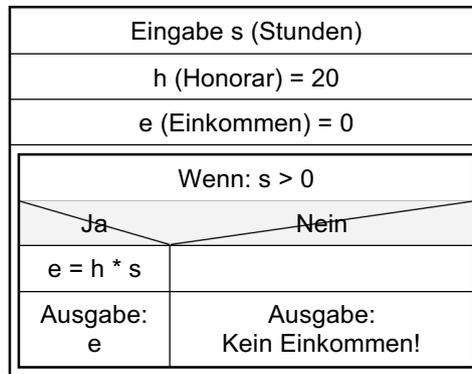
Versuchen Sie, für die untenstehenden Datentypen eine sinnvolle Deklaration zu finden (z.B. Datentyp String für name), initialisieren Sie die Variablen (z.B. String name = „Anna“) und geben Sie alle mit `System.out.println()` aus.

Datentypen: String, boolean, int, double, float, long, char;

Aufgabe 8. Einkommen

Sehen Sie sich nebenstehendes Nassi-Shneiderman-Diagramm an. Formulieren Sie dafür

- einen passenden Algorithmus in Pseudocode und
- stellen Sie ihn als Flussdiagramm dar.
- Implementieren Sie dazu eine Java-Klasse „Einkommen“ in Eclipse.



Aufgabe 9. Nachhilfe

Formulieren Sie Algorithmen für folgende Aufgaben und stellen Sie sie in Form von Form von Pseudocode und Flussdiagrammen dar. Überlegen Sie sich, welche Variablen und Datentypen gebraucht werden.

- Nachhilfestunden*
Sie möchten wöchentlich Ihre Nachhilfestunden eingeben und am Monatsende die Gesamtstunden ausgeben.
- Einkommen*
Aus obigen Gesamtstunden soll das Monateinkommen berechnet werden.
- Statistik* (für Fortgeschrittene)
Die Nachhilfestunden sind unregelmäßig verteilt, daher möchten Sie sie täglich eingeben und die durchschnittlichen Stunden pro Woche sowie pro Monat berechnen. Außerdem sollen die maximale und minimale Stundenzahl pro Woche und pro Monat ausgegeben werden.

Aufgabe 10. Algorithmen & Co

- Formulieren Sie Algorithmen für folgende Problemstellungen. Überlegen Sie dabei, was die einzelnen Anwendungen können sollen und welche Schritte dabei ablaufen. Den Umfang und die Komplexität der Anwendungen bestimmen Sie, je nach Ihren persönlichen Vorkenntnissen, selbst. Das kann z.B. vom einfachen Nachschlagen einer Nummer im Telefonbuch bis zum Erstellen eines neuen Telefonbuchs mit Eingabe und Abfrage von Nummern und Adressen gehen.
 - ein Telefonbuch
 - einen Stundenplan
 - ein Deutsch – Englisch Wörterbuch
 - einen Vokabeltrainer
- Stellen Sie außerdem obige Algorithmen in Form von
 - Flussdiagrammen und / oder
 - Pseudocode dar.

Aufgabe 11. Datentypen

1. Sehen Sie sich die nebenstehende Ausgabe eines Java-Programms an und überlegen Sie sich dafür
 - a) einen Algorithmus,
 - b) welche Datentypen benötigt werden.
2. Ergänzen Sie Ihren „Schwindelzettel“ aus Aufgabe 3 mit den wichtigsten Informationen zu den verschiedenen Datentypen.

```
Hallo lieber Nutzer!  
Bitte 1. Zahl eingeben: 4  
Bitte 2. Zahl eingeben: 3  
Bitte Operator eingeben: *  
Ihr Ergebnis ist: 12.0  
Noch etwas rechnen?  
ja = 1, nein = 2:
```

Aufgabe 12. Taschenrechner

1. Im folgenden Programmcode wurden alle Datentypen übermalt und einige Fehler noch nicht korrigiert. Lesen Sie ihn aufmerksam durch und lösen Sie folgende Aufgaben:
 - a) Ergänzen Sie den Code mit passenden Datentypen.
 - b) Markieren Sie die Fehler und versuchen Sie sie zu korrigieren.
2. Beantworten Sie außerdem folgende Fragen:
 - a) Welche Kontrollstrukturen sind vorhanden?
 - b) Welche Schlüsselwörter (reservierte Wörter) wurden verwendet?
 - c) Welche Methoden beinhaltet das Programm?
 - d) Welche Zugriffsattribute werden verwendet?

```
public class Taschenrechner {  
    static Scanner in = new Scanner(System.in);  
    static String fehlermeldung = "Ungültiger Operator!"  
    static int zahl1 = 0, zahl2 = 0  
    static String operator = ""  
    static String verabschiedung = "Danke fürs Benutzen! :)"  
    static String menueoption  
    static boolean kleinerals = false  
    static String begruessung = "Hallo lieber Nutzer!";  
    static boolean fehler = false  
    static int ergebnis = 0  
    static String operator2 = ' '  
    public static void main(String[] args) {  
        System.out.println(begruessung);  
        do {  
            System.out.println("Bitte 1. Zahl eingeben: ");  
            zahl1 == in.nextString ();  
            System.out.println("Bitte 2. Zahl eingeben: ");  
            zahl2 == in.nextString ();  
            System.out.println("Bitte Operator eingeben: ");  
            operator == in.next();  
            if (zahl1 < zahl2) {  
                kleinerals = true;  
            }  
            ergebnis = calculate(zahl1, zahl2, operator, kleinerals);  
            if (fehler)  
                System.out.println(fehlermeldung);  
                fehler = false;  
            else {  
                System.out.println(Ihr Ergebnis ist: + ergebnis);  
            }  
            System.out.println("Noch etwas rechnen? ja = 1, nein = 2: ");  
            menueoption = in.nextString ();  
        } while (menueoption = 1);  
        System.out.println(verabschiedung);  
    }  
}
```

```
public static double calculate(           a,            b, String x,           
zahl) {
               e = 0;
    if (zahl)
        e = a;
        a = b;
        b = e;
        e = 0;

    operator2 = x.charAt(0);
    switch (operator2) {
    case '+':
        e = a + b;
        break;
    case '-':
        e = a - b;
        break;
    case '*':
        e = a * b;
        break;
    case '/':
        e = a / b;
        break;
    default
        fehler = true;
    }
    return e;
}
}
```

2 Einfache Programme

A Entdecken

A 1 Lesecke Einfache Programme

```
import java.util.Scanner;

public class Bestanden {
    static Scanner in = new Scanner(System.in);
    public static void main(String[] args) {
        System.out.println("Punkte Zwischenklausur: ");
        int z, k, m;
        z = in.nextInt();
        System.out.println("Punkte Klausur: ");
        k = in.nextInt();
        System.out.println("Punkte Mitarbeit: ");
        m = in.nextInt();
        int p=z+k+m;
        System.out.println("Gesamtpunkte = " + p);
        if (p>50)
            System.out.println("Bestanden");
        else
            System.out.println("Leider nicht bestanden");
    }
}
```

Programm 1

```
import java.util.Scanner;

public class Noten {
    static Scanner in = new Scanner(System.in);
    public static void main(String[] args) {
        System.out.println("Note Zwischenklausur: ");
        double z, k, m;
        z = in.nextDouble();
        System.out.println("Note Klausur: ");
        k = in.nextDouble();
        System.out.println("Note Mitarbeit: ");
        m = in.nextDouble();

        double x = (z+k+m)/3;
        System.out.println("Gesamtnote = " + x);
        System.out.println("Soll die Note aufgerundet werden? 1 = ja, 0 = nein");
        int r = in.nextInt();
        int n = 0;

        if (r == 0){
            n = (int) x;
            System.out.println("Note = "+n);
        }
        else
            n = (int) (x + 0.5);
        System.out.println("Note = "+n);
    }
}
```

Programm 2

```
public class Exponent {
    public static void main(String[] args) {
        int anzahl;
        int basis = 5;
        int exponent = 2;
        int ergebnis = 1;

        for (anzahl = 0; anzahl < exponent; ++anzahl) {
            ergebnis *= basis;
        }
        System.out.print(basis + " hoch " + exponent + " = " + ergebnis);
        System.out.print('\n' + "Anzahl der Durchläufe: " + anzahl);
    }
}
```

Programm 3

<pre> import java.util.Scanner; public class Noten { static Scanner sc = new Scanner(System.in); public static void main(String[] args) { String name; int alter; int hilfsvARIABLE; boolean student; double notendurchschnitt; System.out.println("Bitte geben Sie Ihren Namen ein:"); name = sc.next(); System.out.print("Bitte geben Sie Ihr Alter ein: "); alter = sc.nextInt(); System.out.print("Sind Sie Student? (Für Ja 1 eingeben, für Nein 0): "); hilfsvARIABLE = sc.nextInt(); if(hilfsvARIABLE == 1) { student = true; System.out.print("Bitte geben Sie Ihren Notendurchschnitt ein: "); notendurchschnitt = sc.nextDouble(); } else { student = false; notendurchschnitt = 0.0; } System.out.print("Ihr Name ist " + name + ", Sie sind " + alter + " Jahre alt"); if(student == true) { System.out.println(" und Sie sind Student mit einem Notendurchschnitt von " + notendurchschnitt + "."); } else { System.out.println(" und Sie sind kein Student."); } } } </pre>	<p>Ausgabe:</p> <pre> Bitte geben Sie Ihren Namen ein: Ferdinand Bitte geben Sie Ihr Alter ein: 20 Sind Sie Student? (Für Ja 1 eingeben, für Nein 0): 1 Bitte geben Sie Ihren Notendurchschnitt ein: 3,2 Ihr Name ist Ferdinand, Sie sind 20 Jahre alt und Sie sind Student mit einem Notendurchschnitt von 3.2. </pre>	<p>Programm 4</p>
--	--	--------------------------

A 2 Lesecke Schleifen

<pre> import java.util.Scanner; public class Zahlen { static Scanner sc = new Scanner(System.in); public static void main(String[] args) { int zahl; int versuche = 0; int eingabe; System.out.print("Spieler 1, bitte Zahl zwischen 1 und 20 eingeben: "); zahl = sc.nextInt(); do { System.out.print("Spieler 2, bitte Zahl raten: "); eingabe = sc.nextInt(); versuche++; } while(zahl != eingabe); System.out.println("Gratuliere! Du hast die Zahl nach " + versuche + " Versuch(en) gefunden!"); } } </pre>	<p>Ausgabe:</p> <pre> Spieler 1, bitte Zahl zwischen 1 und 20 eingeben: 7 Spieler 2, bitte Zahl raten: 2 Spieler 2, bitte Zahl raten: 14 Spieler 2, bitte Zahl raten: 7 Gratuliere! Du hast die Zahl nach 3 Versuch(en) gefunden! </pre>	<p>Programm 5</p>
---	---	--------------------------

A 3 Lesecke Verzweigungen

```
public class Verzweigungen {  
  
    static Scanner sc = new Scanner(System.in);  
  
    public static void main(String[] args) {  
        System.out.println("Bitte den Tag eingeben: ");  
        int day = sc.nextInt();  
  
        if (day < 1 || day > 31) {  
            System.out.println("Kein gueltiges Datum fuer den Monat Dezember.");  
        } else {  
            switch (day) {  
                case 8:  
                    System.out.println("Feiertag: Maria Empfaengnis.");  
                    break;  
                case 24:  
                    System.out.println("Weihnachten, aber kein Feiertag");  
                    break;  
                case 25:  
                    System.out.println("Feiertag: Christtag.");  
                    break;  
                case 26:  
                    System.out.println("Feiertag: Stephanitag.");  
                    break;  
                case 31:  
                    System.out.println("Silvester, aber kein Feiertag.");  
                    break;  
                default:  
                    System.out.println("Leider kein Feiertag :(");  
            }  
        }  
    }  
}
```

A 4 Lesecke Schleifen

```
public class Schleifen {  
  
    static Scanner sc = new Scanner(System.in);  
  
    public static void main(String[] args) {  
        int i = 0;  
        int sum = 0;  
  
        do {  
            System.out.print("Bitte " + (i+1) + ". Zahl eingeben: ");  
            sum += sc.nextInt();  
            i++;  
        } while(i < 5);  
  
        System.out.println("1. Zwischensumme: " + sum);  
        i = 0;  
  
        while(i < 5) {  
            System.out.print("Bitte " + (i+1) + ". Zahl eingeben: ");  
            sum += sc.nextInt();  
            i++;  
        }  
  
        System.out.println("2. Zwischensumme: " + sum);  
  
        for(i=0; i < 5; i++) {  
            System.out.print("Bitte " + (i+1) + ". Zahl eingeben: ");  
            sum += sc.nextInt();  
        }  
  
        System.out.println("Endsumme 3: " + sum);  
    }  
}
```

Aufgabe 13. Lesen und Verstehen

Lesen Sie die obigen Programmcodes und beantworten Sie die folgenden Fragen für jedes Programm.

a) Was leistet jedes einzelne Programm?

- b) Welche Kontrollstrukturen werden verwendet?
- c) Welche Schlüsselwörter werden verwendet?

Aufgabe 14. Merken

Gestalten Sie einen „Schwindelzettel“ für ProgrammieranfängerInnen, der die in den obigen Beispielen ersichtlichen Gesetzmäßigkeiten und Regeln in übersichtlicher Form erfasst. Beachten Sie dabei

- a) formale und sprachliche Besonderheiten,
- b) Syntaxregeln,
- c) Verzweigungen,
- d) Schleifen etc.

Aufgabe 15. Nachmachen

Versuchen Sie nun, obige Programme der Lesecke selbst zu erstellen und eventuell leicht abzuändern (verbessern, ergänzen etc.).

B Üben

Aufgabe 16. Schleifen

Ändern Sie das Programm 3 der Lesecke *Einfache Programme* und implementieren Sie es mit

- a) einer while- Schleife,
- b) einer do-while-Schleife. Vergleichen Sie die beiden Schleifen mit der ursprünglichen for-Schleife. Welchen Unterschied zwischen den drei Möglichkeiten können Sie feststellen?

Aufgabe 17. Notenschlüssel

Das Programm soll eine Note auf der Basis des folgenden Notenschlüssels berechnen. Der Benutzer gibt eine Punkteanzahl ein (z.B. 74) und danach soll die richtige Note ausgegeben werden.

Notenschlüssel:

- weniger als 50 Punkte = Negativ
- 50 – 62 Punkte = Genügend
- 63 – 75 Punkte = Befriedigend
- 76 – 88 Punkte = Gut
- 88 Punkte = Sehr Gut

Beispiel Ausgabe:

```
Bitte Punkte eingeben: 74  
Befriedigend.
```

Aufgabe 18. Menüauswahl

Bei einer Hochzeit gibt es vier verschiedene Gerichte. Falls das Servicepersonal nicht weiß, welches Gericht zu welcher Nummer gehört, soll es mit einem kleinen Programm unterstützt werden. Der Benutzer gibt eine Zahl ein und das passende Gericht soll ausgegeben werden.

Menüauswahl:

- 1 = Zanderfilet in Paprikaschaum mit Kartoffeln und Gemüse
- 2 = Schweinsmedaillons in Pfefferrahmsauce mit Kroketten und Gemüse
- 3 = Risotto mit frischen Steinpilzen (Vegetarisch)
- 4 = Wienerschnitzel mit Pommes (Kinder)
- Default = Ungültige Menüauswahl.

Beispiel Ausgabe:

```
Bitte Menü eingeben: 4  
Wienerschnitzel mit Pommes (Kinder).
```

Aufgabe 19. Potenzrechnen

Sie wollen Potenzen berechnen. Der Benutzer soll die Basis (z.B.: 2) und den Exponenten (Hochzahl, z.B.: 4) eingeben. Das Ergebnis ($2^4 = 2*2*2*2 = 16$) soll jeweils mit Hilfe einer while-Schleife, einer do-while-Schleife und einer for-Schleife implementiert werden und danach ausgegeben werden.

Beispiel Ausgabe:

```
Bitte Basis eingeben: 2
Bitte Exponent eingeben: 4
Das Ergebnis lautet: 16.
```

Aufgabe 20. Brutto-Netto-Rechner

Schreiben Sie ein Java-Programm, das aus einem beliebigen Bruttobetrag die Umsatzsteuer (10% oder 20%) herausrechnet und diese gemeinsam mit dem Netto- und Bruttobetrag ausgibt.

