```
class Testprogramm {                                          // Die Quellcode-Datei sollte wie die
                                                              // Klasse heißen (hier Testprogramm.java)
  public static void main (String [] args) {                  // Aufruf der main-Methode

    int i,j,k,l;                                              // Deklarationen
    double dezimalzahl = 2.3;                                 // Deklaration mit Wertzuweisung
    int z = Integer.valueOf(args[0]).intValue();              // Umwandlung von als String übergebenen
    double d = Double.valueOf(args[1]).doubleValue();         // Parametern in int-/double-Variable
    final double PI = 3.1415927;                              // Deklaration einer Konstanten

    i++; j+=5;                                                // erhöht i um 1 und j um 5

    System.out.println("Text");                               // gibt den Text auf dem Bildschirm aus
    int parameterzahl = args.length;                          // enthält Anzahl der übergeb. Parameter

    l = i < j ? x : y                                         // bedingter Ausdruck

    if (i==0) {                                               // von einem Ausdruck abhängige Auswahl
      System.out.println("Bedingung erfuellt"); }             // Anweisungen, wenn Ausdruck erfüllt
    else {
      System.out.println("sonst"); }                         // Anweisungen in jedem anderen Fall

    if (i==0) {                                               // von einem Ausdruck abhängige Auswahl
      System.out.println("erste Bedingung erfuellt"); }       // Anweisungen, wenn 1.Ausdruck erfüllt
    else if (i==1) {                                          // Vergleichsoperator: ==, Zuweisungen: =
      System.out.println("zweite Bedingung erfuellt"); }      // Anweisungen, wenn 2.Ausdruck erfüllt
    else {
      System.out.println("Bedingung nicht erfuellt"); }       // Anweisungen in jedem anderen Fall

    switch (j) {                                              // Mehrfachauswahl für einen Ausdruck
      case 0: System.out.println("nur 1 Anweisung"); break;   // für jeden Fall gibt es Anweisungen, die
      case 1: System.out.println(j); break;                   // nicht von {} eingeschlossen werden. Der
      case 2: j++;                                            // Fall wird mit break abgeschlossen, weil
      case 3: System.out.println ("kein Break fuer j=2");     // sonst alle Anweisungen der noch folgen-
      default: System.out.println ("falls j nicht 0,1,2,3");  // den Fälle auch noch ausgeführt werden.
    }                                                         // default ist für alle anderen Fälle
```

```
    while (j<20) {                                          // kopfgesteuerte Schleife, die die Anwei-
      j++;                                                  // sungen im Schleifenkörper ausführt,
      System.out.println ("Der Wert von j ist jetzt "+j);   // solange die Bedingung erfüllt ist
    }                                                       // Achtung vor Endlosschleifen!

    for (int zaehler=1; zaehler<=10; zaehler++) {           // Schleife mit im Kopf vorbestimmter An-
      System.out.print ("*");                               // zahl von Durchläufen. Kopf: Initiali-
      if (zaehler % 5 == 0) break;  //Anweisung zum        // sierung der Laufvariable, Bedingung
    }                               //vorzeitigen Ausstieg  // und Aktualisierungsausdruck

    String[] tiere;                                         // Deklaration eines Arrays
    tiere = new String[5];                                  // Instanziierung des Arrays mit 5 Plätzen
    int [] primzahlen = {2,3,5,7,11,13};                    // Deklaration und Initialisierung
    tiere[3] = "Elefant";                                   // Ansprechen eines Arrayelements
    for (int a=0;a<5;a++) { System.out.println (tiere[i]); } // Ansprechen aller Arrayelemente

    ausgabe("Aufruf einer Methode vom Typ void");           // Methodenaufruf; die Argumente müssen
                                                            // vom Typ mit den im Methodenkopf ange-
                                                            // gebenen Variablen übereinstimmen
    k = bhochn (2,5);                                       // Methodenaufruf mit Rückgabe; Rückgabe-
                                                            // typ der Methode und Typ der zugewie-
                                                            // senen Variable müssen übereinstimmen

  }                                                         // Ende der Methode main

  static void ausgabe (String text) {                       // benutzerdef. Methode: es werden Variab-
    System.out.println (text);                              // le übergeben, verarbeitet und ein Wert
  }                                                         // zurückgegeben (Ausnahme: Typ void)

  static int bhochn (int basis, int exponent) {             // Methode vom Rückgabetyp int; werden
    int ergebnis=1;                                         // mehrere Variable übergeben, muß jede im
    for (int a=0;a<exponent;a++) {                          // Methodenkopf mit Typ deklariert werden
      ergebnis *= basis;                                    // Die Variable ergebnis ist nur innerhalb
    }                                                       // der Methode deklariert. Mit return wird
    return ergebnis;                                        // die Methode verlassen und der hinter
  }                                                