Aufgabe 21. Gesamtbetrag brutto

Erstellen Sie ein einfaches Java-Programm, das folgende Ausgabe erzeugt:

```
Artikel 1 = 50
Artikel 2 = 20
Artikel 3 = 30
Gesamtbetrag netto = 100.0
20% Umsatzsteuer = 20.0
Gesamtbetrag brutto = 120.0
```

Aufgabe 22. Schleifen

1. Was leistet das folgende Programm, das leider nicht optimal gelöst ist?

```
import java.util.Scanner;
public class Punkte1 {
    static Scanner in = new Scanner(System.in);
    public static void main(String[] args) {
        int p=0, g=0;
        System.out.println("Punkte eingeben: ");
        p = in.nextInt();
        if (p>0) {
            g = g + p;
            System.out.println("Weitere Punkte eingeben: ");
            p = in.nextInt();
            g = g + p;
        }
        System.out.println("Gesamtpunkte : "+g);
        if (g >50)
            System.out.println("Bestanden");
        else
            System.out.println("Leider nicht bestanden");
    }
}
```

2. Was ist nicht optimal gelöst? Welche Verbesserungen könnten vorgenommen werden?

C Programmieren

Aufgabe 23. Punkteberechnung

Ändern Sie das Programm aus Aufgabe 22 und berücksichtigen Sie dabei folgendes:

- Sie möchten nur höchstens viermal Punkte eingeben.
- Sie wollen solange weitere Punkte eingeben, bis Sie mit einem Befehl die Eingabe beenden.
- Sie wollen auch den guten und sehr guten Erfolg (mehr als 80 bzw. 90 Pkt.) berücksichtigen.

Aufgabe 24. Nachhilfestunden verwalten

Schreiben Sie ein Java-Programm, das Nachhilfestunden verwalten soll. Um die Problemstellung zu analysieren und ein gutes Programm zu schreiben, beachten Sie wieder die Hinweise „Vom Problem zum Programm“ im Kapitel *Algorithmen und Datentypen*.

Folgende Punkte sollten in Ihrem Programm berücksichtigt werden:

a) *Nachhilfestunden*

Sie möchten Ihre Nachhilfestunden täglich eingeben und am Wochenende sowie am Monatsende die Gesamtstunden ausgeben.

b) *Einkommen/Stunde*

Aus obigen Gesamtstunden soll das Monatseinkommen berechnet werden.

c) *Statistik* (für Fortgeschrittene)

Die Nachhilfestunden sind unregelmäßig verteilt, daher möchten Sie die durchschnittlichen Stunden pro Woche und pro Monat berechnet. Außerdem sollen die maximale und minimale Stundenzahl pro Woche und pro Monat ausgegeben werden.

3 Eindimensionale Arrays

A Entdecken

A 1 Lesecke Arrays

Arrays	Deklarationen		Programm 1
	<pre>String[] namen = new String[6]; namen[0] = "Anita"; namen[1] = "David"; namen[2] = "Mario"; namen[3] = "Bonifaz"; namen[4] = "Rudolf"; namen[5] = "Erich"; double gleitkomma[] = {1.1, 2.2, 3.3, 4.4, 5.5,}; char[] zeichen = new char[5]; zeichen[0] = ':'; zeichen[1] = '/'; zeichen[2] = '&'; zeichen[3] = '!'; zeichen[4] = '?'; int zahl[] = new int[10]; for(int i=0; i < zahl.length; i++) { zahl[i] = i; }</pre>	<pre>import java.util.Scanner; public class Arrays1 { public static void main(String[] args) { int[] array = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}; printArray(array); array = turnAround(array); printArray(array); } public static void printArray(int[] array) { for(int i=0; i < array.length; i++) { System.out.print(array[i]+" "); } System.out.println(); } public static int[] turnAround(int[] array) { int temp; for(int i=0; i < array.length/2; i++) { temp = array[i]; array[i] = array[array.length - i - 1]; array[array.length - i - 1] = temp; } return array; } }</pre>	
			Programm 2

```
import java.util.Scanner;

public class Palindrom {

    public static void main(String[] args) {
        char[] word;
        System.out.print("Bitte Wort eingeben: ");
        word = readWord();

        if (isPalindrom(word)) {
            System.out.println(new String(word) + " ist ein Palindrom.");
        } else {
            System.out.println(new String(word) + " ist kein Palindrom.");
        }
    }

    public static char[] readWord() {
        return new Scanner(System.in).next().toCharArray();
    }

    public static boolean isPalindrom(char[] word) {
        boolean x = true;
        int i = 0;
        do {
            if (word[i] != word[word.length - 1 - i]) {
                x = false;
            }
            i++;
        } while (x == true && i < word.length / 2);
        return x;
    }
}
```

```

public class InsertSpaces {

    public static void main(String[] args) {

        String[] withoutSpaces = { "Informatik", "an",
"der", "Universitaet", "Klagenfurt" };
        System.out.println("Satz ohne Leerzeichen:");

        for (int i = 0; i < withoutSpaces.length; i++) {
            System.out.print(withoutSpaces[i]);
        }

        String[] withSpaces = new String[withoutSpaces.length * 2];

        for (int i = 0; i < withoutSpaces.length; i++) {
            withSpaces[i * 2] = withoutSpaces[i];
            withSpaces[i * 2 + 1] = " ";
        }

        System.out.println("\nSatz mit Leerzeichen:");

        for (int i = 0; i < withSpaces.length; i++) {
            System.out.print(withSpaces[i]);
        }
    }
}

```

Programm 3

Ausgabe:

```

Satz ohne Leerzeichen:
InformatikanderUniversitaetKlagenfurt
Satz mit Leerzeichen:
Informatik an der Universitaet Klagenfurt

```

```

public class PrintMethode {

    public static void main(String[] args) {

        String[] withoutSpaces = {"Informatik", "an", "der", "Universitaet",
"Klagenfurt" };
        System.out.println("Satz ohne Leerzeichen:");

        printArray(withoutSpaces);
        String[] withSpaces = new String[withoutSpaces.length * 2];

        for (int i = 0; i < withoutSpaces.length; i++) {
            withSpaces[i * 2] = withoutSpaces[i];
            withSpaces[i * 2 + 1] = " ";
        }

        System.out.println("\nSatz mit Leerzeichen:");
        printArray(withSpaces);
    }

    public static void printArray(String[] sentence) {
        for (int i = 0; i < sentence.length; i++) {
            System.out.print(sentence[i]);
        }
    }
}

```

Programm 4

Ausgabe:

```

Satz ohne Leerzeichen:
InformatikanderUniversitaetKlagenfurt
Satz mit Leerzeichen:
Informatik an der Universitaet Klagenfurt

```

Aufgabe 25. Lesen und Verstehen

Lesen Sie die obigen Programmcodes und beantworten Sie die folgenden Fragen für jedes Programm.

- Was leisten die einzelnen Programme?
- Wie werden Arrays deklariert und befüllt?
- Welche Kontrollstrukturen werden verwendet?
- Welche Schlüsselwörter werden verwendet?

Aufgabe 26. Merken

Gestalten Sie einen „Schwindelzettel“ für ProgrammieranfängerInnen, der die in den obigen Beispielen ersichtlichen Gesetzmäßigkeiten und Regeln rund um das Thema Arrays in übersichtlicher Form erfasst. Beachten Sie dabei auch

- a) formale und sprachliche Besonderheiten,
- b) Syntaxregeln,
- c) Verzweigungen,
- d) Schleifen etc.

Aufgabe 27. Nachmachen

Versuchen Sie nun, obige Programme selbst zu erstellen und eventuell leicht abzuändern (verbessern, ergänzen etc.).

A 2 Schritt-für-Schritt: Eindimensionale Arrays

Das Programm soll überprüfen, ob in einem Supermarkt ein spezielles Lebensmittel noch lagernd ist. Dazu benötigt man zuerst einen Scanner, um das Lebensmittel einlesen zu können:

```
public class Arrays {  
    static Scanner sc = new Scanner(System.in);
```

Als nächstes wird in der `main`-Methode ein `String[]` `supermarkt` erstellt und mit ein paar Lebensmitteln (Strings) befüllt. Außerdem erstellt man eine String Variable, in der man das eingelesene Lebensmittel speichert:

```
public static void main(String[] args) {  
    String[] supermarkt = new String[] {"brot", "nudel", "milch", "kaffee", "zucker"};  
    String lm = "";
```

Nun soll das Lebensmittel eingegeben werden:

```
System.out.println("Bitte Lebensmittel eingeben: ");  
lm = sc.next();
```

Nun erstellt man eine Variable für die `for`-Schleife, damit man später noch darauf zugreifen kann:

```
int i;
```

A 3 Schritt-für-Schritt: Ein Bibliotheksprogramm

Sie sollen ein Bibliotheks-Programm so erweitern, dass die Bücher in einem Array gesammelt werden. Dazu müssen die Menüpunkte „Buch hinzufügen“ und „Alle Bücher anzeigen“ wie folgt überarbeitet werden:

1. Buch hinzufügen: Der Benutzer soll nach einem Namen für das Buch gefragt werden. Existiert bereits ein Bücher-Array, soll das Buch an der letzten Stelle eingefügt werden. Besteht noch kein Array, muss dieses erst, mit dem hinzugefügten Buch als erstes Element, erzeugt werden.
2. Alle Bücher anzeigen: Alle Bücher aus dem Bücher-Array sollen der Reihe nach ausgegeben werden. Sie sollen nummeriert sein und der Titel soll erscheinen.

Wenn der Benutzer nun 1 wählt, erscheint zuerst der Aufruf, den Titel des Buches einzugeben. Ist dies erfolgt, gelangt der Benutzer wieder zurück in das Menü. Wählt der Benutzer Punkt 2 aus, so werden alle im Array gespeicherten Bücher der Reihe nach angezeigt. Mit Auswahl 3 wird das Programm weiterhin beendet.

Schritt-für-Schritt Lösung

Im ersten Schritt müssen Sie sich überlegen, ob weitere Variablen benötigt werden. Zusätzlich zu der bereits bestehenden Scanner Variable `in` und der `int`-Variable `wahl` wird nun ein Array für die Speicherung der Bücher benötigt. Da die Bücher mit Namen gespeichert werden sollen, muss das Array vom Typ `String` sein:

```
import java.util.Scanner;
public class Arrays {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int wahl = 0;
        String[] buecher = null;
    }
}
```

Bis zu der `switch`-Auswahl bleibt das ursprüngliche Programm unverändert:

```
while (wahl != 3) {
    System.out.println("Willkommen in der Bibliotheksverwaltung");
    System.out.println("Sie haben folgende Moeglichkeiten: ");
    System.out.println(" 1 - Buch hinzufuegen");
    System.out.println(" 2 - Alle Buecher anzeigen");
    System.out.println(" 3 - Programm beenden");
}

wahl = in.nextInt();
switch (wahl) {
```

In den nächsten Schritten wird nun der Menüpunkt „1 – Buch hinzufügen“ erweitert.

Zuerst muss die Ausgabe so geändert werden, dass ein Aufruf an den Benutzer erfolgt, einen neuen Buchtitel einzugeben:

```
case 1:
    System.out.println("Bitten neuen Buchnamen eingeben: \n");
```

Die Eingabe des Benutzers muss in einem `String` gespeichert werden. Dazu wird eine Variable erzeugt, der durch den Befehl `in.next()` der vom Benutzer eingegebene Buchnamen zugewiesen wird:

```
String buch = in.next();
```

Nun ist es notwendig zu überprüfen, ob das Array `buecher` bereits existiert oder erst erzeugt werden muss. Dies geschieht mit einer `if`-Abfrage. Ist `buecher` gleich `null`, wird das Array durch `new` mit dem Element `buch` erzeugt:

```
if (buecher == null) {
    buecher = new String[]{buch};
}
```

Besteht bereits ein Array `buecher`, wird ein temporäres Hilfsarray `tmp` benötigt, da sich die Länge eines Arrays nachträglich nicht verändern lässt. Dieses wird wiederum mit `new` erzeugt und soll genau eine Stelle mehr haben als das Array `buecher`. Durch eine `for`-Schleife wird jedes Element des Arrays `buecher` in das `tmp`-Array „kopiert“. In das letzte Feld des `tmp`-Arrays, also an die Stelle `buecher.length`, wird das neue Buch `buch` geschrieben. Danach wird dem Array `buecher` das Array `tmp` zugewiesen. Somit beinhaltet `buecher` nun auch das neu hinzugefügte Buch:

```
else {
    String[] tmp = new String[buecher.length+1];
    for (int i=0; i < buecher.length; i++) {
        tmp[i] = buecher[i];
    }
    tmp[buecher.length] = buch;
    buecher = tmp;
}
```

Damit ist der Menüpunkt „1 – Buch hinzufügen“ fertig.

Nun kommen wir zum Menüpunkt „2 – Alle Bücher anzeigen“. Hier muss ebenfalls zuerst die erste Ausgabe verändert werden. Danach wird jedes Feld des Arrays `buecher` über eine `for`-Schleife in der Form „Buch Stelle im Array: Buchname“ ausgegeben (z.B. Buch 2: Programmieren lernen mit Java).

```
case 2:
    System.out.println("Alle Buecher: \n");
    for (int i=0; i < buecher.length; i++) {
        System.out.println("Buch " + (i+1) + ": " + buecher[i]);
    }
    System.out.println("");
    break;
```

Damit sind die neuen Elemente eingebaut und der Rest des Programms wird unverändert übernommen:

```
case 3:
    System.out.println("Das Programm wird beendet\n");
    break;
default:
    System.out.println("Ungueltige Eingabe, bitte 1, 2 oder 3 waehlen\n");
}
}
in.close();
}
```

In der `for`-Schleife wird nun geprüft, ob das Array `supermarkt` das Lebensmittel enthält. Wenn das Lebensmittel im Array vorkommt, wird das auf der Konsole ausgegeben und die Schleife wird abgebrochen.

```
for(i = 0; i < supermarkt.length; i++) {
    if(!lm.equalsIgnoreCase(supermarkt[i])) {
        System.out.println("Lebensmittel ist im Supermarkt lagernd.");
        break;
    }
}
```

Am Ende wird nochmals überprüft, ob das Lebensmittel im Array war. Dazu überprüft man, ob die Zählvariable `i` gleich der `supermarkt` Länge ist. Somit kann man auf der Konsole ausgeben, dass das Lebensmittel nicht gefunden wurde.

```
if(i == supermarkt.length) {
    System.out.println("Lebensmittel leider nicht lagernd.");
}
}
```

Aufgabe 28. Puzzle – Binäre Suche

Das folgende Java-Programm ist durcheinander geraten. Versuchen Sie, die richtige Reihenfolge wieder herzustellen.

```
public static int binaerSuchenIterativ(int[] liste, int suchen, int
first, int last) {
    boolean found = false;
    int m = -1;
```

```
import java.util.Scanner;

public class Binaere_Suche {
```

```
    else if(suchen < liste[m]) {
        last = m-1;
    } else {
        first = m+1;
    }
}
```

```

    e = binaerSuchenIterativ(liste, suchen, 0, liste.length - 1);
    if (e == -1) {
        System.out.println("Element nicht gefunden.");
    } else {
        System.out.println("Position des Elements (startet bei 0):
        " + (e));
    }
}

```

```

public static void main(String[] args) {
    int[] liste = {2, 3, 5, 6, 9, 10, 14, 16, 17, 19, 20, 22, 25, 30,
    32, 35, 36, 37, 40};
    int suchen;
    int e;
}

```

```

if(!found) {
    m = -1;
}
return m;
}

```

```

while(first <= last && found == false) {
    m = (first + last) / 2;

    if(suchen == liste[m]) {
        found = true;
    }
}

```

B Üben

Aufgabe 29. Arrays ausgeben (Lesecke)

Die Arrays `namen`, `gleitkomma` und `zahl` aus der Lesecke (*Arrays - Deklarationen* links oben in der Lesecke in A.1) sollen ausgegeben werden. Vervollständigen Sie den Code in Eclipse so, dass alle Arrays am Bildschirm ausgegeben werden.

Aufgabe 30. Arrays ausgeben

Deklarieren Sie zur Übung eigene Arrays mit verschiedenen Datentypen und geben Sie diese am Bildschirm aus.

Aufgabe 31. Array initialisieren mittels `random()`

Deklarieren und initialisieren Sie ein `int`-Array der Größe 10. Füllen Sie das Array mit Zufallszahlen zwischen 0 und 99. Um eine Zufallszahl zu generieren, benötigen Sie die Klasse `Math` mit der Methode `random()`. Die Anweisung `double x = Math.random()` initialisiert `x` mit einem Wert größer gleich 0.0 und kleiner 1.0 (z.B. 0.6343...). Danach geben Sie das Array wieder aus.

Aufgabe 32. Ein kleiner Modulo Rechner

Der Benutzer möchte mehrere Zahlen eingeben und von diesen den Modulo berechnen. Am Ende sollen alle Ergebnisse ausgegeben werden. Für den genauen Ablauf des Programms halten Sie sich bitte an die Ausgabe rechts.

Beispiel Ausgabe:

```

Wie viele Zahlen moechten Sie eingeben?: 2
Mit welcher Zahl moechten Sie Modulo rechnen?: 5
Bitte 1. Zahl eingeben: 3
Bitte 2. Zahl eingeben: 7
Die berechneten Zahlen sind:
3, 2,

```

Aufgabe 33. Aus 2 mach 1

Sie haben ein `char`-Array mit dem Inhalt `{'B', 'a', 'u', 'm'}` und ein zweites `char`-Array mit dem Inhalt `{'h', 'a', 'u', 's'}`. Sie möchten daraus das Wort `Baumhaus` bilden, also ein drittes `char`-Array, mit dem Inhalt von Array 1 und Array 2 erstellen. Geben Sie danach das neue Array aus.

Aufgabe 34. Unterprogramm ohne Parameter und ohne Rückgabe

Erstellen Sie eine Methode `menue()`, die folgende drei Punkte auf die Konsole schreiben soll:

1. Vokabel lernen
2. Grammatiktest
3. Programm beenden

Beispiel Ausgabe:

1. Vokabel lernen
2. Grammatiktest
3. Programm beenden

Schreiben Sie eine `main()` Methode, die dieses Unterprogramm testet, also die Methode einmal aufruft.

Aufgabe 35. Unterprogramm mit Parameter und ohne Rückgabe

Schreiben Sie eine Methode `printArray(int[] x)`, die ein Array als Parameter übergeben bekommt und dieses Array auf der Konsole anschließend ausgibt. Testen Sie die Methode wieder in der `main()`.

Aufgabe 36. Unterprogramm ohne Parameter und mit Rückgabe

Schreiben Sie eine Methode `readInt()`, die von der Konsole eine `int`-Zahl einlesen soll (mit Hilfe des `Scanner`), diese Zahl in eine Variable speichert und anschließend an die `main()` zurückgibt. In der `main()` geben Sie dann diese Zahl mit `System.out.println()` auf der Konsole aus.

Aufgabe 37. Unterprogramm mit Parameter und mit Rückgabe

Schreiben Sie eine Methode `countArray(int[] x)`, die ein Array übergeben bekommt und zählen soll, wie viele Elemente das Array enthält (zählen mit `for`-Schleife und nicht mit `x.length` die Länge abfragen). Geben Sie die Länge zurück an die `main()` und schreiben Sie dort die Zahl mittels `System.out.println()` auf die Konsole.

Aufgabe 38. Arrays

Deklarieren und initialisieren Sie passende Arrayvariablen für folgende Daten und geben Sie anschließend zuerst alle Elemente und dann ein bestimmtes Element aus.

- a) Wochentage
- b) Monate
- c) Uhrzeit (für Terminplaner)
- d) Währungen (€, \$...)
- e) Steuersätze: Einkommen (0%, 36,5%, 43,2143%, 50%), USt (10%, 20%)
- f) Primzahlen
- g) Klausurnoten (1 -5, „Sehr gut“ ...)
- h) Mitarbeiter

C Programmieren

Aufgabe 39. Rechnen

Deklarieren und initialisieren Sie ein Array mit sechs ganzen Zahlen und führen Sie folgende Berechnungen durch:

- a) Bilden Sie die Summe und das Produkt aller Arrayelemente.
- b) Dividieren Sie die Summe der ersten drei Elemente durch die Summe der letzten drei.
- c) Berechnen Sie den Mittelwert, das Minimum und das Maximum der Arrayelemente.

Aufgabe 40. Punkteverwaltung - Menü

Sie haben bereits in Kapitel 3 - Einfache Programme (Aufgabe 22) eine kleine Punkteverwaltung kennengelernt, für die Sie nun ein Menü mit folgenden Optionen erstellen können:

1. Student hinzufügen
2. Punkte ändern
3. Punkte ausgeben
4. Noten ausgeben

Wählen Sie eine Menüoption aus:

4 Mehrdimensionale Arrays

A Entdecken

A 1 Lesecke Kyrrillische Tastatur

```
1 import java.util.Scanner;
2 public class Kyrillisch {
3
4     static char[][] translation = {
5         {'A', 0x410}, {'a', 0x430}, {'B', 0x411}, {'b', 0x431}, {'W', 0x412},
6         {'w', 0x432}, {'G', 0x413}, {'g', 0x433}, {'D', 0x414}, {'d', 0x434},
7         {'E', 0x415}, {'e', 0x435}, {'L', 0x416}, {'l', 0x436}, {'Z', 0x417},
8         {'z', 0x437}, {'I', 0x418}, {'i', 0x438}, {'J', 0x419}, {'j', 0x439},
9         {'K', 0x41A}, {'k', 0x43A}, {'L', 0x41B}, {'l', 0x43B}, {'M', 0x41C},
10        {'m', 0x43C}, {'N', 0x41D}, {'n', 0x43D}, {'O', 0x41E}, {'o', 0x43E},
11        {'P', 0x41F}, {'p', 0x43F}, {'R', 0x420}, {'r', 0x440}, {'S', 0x421},
12        {'s', 0x441}, {'T', 0x422}, {'t', 0x442}, {'U', 0x423}, {'u', 0x443},
13        {'F', 0x424}, {'f', 0x444}, {'H', 0x425}, {'h', 0x445}, {'C', 0x426},
14        {'c', 0x446}, {'3', 0x427}, {'4', 0x447}, {'5', 0x428}, {'6', 0x448},
15        {'7', 0x429}, {'8', 0x449}, {'-', 0x44A}, {'Y', 0x42B}, {'y', 0x44B},
16        {'_', 0x44C}, {'9', 0x42D}, {'0', 0x44D}, {'Q', 0x42E}, {'q', 0x44E},
17        {'X', 0x42F}, {'x', 0x44F},
18    };
19
20    public static void main(String[] args) {
21
22        System.out.println("Bitte Satz eingeben: ");
23        Scanner sc = new Scanner(System.in);
24        String line = sc.nextLine();
25        char[] lineArray = line.toCharArray();
26
27        String ausgabe = "";
28        for (int i = 0; i < lineArray.length; i++) {
29            ausgabe += translate(lineArray[i]);
30        }
31        System.out.println(ausgabe);
32
33        System.out.println("Der Satz wurde erfolgreich umgesetzt.");
34    }
35
36    public static char translate(char x) {
37        int i = 0;
38        char tmp = ' ';
39
40        while (tmp == ' ' && i < translation.length) {
41            if (x == translation[i][0]) {
42                tmp = translation[i][1];
43            } else {
44                i++;
45            }
46        }
47        return tmp;
48    }
49 }
```

Kyrillische Tastatur

Ausgabe:
Bitte Satz eingeben:
X u2inaq.
Я ужинаю.
Der Satz wurde erfolgreich umgesetzt.

Anmerkung:
Das kyrillische Alphabet ist in UTF-8 codiert. Damit es in Eclipse angezeigt werden kann, muss über *Window/Preferences* bei *Workspace Text file encoding* auf UTF-8 umgestellt werden. Buchstaben, die im lateinischen Alphabet keine Entsprechung haben, sind folgendermaßen codiert:

Display	UTF-8 Java	Display	UTF-8 Java
Ж	{'1', 0x416},	ь	{'1-', 0x44A},
ж	{'2', 0x436},	ь	{'1_', 0x44C},
Ч	{'3', 0x427},	Э	{'9', 0x42D},
ч	{'4', 0x447},	э	{'0', 0x44D},
Ш	{'5', 0x428},	Ю	{'Q', 0x42E},
ш	{'6', 0x448},	ю	{'q', 0x44E},
Щ	{'7', 0x429},	Я	{'X', 0x42F},
щ	{'8', 0x449},	я	{'x', 0x44F},

A 2 Lesecke Wochenstunden

```
1 public class Woche {
2
3     final static int Stunden = 10, ErsteStd = 8, mathe=0, ite=1, englisch=2;
4     public static void main(String[] args) {
5         String[]Tage = {"Montag","Dienstag","Mittwoch","Donnerstag","Freitag"};
6         String[]Fach = {"Mathematik", "Italienisch","Englisch"};
7         String[][]Plan = new String [Tage.length][Stunden];
8
9         for (int day = 0; day<Tage.length;day++)
10            for (int hour = 0; hour < Stunden; hour++)
11                Plan[day][hour]="";
12
13         for (int day = 0; day < Tage.length; day++) {
14             if (day!=2) Plan[day][0] = Fach[mathe];
15             if ((day%2)==1) Plan[day][2] = Fach[ite];
16             else Plan[day][2] = Fach[englisch];
17         }
18
19         for (int day = 0; day < Tage.length; day++) {
20             System.out.println(Tage[day] + ":");
21             for (int hour = 0; hour < Stunden; hour++) {
22                 if (!(Plan[day][hour]).equals(""))
23                     System.out.println(" " + (hour + ErsteStd) + ":" +
24 Plan[day][hour]);
25             }
26         }
27     }
28 }
```

Wochenstunden 1

```
Montag:
 8:Mathematik
10:Englisch
Dienstag:
 8:Mathematik
10:Italienisch
Mittwoch:
10:Englisch
Donnerstag:
 8:Mathematik
10:Italienisch
Freitag:
 8:Mathematik
10:Englisch
```

```
1 public class Woche2 {
2
3     final static int Stunden = 10, ErsteStd = 8, mathe=0, ite=1, englisch=2;
4     public static void main(String[] args) {
5         String[]Tage ={"Montag","Dienstag","Mittwoch","Donnerstag","Freitag"};
6         String[]Fach = {"Mathematik", "Italienisch","Englisch"};
7         String[][]Plan = new String [Tage.length][Stunden];
8
9         for (int day=0; day<Tage.length;day++)
10            for (int hour=0; hour<Stunden; hour++)
11                Plan[day][hour]="";
12
13         for (int day = 0; day < Tage.length; day++) {
14             if (day!=2) Plan[day][0] = Fach[mathe];
15             if ((day%2)==1) Plan[day][2] = Fach[ite];
16             else Plan[day][2] = Fach[englisch];
17         }
18
19         for (int day=0; day<Tage.length; day++)
20             System.out.print("\t"+ Tage[day]);
21         System.out.println("\n");
22
23         for (int hour=0; hour<Stunden; hour++) {
24             System.out.print((hour+ErsteStd)+ "\t");
25             for (int day=0; day<Tage.length; day++) {
26                 System.out.print(Plan[day][hour)+"\t");
27             }
28             System.out.println();
29         }
30     }
31 }
```

Wochenstunden 2

Aufgabe 41. Fragen und Aufgaben zur Lesecke

1. Lesen Sie die drei Programme und ergänzen Sie Ihren Array-„Schwindelzettel“ mit den wesentlichen Informationen zu mehrdimensionalen Arrays.
2. Markieren Sie mit unterschiedlichen Farben die einzelnen Bauteile eines Java-Programms: Deklarationen, Datentypen, Verzweigungen, Zuweisungen, Methoden etc.
3. Überlegen Sie, ob und wie die Programme anders / besser / effektiver gestaltet werden können.
4. Versuchen Sie, einen komfortableren und übersichtlicheren Wochenplan zu programmieren.

A 3 Schritt-für-Schritt: Mehrdimensionale Arrays

In einem zweidimensionalen Array sind die Zimmerkapazität und die Zimmerbelegung einer Jugendherberge gespeichert. Es soll für jedes Zimmer die Kapazität und Belegung ausgegeben werden. Dazu benötigt man zuerst das zweidimensionale Array `room`:

```
public class Arrays2 {
    public static void main(String[] args) {
        //Zeile 1: Kapazitaet, Zeile 2: Belegung
        //Spalte 1: Zimmer 1, Spalte 2: Zimmer 2...
        int[][] room = new int[][] {{4,8,4,3,5,7}, {2,0,4,3,1,5}};
    }
}
```

Im Anschluss benötigt man eine doppelte `for`-Schleife, um das gesamte Array durchlaufen zu können:

```
for(int i = 0; i < room.length; i++) {
    for(int j = 0; j < room[i].length; j++) {
        System.out.println("Das " + (j+1) + ". Zimmer hat " + room[i][j] + " Plaetze und
            davon sind " + room[i+1][j] + " belegt.");
    }
    i++;
}
}
```

Da immer sofort Kapazität und Belegung gleichzeitig ausgegeben werden, müsste man nicht unbedingt eine doppelte Schleife schreiben, sondern man könnte auch nur die Spalten durchlaufen und bei der Ausgabe der Kapazität die Zeile 0 und bei der Belegung die Zeile 1 verwenden:

```
for (int j = 0; j < room[0].length; j++) {
    System.out.println("Das " + (j + 1) + ". Zimmer hat " + room[0][j]
        + " Plaetze und davon sind " + room[1][j] + " belegt.");
}
}
```

Aufgabe 42. Puzzle Konjugierer

Das folgende Java-Programm ist durcheinander geraten.

1. Versuchen Sie, die richtige Reihenfolge wieder herzustellen.
2. Erklären Sie, was das Programm leistet.
3. Das Programm funktioniert zwar, es ist aber nicht optimal. Wie könnten Sie es ausbauen bzw. verbessern? Beachten Sie dazu die Fragen aus Kapitel 1: B Üben - Vom Problem zum Algorithmus: Problemstellungen „verstehen“ und „übersetzen“.

```
if (zahl > 7 || zahl < 0) {
    System.out.println("ungueltige Eingabe!");
} else {
    konjugierer(input, zahl);
}
}
```

```
static String[][] verben = {
    { "sein", "essere", "sono", "sei", "e", "siamo", "siete", "sono" },
    { "haben", "avere", "ho", "hai", "ha", "abbiamo", "avete", "hanno" },
}
```

```
    { "gehen", "andare", "vado", "vai", "va", "andiamo", "andate", "vanno"
},
    { "essen", "mangiare", "mangio", "mangi", "mangia", "mangiamo",
"mangiate", "mangiano" },
    { "schlafen", "dormire", "dormo", "dormi", "dorme", "dormiamo",
"dormite", "dormono" } };
```

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

```
    String input;
    int zahl;
```

```
    System.out.println("Für gewünschte Person bitte Zahl eingeben:");
    System.out.println("1 für Nennform");
    System.out.println("2 für 1. Person EZ");
    System.out.println("3 für 2. Person EZ");
    System.out.println("4 für 3. Person EZ");
    System.out.println("5 für 1. Person MZ");
    System.out.println("6 für 2. Person MZ");
    System.out.println("7 für 3. Person MZ");
    System.out.println("0 für alle Personen");
    zahl = in.nextInt();

    else if (verben[i][0].equals(input)) {
        System.out.println(verben[i][zahl]);
    }
}
}
```

```
package zusatz;
import java.util.Scanner;

public class Verbenkonjungierer {
    static Scanner in = new Scanner(System.in);
```

```
    System.out.print("Bitte Verb eingeben: ");
    input = in.next();
```

```
public static void konjungierer(String input, int zahl) {
    for (int i = 0; i < verben.length; i++) {
        if (verben[i][0].equals(input) && zahl == 0) {
            for (int j = 0; j < verben[0].length; j++) {
                System.out.print(verben[i][j] + ", ");
            }
        }
    }
}
```

B Üben

Aufgabe 43. Bingospiel

Erstellen Sie ein Programm, das ein Los für ein Bingospiel simuliert und folgende Ausgabe liefert.

Ausgabe:

```
Sie haben einen Bingoschein mit 3 Zeilen und 3 Spalten.  
Bitte Nummer fuer Zeile 1 und Spalte 1 eingeben: 35  
Bitte Nummer fuer Zeile 1 und Spalte 2 eingeben: 75  
Bitte Nummer fuer Zeile 1 und Spalte 3 eingeben: 98  
Bitte Nummer fuer Zeile 2 und Spalte 1 eingeben: 13  
Bitte Nummer fuer Zeile 2 und Spalte 2 eingeben: 43  
Bitte Nummer fuer Zeile 2 und Spalte 3 eingeben: 76  
Bitte Nummer fuer Zeile 3 und Spalte 1 eingeben: 98  
Bitte Nummer fuer Zeile 3 und Spalte 2 eingeben: 11  
Bitte Nummer fuer Zeile 3 und Spalte 3 eingeben: 25  
Ihr Bingoschein:  
35 75 98  
13 43 76  
98 11 25
```

Aufgabe 44. Namenstabelle

Erstellen Sie ein Programm, das die Eingabe von mehreren Vor- und Nachnamen ermöglicht und als Output folgende Tabelle liefert.

Beispiel Ausgabe:

```
Wie viele Personen wollen Sie erfassen?: 4  
Bitte 1. Vornamen eingeben: Melanie  
Bitte 1. Nachnamen eingeben: Maier  
Bitte 2. Vornamen eingeben: Stefan  
Bitte 2. Nachnamen eingeben: Huber  
Bitte 3. Vornamen eingeben: Paul  
Bitte 3. Nachnamen eingeben: Sonntag  
Bitte 4. Vornamen eingeben: Andrea  
Bitte 4. Nachnamen eingeben: Muster  
  
Vorname      Nachname  
  
Melanie      Maier  
Stefan       Huber  
Paul         Sonntag  
Andrea       Muster
```

C Programmieren

Wie gehen Sie vor? Überlegen Sie sich bei den folgenden Aufgaben, welche Schritte von der Analyse der Problemstellung über den Entwurf eines Algorithmus bis zur Implementierung und dem reibungslosen Ablauf des Programms nötig sind. Je genauer Sie planen, desto besser wird das Ergebnis!

Aufgabe 45. Glossar

Sie möchten ein komfortables Glossar erstellen, in dem ProgrammieranfängerInnen alle wichtigen Fachbegriffe zur Programmierung in Java nachschlagen können.

Aufgabe 46. Wahrung

Sie mochten einen Wahrungsrechner erstellen, der jede Wahrung in jede andere Wahrung umrechnen kann.

Aufgabe 47. bung Partyservice

Sie benotigen eine kleine Verwaltung fur einen Partyservice. Der Preis ist abhangig von der Anzahl der Partybesucher. Kommen weniger als 20 Personen, bezahlt man € 500 fur den Partyservice, bei einer Besucherzahl zwischen 20 und 100 liegt der Preis bei € 4000 und ab 100 Personen kostet der Partyservice € 12000. Falls mehr als 340 Personen kommen, ist kein Partyservice mehr moglich. Das soll auch auf der Konsole ausgegeben werden.

Zusatzlich zu obigem Preismodell gibt es Zuschlage in der Hochsaison (Juli, August) und Abzuge fur die zwei Monate, die sehr wenig gebucht werden (November, Dezember). Es soll angegeben werden, in welchem Monat die Party stattfindet (im Idealfall als int). Findet sie im Juli statt, wird der Preis um 10 % erhohet, im August um 15 %. Findet die Party im November statt, werden 12 % vom Preis abgezogen, im Dezember sogar 18 %. Am Ende soll auf der Konsole ausgegeben werden, wie viel der Partyservice kostet.

5 Strings

A Entdecken

A 1 Lesecke Stringbeispiele

```
String satz1 = "Hallo. Ich bin der erste Satz. Man nennt mich auch String;";
String satz2 = "Hier gibt es einen 2. Satz, der ebenfalls ein String ist;";

System.out.println("8. Zeichen im 1. String: " + satz1.charAt(7));

for(int i=32; i < satz2.length(); i++) {
    System.out.print(satz2.charAt(i));
}

String wort1 = "ebenfalls";
String wort2 = "EbEnFaLls";

System.out.println("\n");
System.out.println("Compare gibt an, dass die Strings ungleich sind: " +
wort1.compareTo(wort2));
System.out.println("Bei Rückgabewert 0 sind die Strings gleich, da
Gross/Kleinschreibung ignoriert wird: " + wort1.compareToIgnoreCase(wort2) + "\n");

System.out.println("Equals gibt einen Boolean zurueck: " + wort1.equals(wort2));
System.out.println("Equals gibt einen Boolean zurueck, ignoriert
Gross/Kleinschreibung: " + wort1.equalsIgnoreCase(wort2) + "\n");

System.out.println("Gibt an, an welchem Index 'S' zuerst vorkommt: " +
satz1.indexOf("S"));
System.out.println("Äquivalent zu indexOf, nur dass der letzte Index angegeben
wird, an dem ein 'S' vorkommt: " + satz1.lastIndexOf("S") + "\n");

System.out.println("Gibt an, an welchem Index der Substring 'gibt es ein' zuerst
vorkommt: " + satz2.indexOf("gibt es ein"));
System.out.println("Gibt an, an welchem Index das wort1 ab der Stelle 10 vorkommt:
" + satz2.indexOf(wort1, 9) + "\n");

String neuerString = satz1.replace('a', 'f');
System.out.println("Replace ersetzt ein char (a) mit einem anderen char(f): " +
neuerString);
    neuerString = satz2.replaceAll(wort1, wort2);
    System.out.println("Und replaceAll ersetzt den 1. Substring mit dem 2. Substring:
" + neuerString + "\n");

    String[] stringArray = satz1.split(" ");
    System.out.println("Der satz1 wurde nun in ein Array unterteilt, mit dem
Trennzeichen ' ': " + stringArray[0] + " " + stringArray[5] + "\n");

neuerString = satz1.substring(7, 30);
System.out.println("Der Substring vom satz1 vom Index 7 - 30: " + neuerString +
"\n");

char[] charArray = satz2.toCharArray();
System.out.println("Erstellt ein char Array vom satz2: " + charArray[3] + " " +
charArray[19] + "\n");

System.out.println("Wandelt alle Grossbuchstaben in Kleinbuchstaben um: " +
wort2.toLowerCase());
System.out.println("Wandelt alle Kleinbuchstaben in Grossbuchstaben um: " +
wort2.toUpperCase());
```

Aufgabe 48. Fragen und Aufgaben zur Lesecke

1. Lesen Sie die obigen Beispiele für Strings und schreiben Sie einen „Schwindelzettel“ mit den wesentlichen Informationen zu Strings.
2. Überlegen Sie sich eigene Beispiele, in denen Sie die Methoden `Compare`, `Equals` und `Replace` ausprobieren.

A 2 Puzzle

Aufgabe 49. Puzzle

Welcher Quellcode passt zu welcher Methode? Ordnen Sie den einzelnen Quellcodes die jeweils passende Methodenbeschreibung (1-4) zu. Was geben die einzelnen Methoden aus?

1. Methode prüft das Ende eines Strings
2. Methode liefert ein Zeichen eines Strings
3. Methode liefert eine Zeichenfolge eines Strings
4. Methode prüft, ob ein String mit einer speziellen Zeichenkette beginnt

```
a) public static void main(String[] args)
{
    String string = "0123456789";
    System.out.println(string.substring(2,7));
}
```

```
b) public static void main(String[] args)
{
    String string = "0123456789";
    System.out.println(string.charAt(6));
}
```

```
c) public static void main(String[] args)
{
    String string = "JAVA.ProgrammersBase.NET";
    String search = "NET";

    System.out.println(string.startsWith(search));
    System.out.println(string.startsWith(search,21));
}
```

```
d) public static void main(String[] args)
{
    String string = "JAVA.ProgrammersBase.NET";
    String search = "NET";

    System.out.println(string.endsWith(search));
}
```

B Üben

Aufgabe 50. Array einlesen und ausgeben 1

Sie möchten mehrere Wörter einlesen, in ein Array von Strings speichern und anschließend wieder ausgeben.

Beispiel Ausgabe:

```
Wie viele Woerter moechten Sie eingeben?: 2
Bitte 1. Wort eingeben: programmieren
Bitte 2. Wort eingeben: entdecken
Sie haben folgende Woerter eingegeben:
programmieren, entdecken,
```

C Programmieren

Aufgabe 51. Array einlesen und ausgeben 2

Implementieren Sie ein kleines Wortratespiel. Dazu soll SpielerIn 1 ein Wort zu einer gewissen Kategorie (zum Beispiel Automarken oder Lehrveranstaltungen) eingeben. Der/die zweite SpielerIn soll nun raten, welches Wort der/die erste SpielerIn eingegeben hat (Groß- und Kleinschreibung soll nicht beachtet werden). Wenn SpielerIn 1 das Wort erratet, soll „Congratulations!“ ausgegeben werden und gefragt werden, ob er/sie noch eine Runde spielen möchten. Außerdem soll das Spiel mit dem Wort „Ende“ auch frühzeitig beendet werden können (auch während man rät). Am Ende soll auf der Konsole „goodbye“ ausgegeben werden.

Implementieren Sie das Beispiel jeweils mit einer for-Schleife, while-Schleife und do-while-Schleife.

6 Projekt – Sprachlernprogramm

Sie schreiben ein Programm, das FremdsprachenlehrerInnen unterstützen soll. Den Beginn machen Sie mit einer Menüauswahl und einem Vokabellernspiel. In den nächsten Aufgaben wird das Programm schrittweise erweitert und zu einem mehrsprachigen Lernprogramm ausgebaut.

Aufgabe 52. Menü (Arrays)

Ihr Sprachenlernprogramm ist über die Konsole steuerbar. Der/die BenutzerIn kann zwischen den möglichen Funktionen wählen bzw. das Programm beenden. Wenn eine Menüoption ausgewählt wird, wird das jeweilige Unterprogramm aufgerufen. Wenn eine Funktion ausgeführt wurde, wird das Menü wieder angezeigt. Wenn das Programm gestartet wird, soll folgende Ausgabe erscheinen:

```
1. Wörterbuch
2. Wortschatz lernen
3. Grammatik
4. Ende

Wählen Sie eine Menüoption aus:
```

Achten Sie darauf, auch ungültige Eingaben entsprechend zu verarbeiten und überlegen Sie außerdem, wie das Programm vorzeitig beendet werden kann.

Jede der oben genannten Menüoptionen enthält ein Untermenü, das Sie ebenfalls schon programmieren können. Überlegen Sie sich, welche Optionen in den einzelnen Untermenüs sinnvoll wären. Das Wörterbuch könnte z.B. folgende Menüeinträge haben:

```
1. Ausgangssprache
2. Zielsprache wählen
3. Wort hinzufügen
4. Wort nachschlagen
5. Programm beenden
```

Aufgabe 53. Vokabellernspiel „Geschüttelte Wörter“ (Strings)

1. Problemstellung:

Sie möchten für eine/n FremdsprachenlehrerIn ein Vokabellernspiel programmieren, das die Buchstaben vorgegebener Wörter¹ schüttelt bzw. durchmischt und diese am Bildschirm ausgibt. Die Lernenden sollen dann das passende Wort erraten und eingeben. Ist das eingegebene Wort falsch, soll solange weitergemacht werden, bis das richtige Wort erraten ist.

Die Ausgabe könnte z.B. so aussehen:

```
Welches Wort soll geschüttelt werden?
Für zufälliges Wort bitte -1 eingeben): 3
Ihr Wort lautet: mianla
Wie lautet das Wort?: animan
Leider falsch! Wie lautet das Wort?: animal
Super! Richtig!
```

2. Hinweise:

Bevor Sie sich ans Programmieren machen, beachten Sie folgende Fragen:

- Welche Vokabeln möchten Sie verwenden? – Möchten Sie sie direkt ins Array eingeben oder eine fertige Vokabelliste (als xls-, txt- und csv-Datei ist auf der Webseite zum Buch verfügbar) verwenden? (10 Vokabel reichen, außer für begeisterte FremdsprachenlernerInnen)

¹ Eine mehrsprachige Vokabelliste zum Thema „Familie“ ist auf der Webseite zum Buch als xls-, txt- und csv - Datei verfügbar.

- Welches Wort wird geschüttelt? Soll es zufällig ausgewählt oder in der Reihenfolge der Liste ausgegeben werden?
- Wie soll das geschüttelte Wort aussehen? Werden die Buchstaben nach einem Zufallsprinzip durchgemischt oder nach einem bestimmten System? Fortgeschrittene Studierende können sich selbst etwas überlegen und eine Methode schreiben. AnfängerInnen können die Methode „schuetteln“ (zwei nebeneinanderliegende Buchstaben werden vertauscht) von der Webseite zum Buch herunterladen.
- Wie viele Versuche haben die Lernenden? Soll die Leistung der Lernenden bewertet werden, z.B. mit Punkten für richtige Wörter und Punkteabzug für fehlerhafte Versuche? (Hinweis zum Lösungsweg: Punkte können in einem Array gespeichert oder durch einen eigenen Zähler ermittelt werden.)
- Was soll alles ausgegeben werden? Das richtige Wort, die Anzahl der Versuche, eventuell die erreichte Punktezahl etc.

Aufgabe 54. Wörterbuch

Es sollte bereits ein Menü wie in der Abbildung rechts vorhanden sein. Nun wird das Sprachlernprogramm um das Unterprogramm Wörterbuch erweitert.

```
1. Wörterbuch
2. Wortschatz lernen
3. Grammatik
4. Ende

Wählen Sie eine Menüoption aus:
```

- Das Wörterbuch soll mehrsprachig sein und die Möglichkeit bieten, von jeder beliebigen Ausgangssprache ein Wort in jeder beliebigen Zielsprache nachzuschlagen. Als Basis können Sie die fertige Vokabelliste „Familie“ (verfügbar auf der Website zum Buch) verwenden.
- Um ein möglichst komfortables Wörterbuch zu programmieren, beachten Sie bei der Planung wieder die Fragen aus Kapitel 1 - B Üben - Vom Problem zum Algorithmus: Problemstellungen „verstehen“ und „übersetzen“.

Erweitern Sie das Sprachlernprogramm um die folgenden Unterprogramme (Aufgabe 55 – 60). Die fremdsprachigen Beispiele (Wörter, Sätze und Texte) sind als txt-Dateien auf der Website zum Buch verfügbar.

Aufgabe 55. Sätze schütteln

Die Wörter eines vollständigen fremdsprachigen Satzes sollen in diesem Programm „geschüttelt“ werden und vom Lernenden in die richtige Reihenfolge gebracht werden. Sie können die Sätze aus einer externen Datei einlesen oder direkt in Ihr Java-Programm hineinschreiben.

Aufgabe 56. Lückentextgenerator jedes 7. Wort streichen

1. Aus einem fremdsprachigen Text soll ein Lückentext erstellt werden. Jedes 7. Wort soll gelöscht oder in einer Liste gesammelt werden. Die entstehende Lücke soll durch ein Fragezeichen (oder anderes Zeichen) angezeigt werden.
2. Fortgeschrittene ProgrammiererInnen können sich an der Profiversion versuchen und die Eingabe des Schülers bzw. der Schülerin gleich überprüfen und eventuell ein Feedback dazu geben.

Aufgabe 57. Endungen streichen (Plural), Eingabe überprüfen

1. Alle Pluralendungen des italienischen Texts (Endungen mit „e“ oder „i“) sollen gestrichen und durch Platzhalter ersetzt werden.
2. Die Lerner sollen die richtigen Endungen eingeben und ein Feedback bekommen.

Aufgabe 58. Groß-/Kleinschreibung

1. Sie erstellen ein Programm für VolksschullehrerInnen: In einem deutschen Text sollen in einem ersten Schritt alle großen Anfangsbuchstaben in kleine umgewandelt werden.
2. Die SchülerInnen sollen dann angeben, ob die Schreibweise richtig oder falsch ist. Die Anzahl der richtigen Antworten soll schließlich als Feedback zurückgegeben werden.

Aufgabe 59. Übung Wortratespiel

Implementieren Sie ein kleines Wortratespiel. Dazu soll SpielerIn 1 ein Wort zu einer gewissen Kategorie (zum Beispiel Automarken oder Lehrveranstaltungen) eingeben. Der/die zweite SpielerIn soll nun raten, welches Wort der/die erste SpielerIn eingegeben hat (Groß- und Kleinschreibung soll nicht beachtet werden). Wenn SpielerIn 1 das Wort errätet, soll "Congratulations!" ausgegeben werden und es soll gefragt werden, ob sie/er noch eine Runde spielen möchte. Außerdem soll das Spiel mit dem Wort "Ende" auch frühzeitig beendet werden können (auch während man rät). Am Ende soll auf der Konsole "Goodbye" ausgegeben werden.

7 Lösungen

Aufgabe 2. Programmcode verstehen

Das Programm „Noten“ liefert bei der Eingabe

- 1 die Ausgabe „Sehr gut“,
- 2 die Ausgabe „Gut“,
- 3 die Ausgabe „Befriedigend“,
- 4 die Ausgabe „Genügend“,
- 5 die Ausgabe „Leider nicht bestanden“.

Das Programm „Eingabe“ bildet die Summe von zwei beliebigen, ganzzahligen eingelesenen Zahlen und gibt sie am Bildschirm aus.

Aufgabe 5. Puzzle Datentypen

double	int	float	char	String	byte	boolean
-18.6	-28	5.64f	\u0061	"Hallo"	-28	true
4.98745	-114	-3.3f	\u20AC	"Guten	-114	false
359.27	-25874	1.9865f	'	Morgen!"	19	
3.7	19	1.3f	\n'		16	
	39541		\t'		-9	
	16		\u00ae'		127	
	-9		\f'			
	127					

Aufgabe 6. Einfache Rechnungen

Produkt

```
import java.util.Scanner;
public class Produkt {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Bitte Zahl a eingeben:");
        int a = sc.nextInt();
        System.out.println("Bitte Zahl b eingeben:");
        int b = sc.nextInt();
        int c = a * b;
        System.out.println("Das Produkt ist: " + c);
    }
}
```

Subtraktion

```
import java.util.Scanner;
public class Subtraktion {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Bitte Zahl a eingeben:");
        int a = sc.nextInt();
        System.out.println("Bitte Zahl b eingeben:");
        int b = sc.nextInt();
        int c = a - b;
        System.out.println("Die Differenz ist: " + c);
    }
}
```

Division

```
import java.util.Scanner;
public class Division {
    public static void main (String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.println("Bitte Zahl a eingeben:");
        int a = sc.nextInt();
        System.out.println("Bitte Zahl b eingeben:");
        int b = sc.nextInt();
        int c = a / b;
        System.out.println("Das Ergebnis ist: " + c);
    }
}
```

Brutto-Netto

```
public class BruttoNettoRechner {

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        double brutto = 0;
        double netto1 = 0; //für 10% USt
        double netto2 = 0; //für 20% USt
        double ust1 = 0;
        double ust2 = 0;
        Scanner in = new Scanner(System.in);

        System.out.println("Bitte geben Sie einen Bruttobetrag ein.");
        brutto = in.nextDouble();

        ust1 = (brutto/110)*10;
        ust2 = (brutto/120)*20;

        netto1 = brutto - ust1;
        netto2 = brutto - ust2;

        System.out.println("Bei 10% USt. beträgt der Nettobetrag " + netto1);
        System.out.println("Bei 10% USt. beträgt die USt " + ust1);
        System.out.println("Bei 20% USt. beträgt der Nettobetrag " + netto2);
        System.out.println("Bei 10% USt. beträgt die USt " + ust2);
    }
}
```

Aufgabe 7. Datentypen

```
public class Datentypen {
    public static void main(String[] args) {
        String name = "Anna";
        boolean wahr = true;
        int alter = 21;
        double grosseKommazahl = 2020403.14159;
        float kleineKommazahl = 2.2f;
        long grosseZahl = 900433899;
        char buchstabe = 'S';

        System.out.println("name: " + name);
        System.out.println("wahr?: " + wahr);
        System.out.println("alter: " + alter);
        System.out.println("Grosse Kommazahl: " + grosseKommazahl);
        System.out.println("Kleine Kommazahl: " + kleineKommazahl);
    }
}
```

```

        System.out.println("Grosse Zahl: " + grosseZahl);
        System.out.println("Buchstabe: " + buchstabe);
    }
}

```

Aufgabe 8. Einkommen

```

public class Einkommen {

    public static void main(String[] args) {
        int h = 20;
        int e = 0;
        int s = 5;

        e = h * s;

        if (s>0)
            System.out.println("Einkommen = "+e);
        else
            System.out.println("Kein Einkommen!");
    }
}

```

Aufgabe 9. Nachhilfe

1. Algorithmen

Nachhilfestunden & Einkommen:

1. Stunden eingeben
2. Ist die Eingabe > 0?
 - a. Wenn ja: Addiere die Zahl zur Gesamtstundenanzahl und gehe zu Nr. 1
 - b. Wenn nein: gehe zu Nr. 3
3. Ausgabe: Gesamtstunden (bei b) Ausgabe: Einkommen)

Statistik:

1. Anzahl der Monate einlesen
2. Ist die Anzahl der Monate > 0?
 - a. Wenn ja: gehe zu Nr. 3
 - b. Wenn nein: gehe zu Nr. 10
3. Anzahl der Tage einlesen
4. Ist die Anzahl der Tage > 0?
 - a. Wenn ja: gehe zu Nr. 5
 - b. Wenn nein:
 - i. Ausgabe: Gesamtstunden in diesem Monat, Minimum, Maximum, Durchschnitt
 - ii. Verringere Anzahl der Monate um 1
 - iii. gehe zu Nr. 2
5. Stunden einlesen
6. Stunden zu Gesamtstunden addieren
7. Ist die eingegebene Stundenanzahl Minimum?
 - a. Wenn ja: speichere Stundenanzahl als Minimum & gehe zu Nr. 8
 - b. Wenn nein: Gehe zu Nr. 8
8. Ist die eingegebene Stundenanzahl Maximum?
 - a. Wenn ja: speichere Stundenanzahl als Maximum & gehe zu Nr. 9
 - b. Wenn nein: gehe zu Nr. 9

9. Verringere Anzahl der Tage um 1 und gehe zu Nr. 4
10. Ende

3. Benötigte Variablen und Datentypen

Nachhilfestunden & Einkommen

Es wird je eine Variable vom Typ Integer benötigt um die gerade eingelesenen Stunden zu speichern und um die Gesamtstunden zu speichern. Außerdem: eine int-Variable für das Honorar und eine für das Einkommen.

Statistik

Hier werden zusätzliche Variablen vom Typ Integer benötigt um:

- Die Anzahl der Monate zu speichern
- Die Anzahl der Tage zu speichern
- Das aktuelle Minimum zu speichern
- Das aktuelle Maximum zu speichern
- Den Durchschnitt zu speichern
- zwei „Zählvariablen“ für die Schleifen

Aufgabe 10. Algorithmen & Co

1a) Telefonbuch

1. Lies Name ein
2. Vergleiche eingegebenen Namen mit Namen aus Telefonbuch, bis er gefunden ist.
3. Gib Telefonnummer aus

1b) Stundenplan

1. Rufe Stundenplan auf
2. Lies den Beginn der ersten Stunde des nächsten Tages aus
3. Wenn:
 - a. Wenn die erste Stunde um 8 Uhr beginnt: stehe um 7 Uhr auf
 - b. Wenn die erste Stunde um 9 Uhr beginnt: stehe um 8:20 Uhr auf
 - c. Wenn der Tag unterrichtsfrei ist: stehe um 10 Uhr auf
4. Stelle den Wecker für den nächsten Tag

1c) Deutsch-Englisch Wörterbuch

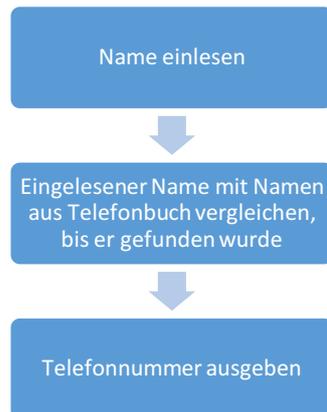
1. Lies das zu suchende Wort ein
2. Durchsuche das Wörterbuch solange, bis das Wort gefunden wird. Wenn:
 - a. Wort gefunden: Gib das englische Wort aus
 - b. Wort nicht gefunden: Gib Fehlermeldung aus
3. Gib die Anzahl der Wörter im Wörterbuch aus

1d) Vokabeltrainer

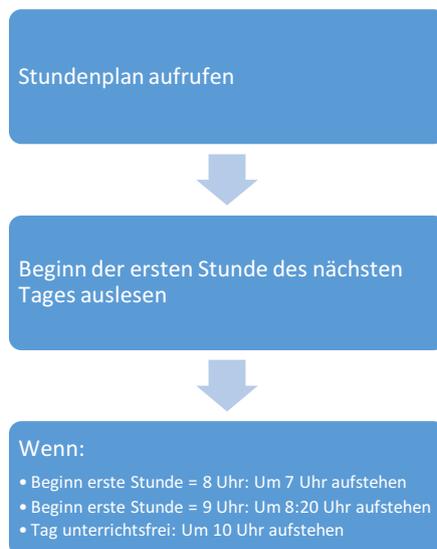
1. Wähle die gewünschte Fremdsprache aus
2. Wähle die Anzahl der zu prüfenden Vokabel aus
3. Initialisiere Variablen *amount* (Anzahl der Schritte) *counter* (zum Zählen der Schritte) und *points* (zum Zählen der Punkte)

4. Wiederhole, solange *counter* \leq *amount*:
 - a. Gib das Vokabel an Stelle *counter* aus
 - b. Gib das deutsche Wort ein
 - c. Überprüfe, ob das Wort korrekt ist. Wenn:
 - i. Wort korrekt: Erhöhe *points* um 1
 - ii. Wort nicht korrekt: nichts
 - d. Erhöhe *counter* um 1
5. Gib die Anzahl der Punkte aus

2a) Telefonbuch



2b) Stundenplan



2c) Deutsch-Englisch Wörterbuch

```
ausgabe("Gib das zu suchende Wort ein");
ingelesenesWort = eingabe;
solange(wortGefunden = nein) {
  setze counter = 1;
  vergleiche eingelesenesWort mit Wort an Stelle counter
  wenn(eingelesenesWort gleich Wort an Stelle counter) {
    ausgabe("Das englische Wort für eingelesenesWort lautet Wort an Stelle counter");
    wortGefunden = ja;
  } sonst {
    erhöhe counter um 1;
  }
}
```

2d) Vokabeltrainer

```
ausgabe("Gib die Zahl der Fremdsprache ein:");
ausgabe("1 - Italienisch");
ausgabe("2 - Französisch");
ausgabe("3 - Russisch");
gewählteSprache = eingabe;
ausgabe("Gib die Anzahl der zu prüfenden Vokabel ein");
amount = eingabe;
counter = 1;
points = 0;
solange (counter <= amount) {
  ausgabe(Vokabel an Stelle counter);
  deutschesWort = eingabe;
  wenn(deutschesWort gleich Lösung für Vokabel an Stelle counter) {
    erhöhe points um 1;
  }
  erhöhe counter um 1;
}
ausgabe("Anzahl der Punkte: points");
```

Aufgabe 11. Datentypen

1. Gib Hallo-Text aus
2. Solange das Programm läuft:
 - a. Lese 1. Zahl ein
 - b. Lese 2. Zahl ein
 - c. Lese Operator ein
 - d. Gib Ergebnis aus
 - e. Frage nach weiterer Rechnung. Wenn:
 - i. ja: alte Werte löschen, neue Rechnung beginnen
 - ii. nein: Programm beenden

Datentypen:

int/double (Zahlen), char (Operator), int (Auswahlmenü)

Aufgabe 12. Taschenrechner

```
public class Taschenrechner {
    static String fehlermeldung = "Ungültiger Operator!";
    static double zahl1 = 0, zahl2 = 0;
    static String operator = "";
    static String verabschiedung = "Danke fürs Benutzen! :)";
    static int menueoption;
    static boolean kleinerals = false;
    static String begruessung = "Hallo lieber Nutzer!";
    static boolean fehler = false;
    static double ergebnis = 0;
    static char operator2 = ' ';

    public static void main(String[] args) {
        System.out.println(begruessung);
        Scanner sc = new Scanner(System.in);
        do {
            System.out.println("Bitte 1. Zahl eingeben: ");
            zahl1 = sc.nextDouble();
            System.out.println("Bitte 2. Zahl eingeben: ");
            zahl2 = sc.nextDouble();
            System.out.println("Bitte Operator eingeben: ");
            operator = sc.next();
            if (zahl1 < zahl2) {
                kleinerals = true;
            }
            ergebnis = calculate(zahl1, zahl2, operator, kleinerals);
            if (fehler) {
                System.out.println(fehlermeldung);
                fehler = false;
            } else {
                System.out.println("Ihr Ergebnis ist: " + ergebnis);
            }
            System.out.println("Noch etwas rechnen? ja = 1, nein = 2: ");
            menueoption = sc.nextInt();
        } while (menueoption == 1);
        System.out.println(verabschiedung);
    }

    public static double calculate(double a, double b, String x, boolean
zahl) {
        double e = 0;
        if (zahl) {
            e = a;
            a = b;
            b = e;
            e = 0;
        }
        operator2 = x.charAt(0);
        switch (operator2) {
            case '+':
                e = a + b;
                break;
            case '-':
                e = a - b;
                break;
            case '*':
                e = a * b;
                break;
            case '/':
                e = a / b;
                break;
        }
    }
}
```

```

    default:
        fehler = true;
    }
    return e;
}
}

```

Aufgabe 12. Taschenrechner – Antworten zu Fragen

Beantworten Sie außerdem folgende Fragen:

- e) Welche Kontrollstrukturen sind vorhanden?
 - f) Welche Schlüsselwörter (reservierte Wörter) wurden verwendet?
 - g) Welche Methoden beinhaltet das Programm?
 - h) Welche Zugriffsattribute werden verwendet?
-
- a) do-while, if/else, switch, break
 - b) public, class, static, double, int, boolean, char, void, do, new, false, true, if, else, while, switch, case, break, return
 - c) void main(String[] args), double calculate(double a, double b, String x, boolean zahl) bzw. System.out.println(), sc.nextDouble(), sc.next(), sc.nextInt(),
 - d) public

Aufgabe 13

Programm 1:

- a) Liest die Punkte der Zwischenklausur, der Klausur und der Mitarbeit ein. Addiert diese Punkte zur Gesamtpunktzahl. Ist die Gesamtpunktzahl größer als 50, wird „Bestanden“ ausgegeben, anderenfalls „Nicht bestanden“
- b) if/else
- c) import, public, class, static, new, void, int, if, else

Programm 2:

- a) Liest die Noten der Zwischenklausur, der Klausur und der Mitarbeit ein. Bildet aus diesen 3 Noten den arithmetischen Mittelwert und gibt diesen Anschließend aus. Fragt, ob die Note aufgerundet werden soll. Falls 1 eingegeben wird, wird die Note aufgerundet, anderenfalls abgerundet.
- b) if/else
- c) import, public, class, static, new, void, double, int, if, else

Programm 3:

- a) Rechnet den Wert einer Zahl mit Exponent aus und gibt an, wie oft die dafür genutzte Schleife durchlaufen wurde.
- b) for
- c) public, class, static, void, int, for

Programm 4:

- a) Liest den Namen, das Alter, sowie ob man Student ist ein. Ist man Student, wird man zusätzlich nach dem Notendurchschnitt gefragt. Die eingegebenen Werte werden anschließend ausgegeben.

- b) if/else
- c) import, public, class, static, new, void, boolean, int, double, if, else, true, false

Programm 5:

- a) Zahlenratespiel. Spieler 1 gibt eine Zahl ein, welche von Spieler 2 erraten werden soll. Zum Schluss wird ausgegeben, nach wie vielen Versuchen die Zahl erraten wurde.
- b) do/while
- c) import, public, class, static, new, void, int, do, while

Programm Verzweigungen:

- a) Gibt besondere Tage im Dezember aus. Es muss ein gültiger Tag des Dezembers eingegeben werden (anderenfalls wird eine Fehlermeldung ausgegeben), anschließend wird ausgegeben, ob es sich um einen normalen Tag oder um einen Feiertag handelt.
- b) if/else, switch, break
- c) public, class, static, new, void, int, if, else, switch, case, break, default

Programm Schleifen:

- a) Es wird die Funktionsweise der 3 unterschiedlichen Arten von Schleifen demonstriert. Es werden jeweils Zahlen eingegeben, welche zu einer Gesamtsumme addiert werden, die zum Schluss ausgegeben wird.
- b) do/while, while, for
- c) public, class, static, new, void, int, do, while, for

Aufgabe 16. Schleifen

a) While

```
import java.util.Scanner;
public class WhileSchleife {
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) {
        int zahl = 0;
        int versuche = 0;
        int eingabe = -1;
        System.out.print("Spieler 1 bitte Zahl zwischen 1 und 20 eingeben: ");
        zahl = sc.nextInt();
        while(eingabe != 0 && zahl != eingabe) {
            System.out.print("Spieler 2, bitte Zahl raten! (Zum Beenden bitte 0 eingeben): ");
            eingabe = sc.nextInt();
            versuche++;
        }
        if(eingabe == 0) {
            System.out.println("Danke fürs Spielen!");
        } else {
            System.out.println("Gratuliere! Du hast die Zahl nach " + versuche + "
Versuch(en) gefunden!");
        }
    }
}
```

b) Do -While

```
public class DoWhile {
    public static void main(String[] args) {
        int anzahl;
        int basis = 5;
        int exponent = 2;
        int ergebnis = 1;
        anzahl = 0;
        do{
            ergebnis *= basis;
            ++anzahl;
        }while(anzahl < exponent);
        System.out.println (basis + "hoch" + exponent + "=" + ergebnis);
        System.out.println ('\n' + "Anzahl der Durchläufe: " + anzahl);
    }
}
```

Aufgabe 17. Notenschlüssel

```
import java.util.Scanner;
public class Notenschluessel {
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) {
        int note;
        System.out.print("Bitte Punkte eingeben: ");
        note = sc.nextInt();

        if (note < 50) {
            System.out.print("Sie sind negativ.");
        } else if (note >= 50 && note <= 62) {
            System.out.print("Genügend.");
        } else if (note >= 63 && note <= 75) {
            System.out.print("Befriedigend.");
        } else if (note >= 76 && note <= 88) {
            System.out.print("Gut.");
        } else {
            System.out.print("Sehr Gut.");
        }
    }
}
```

```
}
```

Aufgabe 18. Menüauswahl

```
import java.util.Scanner;

public class Menueauswahl {
    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {
        int menueauswahl;

        System.out.print("Bitte Menue eingeben: ");
        menueauswahl = sc.nextInt();

        switch (menueauswahl) {
            case 1:
                System.out.println("Zanderfilet in Paprikaschaum mit Kartoffeln und
Gemuese.");
                break;
            case 2:
                System.out.println("Schweinsmedallions in Pfefferrahmsauce mit Kroketten und
Gemuese.");
                break;
            case 3:
                System.out.println("Risotto mit frischen Steinpilzen (Vegetarisch).");
                break;
            case 4:
                System.out.println("Wienerschnitzel mit Pommes (Kinder).");
                break;
            default:
                System.out.println("Ungueltige Menueauswahl.");
        }
    }
}
```

Aufgabe 19. Potenzrechnen

```
import java.util.Scanner;
public class PotenzWhile {
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) {
        int basis, exponent;
        int ergebnis = 1;
        int anzahl = 0;
        System.out.print("Bitte Basis eingeben: ");
        basis = sc.nextInt();

        System.out.print("Bitte Exponent eingeben: ");
        exponent = sc.nextInt();
        while (anzahl < exponent) {
            ergebnis *= basis;
            anzahl++;
        }
        System.out.print("Das Ergebnis lautet: " + ergebnis);
    }
}

import java.util.Scanner;
public class PotenzDoWhile {
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) {
        int basis, exponent;
        int ergebnis = 1;
        int anzahl = 0;
        System.out.print("Bitte Basis eingeben: ");
        basis = sc.nextInt();
        System.out.print("Bitte Exponent eingeben: ");
        exponent = sc.nextInt();
        do {
```

```

        ergebnis *= basis;
        anzahl++;
    } while (anzahl < exponent);
    System.out.print("Das Ergebnis lautet: " + ergebnis);
}}

import java.util.Scanner;
public class PotenzFor {
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) {
        int basis, exponent;
        int ergebnis = 1;
        System.out.print("Bitte Basis eingeben: ");
        basis = sc.nextInt();
        System.out.print("Bitte Exponent eingeben: ");
        exponent = sc.nextInt();
        for (int anzahl = 0; anzahl < exponent; anzahl++) {
            ergebnis *= basis;
        }
        System.out.print("Das Ergebnis lautet: " + ergebnis);
    }
}

```

Aufgabe 20. Brutto-Netto-Rechner

```

import java.util.Scanner;

public class BruttoNetto {

    public static void main(String[] args) {

        double brutto = 0;
        double netto1 = 0; //fuer 10% USt
        double netto2 = 0; //fuer 20% USt
        double ust1 = 0;
        double ust2 = 0;
        Scanner in = new Scanner(System.in);

        System.out.println("Bitte geben Sie einen Bruttobetrag ein.");
        brutto = in.nextDouble();

        ust1 = (brutto/110)*10;
        ust2 = (brutto/120)*20;

        netto1 = brutto - ust1;
        netto2 = brutto - ust2;

        System.out.println("Bei 10% USt. betraegt der Nettobetrag " + netto1);
        System.out.println("Bei 10% USt. betraegt die USt " + ust1);
        System.out.println("Bei 20% USt. betraegt der Nettobetrag " + netto2);
        System.out.println("Bei 10% USt. betraegt die USt " + ust2);

        in.close();
    }
}

```

Aufgabe 21. Gesamtbetrag brutto

```

public class Rechnung {
    public static void main(String[] args) {
        int a = 50;
        int b = 20;
        int c = 30;
        int n = a+b+c;
        double u = n/100*20;
        double g = n+u;
        System.out.println("Artikel 1 = "+a);
        System.out.println("Artikel 2 = "+b);
    }
}

```

```

        System.out.println("Artikel 3 = "+c);
        System.out.println("Gesamtbetrag netto = "+n);
        System.out.println("20% Umsatzsteuer = "+u);
        System.out.println("Gesamtbetrag brutto = "+g);
    }
}

```

Aufgabe 22. Schleifen

1. Es berechnet die Gesamtpunkte und gibt abhängig davon aus, ob eine Prüfung oder Lehrveranstaltung bestanden oder nicht bestanden wurde.
2. Es können lediglich einmal weitere Punkte eingegeben werden, diese Funktion sollte mit einer Schleife gelöst werden. Wenn eine ungültige Punkteanzahl eingegeben wird, wird keine Fehlermeldung ausgegeben.

Aufgabe 23. Punkteberechnung

23a)

```

import java.util.*;
public class Punkte2 {
    static Scanner in = new Scanner(System.in);

    public static void main(String[] args) {
        int p=0, g=0 ;

        for(int i=0; i<4; i++){
            System.out.println("Punkte eingeben: ");
            p = in.nextInt();
            g = g + p;
        }
        System.out.println("Gesamtpunkte : "+g);
        if (g >50)
            System.out.println("Bestanden");
        else
            System.out.println("Leider nicht bestanden");
    }
}

```

23b)

```

public class Punkte5 {
    static Scanner in = new Scanner(System.in);

    public static void main(String[] args) {
        int p=0, g=0 ;
        System.out.println("Punkte eingeben bis Sie 0 oder -1 drücken");

        do {
            System.out.println("Punkte eingeben: ");
            p = in.nextInt();
            g = g + p;
        }
        while (p>0);

        System.out.println("Gesamtpunkte : "+g);
        if (g >50)
            System.out.println("Bestanden");
        else
            System.out.println("Leider nicht bestanden");
    }
}

```

23c)

```

public class Punkte6 {

```

```

static Scanner in = new Scanner(System.in);
public static void main(String[] args) {
    int p=0, g=0 ;

    for(int i=0; i<4; i++){
        System.out.println("Punkte eingeben: ");
        p = in.nextInt();
        g = g + p;
    }
    System.out.println("Gesamtpunkte : "+g);
    if (g>90){
        System.out.println("Mit sehr gutem Erfolg bestanden");
    }
    else if (g>80){
        System.out.println("Mit gutem Erfolg bestanden");
    }
    else if (g>50){
        System.out.println("Bestanden");
    }
    else {
        System.out.println("Leider nicht bestanden");
    }
}
}

```

Aufgabe 24. Nachhilfestunden verwalten

Nachhilfestunden

```

import java.util.*;

public class Nachhilfestunden {
    static Scanner in = new Scanner(System.in);

    public static void main(String[] args) {
        int s = 0;
        int gesamt = 0;
        System.out.println("Stunden eingeben:");
        s = in.nextInt();

        while (s > 0) {
            gesamt += s;
            System.out.println("Stunden eingeben:");
            s = in.nextInt();
        }

        System.out.println("Gesamtstunden: " + gesamt);
        // Ergänzung zu Aufgabe b) Einkommen
        System.out.println("Einkommen: " + gesamt * 20);
    }
}

```

Nachhilfestunden & Einkommen:

```

import java.util.*;

public class Nachhilfestunden {
    static Scanner in = new Scanner(System.in);
    /**
     * @param args
     */
    public static void main(String[] args) {
        int s=0;
        int gesamt=0;
        System.out.println("Stunden eingeben:");
        s=in.nextInt();
    }
}

```

```

        while (s>0){
gesamt += s;
System.out.println("Stunden eingeben:");
    s=in.nextInt();
    }

    System.out.println("Gesamtstunden: "+gesamt);
    //Ergänzung zu Aufgabe b) Einkommen
    System.out.println("Einkommen: " + gesamt*20);
    }
}

```

Statistik:

```

import java.util.Scanner;

public class Statistik {
static Scanner in = new Scanner(System.in);
/**
 * @param args
 */
public static void main(String[] args) {
    int s =0;// Stunden
    int t =0;//Tage
    int m = 0; //Monate
    int z = 0; //Zähler
    int gesamt=0;
    int min = Integer.MAX_VALUE;
    int max = Integer.MIN_VALUE;

    double d = 0.0; // Durchschnitt
System.out.println("Bitte geben Sie die Anzahl der Monate ein: ");
    m = in.nextInt();
    if(m >= 1){
        do{
            s= 0;
            t= 0;
            gesamt = 0;
            min = Integer.MAX_VALUE;
            max = Integer.MIN_VALUE;
            d= 0.0;

            System.out.println("Anzahl der Tage eingeben:");
            t=in.nextInt();

            for (int i=1; i<=t; i++){
                System.out.println("Stunden eingeben:");
                s=in.nextInt();
                gesamt += s;
                d = gesamt/t;
                if(s < min)
                    min = s;
                if(s > max)
                    max = s;
            }
            System.out.println("Monat " + z+1 +":");
System.out.println("-----");
            System.out.println("Gesamtstunden: "+gesamt);
            System.out.println("Durchschnitt: "+d);
            System.out.println("Minimale Stunden: " + min);
            System.out.println("Maximale Stunden: " + max);
            System.out.println();
            z = z+1;
        }while(z < m);
    }
}
}

```

Aufgabe 28. Puzzle – Binäre Suche

```
import java.util.Scanner;
public class Binaere_Suche {
    public static void main(String[] args) {
        int[] liste = {2, 3, 5, 6, 9, 10, 14, 16, 17, 19, 20, 22, 25, 30, 32, 35, 36,
37, 40};
        int suchen;
        int e;
        System.out.println("Welches Element möchtest du suchen?: ");
        Scanner sc = new Scanner(System.in);
        suchen = sc.nextInt();
        e = binaerSuchenIterativ(liste, suchen, 0, liste.length - 1);
        if (e == -1) {
            System.out.println("Element nicht gefunden.");
        } else {
            System.out.println("Position des Elements (startet bei 0): " + (e));
        }
    }

    public static int binaerSuchenIterativ(int[] liste, int suchen, int first, int
last) {
        boolean found = false;
        int m = -1;
        while(first <= last && found == false) {
            m = (first + last) / 2;
            if(suchen == liste[m]) {
                found = true;
            } else if(suchen < liste[m]) {
                last = m-1;
            } else {
                first = m+1;
            }
        }
        if(!found) {
            m = -1;
        }
        return m;
    }
}
```

Aufgabe 29. Arrays ausgeben (Lesecke)

```
String[] namen = new String[6];
namen[0] = "Anita";
namen[1] = "David";
namen[2] = "Mario";
namen[3] = "Bonifaz";
namen[4] = "Rudolf";
namen[5] = "Erich";
for (int i = 0; i < namen.length; i++) {
    System.out.println("Name an Stelle " + i + ": " + namen[i]);
}

double gleitkomma[] = { 1.1, 2.2, 3.3, 4.4, 5.5, };
for (int i = 0; i < gleitkomma.length; i++) {
    System.out.println("Zahl an Stelle " + i + ": " + gleitkomma[i]);
}

char[] zeichen = new char[5];
zeichen[0] = ':';
zeichen[1] = '/';
zeichen[2] = '&';
zeichen[3] = '!';
zeichen[4] = '?';
for (int i = 0; i < zeichen.length; i++) {
    System.out.println("Zeichen an Stelle " + i + ": " + zeichen[i]);
}

int zahl[] = new int[10];
for (int i = 0; i < zahl.length; i++) {
    zahl[i] = i;
}
}
```

```
for (int i = 0; i < zahl.length; i++) {
    System.out.println("Zahl an Stelle " + i + ": " + zahl[i]);
}
```

Aufgabe 30. Arrays ausgeben

Programm

```
public static void main(String[] args) {
    String[] Tage = {"Montag", "Dienstag", "Mittwoch", "Donnerstag", "Freitag",
    "Samstag", "Sonntag"};
    String[] Monate =
    {"Januar", "Februar", "März", "April", "Mai", "Juni", "Juli", "August", "September", "Oktobe
    r", "November", "Dezember"};
    int[] zeit = {8,9,10,11,12,13,14,15,16,17,18};
    String[] waehrung = {"€", "$", "SEK"};
    double[] einkommen = {0.0, 0.365, 0.432143, 0.5};
    double[] ust = {0.1, 0.2};
    int[] primzahlen = {1,2,3,5,7,11,13,17};
    System.out.println("Monat 5 = "+Monate[4]);
    System.out.println("waehrung = " +waehrung[0]);
    System.out.println("Zeit = "+zeit[0]+" Uhr");
    System.out.println("Einkommenssteuer Stufe 2 = "+ einkommen[1]);
    System.out.println("USt = "+ust[1]*10 +"%");
    System.out.println("Primzahlen bis 17: ");
    for (int i=0; i<primzahlen.length; i++){
        System.out.print(primzahlen[i]+" ");
    }

    System.out.println("Alle Tage: ");
    for (int i=0; i<Tage.length; i++){
        System.out.print(Tage[i]+" ");
    }
    System.out.println();
    System.out.println("5. Tag der Woche: "+Tage[4]);
}
}
```

Ausgabe

```
Monat 5 = Mai
waehrung = €
Zeit = 8 Uhr
Einkommenssteuer Stufe 2 = 0.365
USt = 2.0%
Primzahlen bis 17:
1, 2, 3, 5, 7, 11, 13, 17, Alle Tage:
Montag, Dienstag, Mittwoch, Donnerstag, Freitag, Samstag, Sonntag,
5. Tag der Woche: Freitag
```

Aufgabe 31. Array initialisieren mittels random()

```
public class Array2 {
    public static void main(String[] args) {

        int[] zahlen = new int[10];

        for(int i = 0; i < zahlen.length; i++) {
            zahlen[i] = (int) (Math.random() * 100);
        }

        for(int i = 0; i < zahlen.length; i++) {
            System.out.print(zahlen[i] + ", ");
        }
    }
}
```

Aufgabe 32: Ein kleiner Modulo Rechner

```
import java.util.Scanner;
public class Modulorechner {
    static Scanner sc = new Scanner(System.in);
```

```

public static void main(String[] args) {

    System.out.print("Wie viele Zahlen moechten Sie eingeben?: ");
    int x = sc.nextInt();

    int[] zahlen = new int[x];

    System.out.print("Mit welcher Zahl moechten Sie Modulo rechnen?: ");
    x = sc.nextInt();

    for(int i = 0; i < zahlen.length; i++) {
        System.out.print("Bitte " + (i+1) + ". Zahl eingeben: ");
        zahlen[i] = sc.nextInt();
    }

    System.out.println("Die berechneten Zahlen sind:");

    for(int i = 0; i < zahlen.length; i++) {
        System.out.print(zahlen[i] % x + ", ");
    }
}
}

```

Aufgabe 33: Aus 2 mach 1

```

public class Concatenation {

    public static void main(String[] args) {

        char[] baum = {'B', 'a', 'u', 'm'};
        char[] haus = {'h', 'a', 'u', 's'};

        char[] baumhaus = new char[baum.length + haus.length];

        for(int i = 0; i < baum.length; i++) {
            baumhaus[i] = baum[i];
        }

        for(int i = baumhaus.length -1; i >= haus.length; i--) {
            baumhaus[i] = haus[i-haus.length];
        }

        for(int i = 0; i < baumhaus.length; i++) {
            System.out.print(baumhaus[i]);
        }
    }
}

```

Aufgabe 34 bis Aufgabe 37: Unterprogramme

```

import java.util.Scanner;

public class Methoden {

    public static void main(String[] args) {

        int[] zahlen = {2,4,6,3,6,88,3,2435};
        int geleseneZahl = 0;
        int anzahl = 0;

        System.out.println("Das Menue wird ausgegeben:");
        menue();
        System.out.println("Zahlenarray wird ausgegeben:");
        printArray(zahlen);
        geleseneZahl = readInt();
        System.out.println("Die eingelesene Zahl ist die Zahl: " + geleseneZahl);
        anzahl = countInt(zahlen);
        System.out.println("Das oben initialisierte Array hat " + anzahl + "
        Elemente.");
    }

    public static void menue() { // Aufgabe 34
        System.out.println("1. Vokabel lernen");
        System.out.println("2. Grammatiktest");
    }
}

```

```

    System.out.println("3. Programm beenden");
}

public static void printArray(int[] zahlen) { // Aufgabe 35
    for(int i = 0; i < zahlen.length; i++) {
        System.out.print(zahlen[i] + ", ");
    }
    System.out.println("");
}

public static int readInt() { // Aufgabe 36
    Scanner sc = new Scanner(System.in);
    System.out.print("Bitte Zahl eingeben: ");
    int zahl = sc.nextInt();
    sc.close();

    return zahl;
}

public static int countInt(int[] zahlen) { // Aufgabe 37
    int anzahl = 0;
    for(int i = 0; i < zahlen.length; i++) {
        anzahl++;
    }

    return anzahl;
}
}

```

Aufgabe 38: Arrays

```

public class Ausgabe {

    public static void main(String[] args) {
        String[] Tage = {"Montag", "Dienstag", "Mittwoch", "Donnerstag", "Freitag", "Samstag",
            "Sonntag"};
        String[] Monate = {"Januar", "Februar", "März", "April", "Mai", "Juni", "Juli", "August",
            "September", "Oktober", "November", "Dezember"};
        int[] zeit = {8,9,10,11,12,13,14,15,16,17,18};
        String[] waehrung = {"€", "$", "SEK"};
        double[] einkommen = {0.0,0.365, 0.432143, 0.5};
        double[] ust = {0.1, 0.2};
        int[] primzahlen = {1,2,3,5,7,11,13,17};

        System.out.println("Monat 5 = "+Monate[4]);
        System.out.println("waehrung = " +waehrung[0]);
        System.out.println("Zeit = "+zeit[0]+" Uhr");
        System.out.println("Einkommenssteuer Stufe 2 = "+ einkommen[1]);
        System.out.println("USt = "+ust[1]*10 +"%");

        System.out.println("Primzahlen bis 17: ");
        for (int i=0; i<primzahlen.length; i++){
            System.out.print(primzahlen[i]+", ");
        }

        System.out.println("Alle Tage: ");
        for (int i=0; i<Tage.length; i++){
            System.out.print(Tage[i]+", ");
        }

        System.out.println();
        System.out.println("5. Tag der Woche: "+Tage[4]);
    }
}

```

Ausgabe

```

Monat 5 = Mai
waehrung = €
Zeit = 8 Uhr
Einkommenssteuer Stufe 2 = 0.365
USt = 2.0%
Primzahlen bis 17:

```

1, 2, 3, 5, 7, 11, 13, 17, Alle Tage:
Montag, Dienstag, Mittwoch, Donnerstag, Freitag, Samstag, Sonntag,
5. Tag der Woche: Freitag

Aufgabe 39. Rechnen

```
public class test {  
  
    public static void main(String[] args) {  
        Scanner in = new Scanner(System.in);  
        int[] array = new int[6];  
        int summe, produkt, summe1, summe2, min, max;  
        float div, mittel;  
        produkt = 1;  
        summe = 0;  
        summe1 = 0;  
        summe2 = 0;  
        min = Integer.MAX_VALUE;  
        max = Integer.MIN_VALUE;  
        System.out.println("Bitte geben Sie 6 Zahlen ein.");  
  
        for(int i = 0; i<array.length; i++){  
            array[i] = in.nextInt();  
        }  
        for(int i = 0; i<array.length; i++){  
            summe = summe + array[i];  
            produkt = produkt * array[i];  
            if(array[i] < min){  
                min = array[i];  
            }  
            if(array[i] > max){  
                max = array[i];  
            }  
            if(i <= 2){  
                summe1 = summe1 + array[i];  
            }else{  
                summe2 = summe2 + array[i];  
            }  
        }  
        div = summe1 / summe2;  
        mittel = summe/6;  
  
        System.out.println("Summe: " + summe);  
        System.out.println("Produkt: " + produkt);  
        System.out.println("Division: " + div);  
        System.out.println("Mittelwert: " + mittel);  
        System.out.println("Minimum: " + min);  
        System.out.println("Maximum: " + max);  
  
        in.close();  
    }  
}
```

Aufgabe 40. Punkteverwaltung - Menü

```
public class Punkteverwaltung {  
    static Scanner in = new Scanner(System.in);  
    public static void main(String[] args) {  
        String[] menue = {"1. Student hinzufügen", "2. Punkte ändern", "3. Punkte  
ausgeben", "4. Noten ausgeben"};  
        int o = 0;  
  
        for(int i=0; i<4; i++){  
            System.out.println(menue[i]);  
        }  
    }  
}
```

```

System.out.println("Wählen Sie eine Menüoption aus:");
o=in.nextInt();
switch(o){
case 1:
    System.out.println(menue[0]); break;
case 2:
    System.out.println(menue[1]); break;
case 3:
    System.out.println(menue[2]); break;
case 4:
    System.out.println(menue[3]); break;
}
}
}

```

Aufgabe 42. Puzzle Konjugierer

```

import java.util.Scanner;

public class Verbenkonjugierer {
    static Scanner in = new Scanner(System.in);
    static String[][] verben = {
        { "sein", "essere", "sono", "sei", "e", "siamo", "siete", "sono" },
        { "haben", "avere", "ho", "hai", "ha", "abbiamo", "avete", "hanno" },
        { "gehen", "andare", "vado", "vai", "va", "andiamo", "andate", "vanno" },
        { "essen", "mangiare", "mangio", "mangi", "mangia", "mangiamo", "mangiate",
"mangiano" },
        { "schlafen", "dormire", "dormo", "dormi", "dorme", "dormiamo", "dormite",
"dormono" } };

    public static void main(String[] args) {

        String input;
        int zahl;
        System.out.print("Bitte Verb eingeben: ");
        input = in.next();
        System.out.println("Für gewünschte Person bitte Zahl eingeben:");
        System.out.println("1 für Nennform");
        System.out.println("2 für 1. Person EZ");
        System.out.println("3 für 2. Person EZ");
        System.out.println("4 für 3. Person EZ");
        System.out.println("5 für 1. Person MZ");
        System.out.println("6 für 2. Person MZ");
        System.out.println("7 für 3. Person MZ");
        System.out.println("0 für alle Personen");
        zahl = in.nextInt();

        if (zahl > 7 || zahl < 0) {
            System.out.println("ungueltige Eingabe!");
        } else {
            konjugierer(input, zahl);
        }
    }

    public static void konjugierer(String input, int zahl) {
        for (int i = 0; i < verben.length; i++) {
            if (verben[i][0].equals(input) && zahl == 0) {
                for (int j = 0; j < verben[0].length; j++) {
                    System.out.print(verben[i][j] + ", ");
                }
            } else if (verben[i][0].equals(input)) {
                System.out.println(verben[i][zahl]);
            }
        }
    }
}

```

Aufgabe 43. Bingospiel

```
import java.util.Scanner;

public class Bingo {

    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {
        int[][] bingoschein = new int[3][3];

        System.out.println("Sie haben einen Bingoschein mit 3 Zeilen und 3 Spalten.");

        for(int zeile = 0; zeile < bingoschein.length; zeile++) {
            for(int spalte = 0; spalte < bingoschein[zeile].length; spalte++) {
                System.out.print("Bitte Nummer fuer Zeile " + (zeile+1) + " und
                Spalte " + (spalte+1) + " eingeben: ");
                bingoschein[zeile][spalte] = sc.nextInt();
            }
        }
        printBingoschein(bingoschein);
    }

    public static void printBingoschein(int bingo[][]) {
        System.out.print("Ihr Bingoschein:");

        for(int i = 0; i < bingo.length; i++) {
            System.out.print("\n");
            for(int j = 0; j < bingo[i].length; j++) {
                System.out.print(bingo[i][j] + " ");
            }
        }
    }
}
```

Ausgabe:

```
Sie haben einen Bingoschein mit 3 Zeilen und 3 Spalten.
Bitte Nummer fuer Zeile 1 und Spalte 1 eingeben: 35
Bitte Nummer fuer Zeile 1 und Spalte 2 eingeben: 75
Bitte Nummer fuer Zeile 1 und Spalte 3 eingeben: 98
Bitte Nummer fuer Zeile 2 und Spalte 1 eingeben: 13
Bitte Nummer fuer Zeile 2 und Spalte 2 eingeben: 43
Bitte Nummer fuer Zeile 2 und Spalte 3 eingeben: 76
Bitte Nummer fuer Zeile 3 und Spalte 1 eingeben: 98
Bitte Nummer fuer Zeile 3 und Spalte 2 eingeben: 11
Bitte Nummer fuer Zeile 3 und Spalte 3 eingeben: 25
Ihr Bingoschein:
35 75 98
13 43 76
98 11 25
```

Aufgabe 44. Namenstabelle

```
import java.util.Scanner;

public class Namenstabelle {

    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {
        System.out.print("Wie viele Personen wollen Sie erfassen?: ");
        String[][] tabelle = new String[sc.nextInt()][2];

        for(int i = 0; i < tabelle.length; i++) {
            System.out.print("Bitte " + (i + 1) + ". Vornamen eingeben: ");
            tabelle[i][0] = sc.next();
            System.out.print("Bitte " + (i + 1) + ". Nachnamen eingeben: ");
            tabelle[i][1] = sc.next();
        }

        printTabelle(tabelle);
    }

    public static void printTabelle(String[][] tabelle) {
        System.out.println("\nVorname\t\tNachname");
    }
}
```

```

for(int i = 0; i < tabelle.length; i++) {
    System.out.print("\n");
    for(int j = 0; j < tabelle[i].length; j++) {
        System.out.print(tabelle[i][j] + "\t\t");
    }
}
}
}

```

Beispiel Ausgabe:

```

Wie viele Personen wollen Sie erfassen?: 4
Bitte 1. Vornamen eingeben: Melanie
Bitte 1. Nachnamen eingeben: Maier
Bitte 2. Vornamen eingeben: Stefan
Bitte 2. Nachnamen eingeben: Huber
Bitte 3. Vornamen eingeben: Paul
Bitte 3. Nachnamen eingeben: Sonntag
Bitte 4. Vornamen eingeben: Andrea
Bitte 4. Nachnamen eingeben: Muster

```

Vorname	Nachname
Melanie	Maier
Stefan	Huber
Paul	Sonntag
Andrea	Muster

Aufgabe 45. Glossar

```

public class Glossar {
    static String[][] glossar = {"array", "Zusammenfassung gleichartiger Komponenten
zu einem Datenaggregat" },
    {"variable", "Die Daten des Programms werden in Variablen gespeichert."},
    {"funktion", "Unterprogramm, welches beim Rücksprung einen Wert an das aufrufende
Programm zurücliefert. "}};

    /**
     * @param args
     */
    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner scanner = new Scanner (System.in);
        System.out.println("Welches Wort möchten Sie nachschlagen?");
        String wort;
        wort = scanner.next();
        for(int i = 0; i < glossar.length; i++){
            if(glossar[i][0].equalsIgnoreCase(wort)){
                System.out.println(glossar[i][1]);
            }
        }
    }
}

```

Aufgabe 46. Währung

```

public static void main(String[] args) {
    // TODO Auto-generated method stub
    int from;
    int to;
    float i;
    Scanner scanner = new Scanner(System.in);
    System.out.println("Aus welcher Währung möchten Sie umrechnen? 0=Schilling,
1=Euro, 2=Dollar");
    from = scanner.nextInt();
    System.out.println("In welche Währung möchten Sie umrechnen? 0=Schilling,
1=Euro, 2=Dollar");
    to = scanner.nextInt();
    System.out.println("Welchen Betrag möchten Sie umrechnen?");
    i = scanner.nextFloat();

    if(from==0){
        if(to==1){
            System.out.println(i/13.76);

```

```

        }else if(to == 2){
            System.out.println(i/10.83);
        }
    }else if(from == 1){
        if(to==0){
            System.out.println(i*13.76);
        }else if(to == 2){
            System.out.println(i*1.27);
        }
    }else if(from == 2){
        if(to==0){
            System.out.println(i*10.83);
        }else if(to == 1){
            System.out.println(i/1.27);
        }
    }else{
        System.out.println("Fehlerhafte Eingabe");
    }
}
}

```

Aufgabe 47. Übung Partyservice

```

public class Partyservice {

    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {

        System.out.print("Wie viele Leute kommen zu Ihrer Party?: ");
        int anzahlPersonen = sc.nextInt();

        if (anzahlPersonen > 340) {
            System.out.println("Dieser Partyservice kann maximal fuer 340 Personen liefern");
        } else if(anzahlPersonen <= 0) {
            System.out.println("Ihre Party ist gratis. Juhu!");
        } else {
            double preis = 0.0;
            System.out.print("In welchem Monat findet die Party statt (1-12)? ");
            int monat = sc.nextInt();

            if (anzahlPersonen < 20) {
                preis = 500;
            } else if (anzahlPersonen < 100) {
                preis = 4000;
            } else if (anzahlPersonen < 340) {
                preis = 12000;
            }

            if(monat == 7) {
                preis = preis * 1.1;
            } else if(monat == 8) {
                preis = preis * 1.15;
            } else if(monat == 11) {
                preis = preis * 0.88;
            } else if(monat == 12) {
                preis = preis * 0.82;
            }

            System.out.println("Diese Party kostet : " + preis + " Euro.");
        }
    }
}

```

Aufgabe 49. Puzzle

1. Methode prüft das Ende eines Strings d
2. Methode liefert ein Zeichen eines Strings b
3. Methode liefert eine Zeichensequenz eines Strings a
4. Methode prüft, ob ein String mit einer speziellen Zeichenkette beginnt c

Aufgabe 50. Array einlesen und ausgeben 1

```
import java.util.Scanner;
public class Array1 {

    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {

        System.out.print("Wie viele Woerter moechten Sie eingeben?: ");
        int anzahl = sc.nextInt();

        String[] woerter = new String[anzahl];

        for(int i = 0; i < woerter.length; i++) {
            System.out.print("Bitte " + (i+1) + ". Wort eingeben: ");
            woerter[i] = sc.next();
        }

        System.out.println("Sie haben folgende Woerter eingegeben:");

        for(int i = 0; i < woerter.length; i++) {
            System.out.print(woerter[i] + ", ");
        }
    }
}
```

Aufgabe 51. Array einlesen und ausgeben 2

Allgemeiner Code - Programmanfang

```
public class Wortratespiel {

    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {

        System.out.print("Bitte Ratewort eingeben: ");
        String wort = sc.next();
        String vorschlag = "";
```

While Schleife

```
while (!vorschlag.equalsIgnoreCase("ende")) {
    System.out.print("Raten Sie! (zum Beenden das Wort ende eingeben): ");
    vorschlag = sc.next();

    if (wort.equalsIgnoreCase(vorschlag)) {
        System.out.print("Congratulations! Noch eine Runde? ja/nein eingeben: ");
        vorschlag = sc.next();
        if (vorschlag.equalsIgnoreCase("nein")) {
            vorschlag = "ende";
        } else {
            System.out.print("Bitte Ratewort eingeben: ");
            wort = sc.next();
        }
    } else if (!vorschlag.equalsIgnoreCase("ende")) {
        System.out.println("Leider falsch!");
    }
}
```

```
}  
}
```

Do-while Schleife

```
/* do {  
    System.out.print("Raten Sie! (zum Beenden das Wort ende eingeben): ");  
    vorschlag = sc.next();  
  
    if (wort.equalsIgnoreCase(vorschlag)) {  
        System.out.print("Congratulations! Noch eine Runde? ja/nein eingeben: ");  
        vorschlag = sc.next();  
        if (vorschlag.equalsIgnoreCase("nein")) {  
            vorschlag = "ende";  
        } else {  
            System.out.print("Bitte Ratewort eingeben: ");  
            wort = sc.next();  
        }  
    } else if (!vorschlag.equalsIgnoreCase("ende")) {  
        System.out.println("Leider falsch!");  
    }  
} while (!vorschlag.equalsIgnoreCase("ende"));*/
```

For Schleife

```
/* for (;!vorschlag.equalsIgnoreCase("ende");) {  
    System.out.print("Raten Sie! (zum Beenden das Wort ende eingeben): ");  
    vorschlag = sc.next();  
  
    if (wort.equalsIgnoreCase(vorschlag)) {  
        System.out.print("Congratulations! Noch eine Runde? ja/nein eingeben: ");  
        vorschlag = sc.next();  
        if (vorschlag.equalsIgnoreCase("nein")) {  
            vorschlag = "ende";  
        } else {  
            System.out.print("Bitte Ratewort eingeben: ");  
            wort = sc.next();  
        }  
    } else if (!vorschlag.equalsIgnoreCase("ende")) {  
        System.out.println("Leider falsch!");  
    }  
}*/
```

Nochmals allgemeiner Code – Programmende

```
    System.out.println("goodbye");  
}  
}
```

Aufgabe 52. Menü

```
public static void main(String[] args) {  
    String[] menue = {"1. Wörterbuch", "2. Wortschatz lernen", "3. Grammatik",  
    "4. Ende"};  
    int o = 0;  
  
    for (int i=0; i<4; i++){  
        System.out.println(menue[i]);  
    }  
    System.out.println("Wählen Sie eine Menüoption aus:");  
    o=in.nextInt();  
  
    if (o>0 && o<5)  
        switch(o){  
            case 1:  
                System.out.println(menue[0]); break;
```

```

case 2:
    System.out.println(menue[1]); break;
case 3:
    System.out.println(menue[2]); break;
case 4:
    System.out.println(menue[3]); break;
}
else
    System.out.println("Achtung! Ungültige Eingabe!");
}

```

Aufgabe 53. Geschüttelte Wörter

```

import java.util.*;

public class Vokabelspiel1 {
    static Scanner in = new Scanner(System.in);

    //AB2, 4.7.2 Geschüttelte Wörter
    public static void main(String[] args) {
        String wort;
        System.out.println("Bitte Wort eingeben: ");
        wort = in.next();
        schuettern(wort);
    }
    public static String schuettern(String wort) {

        char[] vertauschen = wort.toCharArray();
        wort = "";
        char temp;

        for (int i = 0; i < vertauschen.length - 1; i += 2) {
            temp = vertauschen[i];
            vertauschen[i] = vertauschen[i + 1];
            vertauschen[i + 1] = temp;
        }

        for (int i = 0; i < vertauschen.length; i++) {
            wort = wort + vertauschen[i];
        }
        System.out.println(wort);
        return wort;
    }
}

```

Aufgabe 54. Wörterbuch

```

static String[][] vocab = {
    { "language", "sprache", "lingua" },
    { "house", "haus", "casa" },
    { "mother", "mutter", "madre" },
    { "garden", "garten", "giardino" },
    { "cat", "katze", "gatto" }
};

public static void main(String[] args) {
    String result, word;
    int input, ausgangssprache, zielsprache;
    Scanner sc = new Scanner(System.in);
}

```

```

do {
    System.out.print("Drücken Sie 0 für Englisch, 1 für Deutsch, 2 für
Italienisch als Ausgangssprache (oder -1 um das Programm zu beenden): ");
    input = sc.nextInt();
    ausgangssprache = input;
    if(input != -1) {
        System.out.print("Drücken Sie 0 für Englisch, 1 für Deutsch, 2 für
Italienisch als Zielsprache (oder -1 um das Programm zu beenden): ");
        input = sc.nextInt();
        zielsprache = input;
        if(input != -1) {
            System.out.print("Geben Sie das Wort ein, welches Sie übersetzen
wollen: ");
            word = sc.next();
            result = translate(word,ausgangssprache,zielsprache);
            if(result.compareTo("") == 0) {
                System.out.println("Das Wort konnte leider nicht gefunden
werden.");
            } else {
                System.out.println("Die Übersetzung lautet: " + result);
            }
        }
    }
} while(input != -1);
}

public static String translate(String word, int from, int to) {
    String result = "";
    int i = 0;
    while (i < vocab.length && result.compareTo("") == 0) {
        if (vocab[i][from].compareTo(word) == 0) {
            result = vocab[i][to];
        } else {
            i++;
        }
    }
    return result;
}

```

Aufgabe 55. Sätze schütteln

```

private static void startShakeWords(String language) {
    String[] tmp = null;
    int points = 0;
    String[] shake_words = readFileContentAsArr(language.toLowerCase() +
".txt");
    boolean goOn = true;

    for(int i = 1; i < shake_words.length && goOn; i++) {
        tmp = shake_words[i].split(" ");
        Arrays.sort(tmp);
        System.out.println("Geschüttelter Satz: ");
        for(String s : tmp) {
            System.out.print(s + " ");
        }
        System.out.println("\n\nBitte richtigen Satz eingeben (Case
Sensitive): ");

        String word = readLine();
        if(word.toLowerCase().equals(shake_words[i].toLowerCase())) {

```

```

        points++;
    }

    System.out.println("Weitermachen? (Ja/Nein)");

    word = readLine();

    if(word.toLowerCase().equals("Nein".toLowerCase())) {
        goOn = false;
    }
}
System.out.println("Du hast " + points + " Sätze richtig!\n\n\n");
}

```

Aufgabe 56. Lückentextgenerator jedes 7. Wort streichen

```

private static void lueckentext() {
    boolean again = true; // wird als Indikator für die Option "Zurück"
    verwendet.
    String language = null;
    GRAMMAR_TYPE grammarType = null;
    boolean artikel_ersetzen = false;
    boolean plural_ersetzen = false;

    do {
        System.out.println("Gewählte Sprache: " + language);
        System.out.println("Artikel ersetzen: " + artikel_ersetzen);
        System.out.println("Plural ersetzen: " + plural_ersetzen);

        System.out.println(LUECKENTEXT_SUBMENU);
        int menuoption = in.nextInt();

        switch(menuoption) {
            case 1: language = getLanguageForShakingWords(); break;
            case 2: artikel_ersetzen = !artikel_ersetzen; break;
            case 3: if(language != null &&
language.toLowerCase().equals("italienisch".toLowerCase())) {
plural_ersetzen = !plural_ersetzen; } else { System.out.println("Bitte
zuerst Italienisch auswählen"); } break;
            case 4: if(language != null) {startLueckentext(language,
artikel_ersetzen, plural_ersetzen);} else {System.out.println("Bitte zuerst
eine Sprache wählen");} break;
            case 5: again = false; break;
            default: System.out.println("Falsche Eingabe, bitte erneut
eingeben: "); break;
        }
    } while(again);
}

/**
 * Diese Prozedur startet das LUEckentext-Spiel.
 */
private static void startLueckentext(String language, boolean
artikel_ersetzen, boolean plural_ersetzen) {
    String[] tmp = null;
    int points = 0;
    String[] shake_words = readFileContentAsArr(language.toLowerCase() +
".txt");
    boolean goOn = true;
}

```

```

    for(int i = 1; i < shake_words.length && goOn; i++) {
        tmp = shake_words[i].split(" ");
        System.out.println("LUEckentext: ");
        for(int j = 0; j < tmp.length; j++) {
            if((j+1) % 7 == 0) {
                System.out.print("? ");
            } else {
                if(artikel_ersetzen) {
                    if(tmp[j].toLowerCase().equals("der") ||
tmp[j].toLowerCase().equals("die") || tmp[j].toLowerCase().equals("das") ||
tmp[j].toLowerCase().equals("the") ||
                    tmp[j].toLowerCase().equals("il") ||
tmp[j].toLowerCase().equals("la") || tmp[j].toLowerCase().equals("lo")
||tmp[j].toLowerCase().equals("l") ||
                    tmp[j].toLowerCase().equals("i")
||tmp[j].toLowerCase().equals("gli") ||tmp[j].toLowerCase().equals("le")) {
                        tmp[j] = "?";
                    }
                }
                if(plural_ersetzen) {
                    if(tmp[j].toLowerCase().endsWith("i") ||
tmp[j].toLowerCase().endsWith("e")) {
                        tmp[j] = tmp[j].substring(0, tmp[j].length()-1) + "?";
                    }
                }
                System.out.print(tmp[j] + " ");
            }
        }
        System.out.println("\n\nBitte richtigen Satz eingeben: ");

        String word = readLine();

        if(word.toLowerCase().equals(shake_words[i].toLowerCase())) {
            points++;
        }

        System.out.println("Weitermachen? (Ja/Nein)");

        word = readLine();

        if(word.toLowerCase().equals("Nein".toLowerCase())) {
            goOn = false;
        }
    }

    System.out.println("Du hast " + points + " Saetze richtig!\n\n\n");
}

/**
 * liest eine Zeile ein, und kUEmmert sich ums flushen der Eingabe
 * @return
 */
private static String readLine() {
    String flushString = in.nextLine(); // muss eingebaut werden, da
\r\n vom vorherigen Eintrag noch vorhanden ist
    // dies muss rausgenommen werden

    String word = "";
    if(flushString.toCharArray().length == 0) { // Ueberpruefung ob es
wirklich ein \r\n gab

```

```

    word = in.nextLine();           // falls ja, neu einlesen
} else {
    word = flushString;           // sonst nimm den momentanen Wert
}
return word;
}

```

Aufgabe 57. Endungen streichen (Plural), Eingabe überprüfen

```

private static void startLueckentext(String language, boolean artikel_ersetzen,
boolean plural_ersetzen) {
    String[] tmp = null;
    int points = 0;
    String[] shake_words = readFileContentAsArr(language.toLowerCase() + ".txt");
    boolean goOn = true;

    for(int i = 1; i < shake_words.length && goOn; i++) {
        tmp = shake_words[i].split(" ");
        System.out.println("Lückentext: ");
        for(int j = 0; j < tmp.length; j++) {
            if((j+1) % 7 == 0) {
                System.out.print("? ");
            } else {
                if(artikel_ersetzen) {
                    if(tmp[j].toLowerCase().equals("der") ||
tmp[j].toLowerCase().equals("die") || tmp[j].toLowerCase().equals("das") ||
tmp[j].toLowerCase().equals("the") ||
                    tmp[j].toLowerCase().equals("il") ||
tmp[j].toLowerCase().equals("la") || tmp[j].toLowerCase().equals("lo")
||tmp[j].toLowerCase().equals("l'") ||
                    tmp[j].toLowerCase().equals("i")
||tmp[j].toLowerCase().equals("gli") ||tmp[j].toLowerCase().equals("le")) {
                        tmp[j] = "?";
                    }
                }
                if(plural_ersetzen) {
                    if(tmp[j].toLowerCase().endsWith("i") ||
tmp[j].toLowerCase().endsWith("e")) {
                        tmp[j] = tmp[j].substring(0, tmp[j].length()-1) + "?";
                    }
                }
                System.out.print(tmp[j] + " ");
            }
        }
        System.out.println("\n\nBitte richtigen Satz eingeben: ");

        String word = readLine();

        if(word.toLowerCase().equals(shake_words[i].toLowerCase())) {
            points++;
        }

        System.out.println("Weitermachen? (Ja/Nein)");

        word = readLine();

        if(word.toLowerCase().equals("Nein".toLowerCase())) {
            goOn = false;
        }
    }
    System.out.println("Du hast " + points + " Sätze richtig!\n\n\n");
}

```

Aufgabe 58. Groß-/Kleinschreibung

```

private static void startGrossUndKleinschreibung(String language) {
    String[] tmp = null;
    int points = 0;
    String[] shake_words = readFileContentAsArr(language.toLowerCase() + ".txt");
    boolean goOn = true;

    for(int i = 1; i < shake_words.length && goOn; i++) {
        System.out.println(shake_words[i].toLowerCase());
        System.out.println("\n\nBitte richtigen Satz eingeben: ");

        String word = readLine();

        if(word.equals(shake_words[i])) {
            System.out.println("Super");
            points++;
        } else {
            System.out.println("Leider falsch");
            System.out.println("Richtig wäre: " + shake_words[i]);
        }
        System.out.println("Weitermachen? (Ja/Nein)");
        word = readLine();

        if(word.toLowerCase().equals("Nein".toLowerCase())) {
            goOn = false;
        }
    }
    System.out.println("Du hast " + points + " Sätze richtig!\n\n\n");
}

```

Aufgabe 59. Übung Wortratespiel

Allgemeiner Code - Programmanfang

```

public class Wortratespiel {

    static Scanner sc = new Scanner(System.in);

    public static void main(String[] args) {

        System.out.print("Bitte Ratewort eingeben: ");
        String wort = sc.next();
        String vorschlag = "";

```

While Schleife

```

while (!vorschlag.equalsIgnoreCase("ende")) {
    System.out.print("Raten Sie! (zum Beenden das Wort ende eingeben): ");
    vorschlag = sc.next();

    if (wort.equalsIgnoreCase(vorschlag)) {
        System.out.print("Congratulations! Noch eine Runde? ja/nein eingeben: ");
        vorschlag = sc.next();
        if (vorschlag.equalsIgnoreCase("nein")) {
            vorschlag = "ende";
        } else {
            System.out.print("Bitte Ratewort eingeben: ");
            wort = sc.next();
        }
    } else if (!vorschlag.equalsIgnoreCase("ende")) {
        System.out.println("Leider falsch!");
    }
}

```

Do-while Schleife

```

/* do {
    System.out.print("Raten Sie! (zum Beenden das Wort ende eingeben): ");

```

```

vorschlag = sc.next();

if (wort.equalsIgnoreCase(vorschlag)) {
    System.out.print("Congratulations! Noch eine Runde? ja/nein eingeben: ");
    vorschlag = sc.next();
    if (vorschlag.equalsIgnoreCase("nein")) {
        vorschlag = "ende";
    } else {
        System.out.print("Bitte Ratewort eingeben: ");
        wort = sc.next();
    }
} else if (!vorschlag.equalsIgnoreCase("ende")) {
    System.out.println("Leider falsch!");
}
} while (!vorschlag.equalsIgnoreCase("ende"));*/

```

For Schleife

```

/* for (;!vorschlag.equalsIgnoreCase("ende");) {
    System.out.print("Raten Sie! (zum Beenden das Wort ende eingeben): ");
    vorschlag = sc.next();

    if (wort.equalsIgnoreCase(vorschlag)) {
        System.out.print("Congratulations! Noch eine Runde? ja/nein eingeben: ");
        vorschlag = sc.next();
        if (vorschlag.equalsIgnoreCase("nein")) {
            vorschlag = "ende";
        } else {
            System.out.print("Bitte Ratewort eingeben: ");
            wort = sc.next();
        }
    } else if (!vorschlag.equalsIgnoreCase("ende")) {
        System.out.println("Leider falsch!");
    }
}*/

```

Nochmals allgemeiner Code – Programmende

```

    System.out.println("goodbye");
}
}

```

Einleitung	1
1 Algorithmen und Datentypen	2
A Entdecken	2
A 1 Lesecke 1: Darstellung von Algorithmen	2
Aufgabe 1. Lesen und Verstehen	3
A 2 Lesecke 2: Programmcode verstehen	3
Aufgabe 2. Programmcode verstehen	3
Aufgabe 3. Merken	3
Aufgabe 4. Nachmachen	4
Aufgabe 5. Puzzle Datentypen	4
B Üben	5
Aufgabe 6. Einfache Rechnungen	5
Aufgabe 7. Datentypen	5
Aufgabe 8. Einkommen	6
Aufgabe 9. Nachhilfe	6
Aufgabe 10. Algorithmen & Co	6
Aufgabe 11. Datentypen	7
Aufgabe 12. Taschenrechner	7
2 Einfache Programme	9
A Entdecken	9
A 1 Lesecke Einfache Programme	9
A 2 Lesecke Schleifen	10
A 3 Lesecke Verzweigungen	11
A 4 Lesecke Schleifen	11
Aufgabe 13. Lesen und Verstehen	11
Aufgabe 14. Merken	12
Aufgabe 15. Nachmachen	12
B Üben	12
Aufgabe 16. Schleifen	12
Aufgabe 17. Notenschlüssel	12
Aufgabe 18. Menüauswahl	12
Aufgabe 19. Potenzrechnen	13
Aufgabe 20. Brutto-Netto-Rechner	13
Aufgabe 21. Gesamtbetrag brutto	13
Aufgabe 22. Schleifen	13
C Programmieren	13

Aufgabe 23. Punkteberechnung _____	13
Aufgabe 24. Nachhilfestunden verwalten _____	14
3 Eindimensionale Arrays _____	15
A Entdecken _____	15
A 1 Lesecke Arrays _____	15
Aufgabe 25. Lesen und Verstehen _____	16
Aufgabe 26. Merken _____	17
Aufgabe 27. Nachmachen _____	17
A 2 Schritt-für-Schritt: Eindimensionale Arrays _____	17
A 3 Schritt-für-Schritt: Ein Bibliotheksprogramm _____	17
Aufgabe 28. Puzzle – Binäre Suche _____	19
B Üben _____	20
Aufgabe 29. Arrays ausgeben (Lesecke) _____	20
Aufgabe 30. Arrays ausgeben _____	20
Aufgabe 31. Array initialisieren mittels random() _____	20
Aufgabe 32. Ein kleiner Modulo Rechner _____	20
Aufgabe 33. Aus 2 mach 1 _____	20
Aufgabe 34. Unterprogramm ohne Parameter und ohne Rückgabe _____	21
Aufgabe 35. Unterprogramm mit Parameter und ohne Rückgabe _____	21
Aufgabe 36. Unterprogramm ohne Parameter und mit Rückgabe _____	21
Aufgabe 37. Unterprogramm mit Parameter und mit Rückgabe _____	21
Aufgabe 38. Arrays _____	21
C Programmieren _____	21
Aufgabe 39. Rechnen _____	21
Aufgabe 40. Punkteverwaltung - Menü _____	21
4 Mehrdimensionale Arrays _____	23
A Entdecken _____	23
A 1 Lesecke Kyrillische Tastatur _____	23
A 2 Lesecke Wochenstunden _____	24
Aufgabe 41. Fragen und Aufgaben zur Lesecke _____	25
A 3 Schritt-für-Schritt: Mehrdimensionale Arrays _____	25
Aufgabe 42. Puzzle Konjugierer _____	25
B Üben _____	27
Aufgabe 43. Bingospiel _____	27
Aufgabe 44. Namenstabelle _____	27
C Programmieren _____	27

	Aufgabe 45. Glossar _____	27
	Aufgabe 46. Wahrung _____	28
	Aufgabe 47. bung Partyservice _____	28
5	Strings _____	29
	A Entdecken _____	29
	A 1 Lesecke Stringbeispiele _____	29
	Aufgabe 48. Fragen und Aufgaben zur Lesecke _____	29
	A 2 Puzzle _____	30
	Aufgabe 49. Puzzle _____	30
	B ben _____	31
	Aufgabe 50. Array einlesen und ausgeben 1 _____	31
	C Programmieren _____	31
	Aufgabe 51. Array einlesen und ausgeben 2 _____	31
6	Projekt – Sprachlernprogramm _____	32
	Aufgabe 52. Men (Arrays) _____	32
	Aufgabe 53. Vokabellernspiel „Geschttelte Wrter“ (Strings) _____	32
	Aufgabe 54. Wrterbuch _____	33
	Aufgabe 55. Stze schtteln _____	33
	Aufgabe 56. Lckentextgenerator jedes 7. Wort streichen _____	33
	Aufgabe 57. Endungen streichen (Plural), Eingabe berprfen _____	33
	Aufgabe 58. Gro-/Kleinschreibung _____	34
	Aufgabe 59. bung Wortratespiel _____	34
7	Lsungen _____	35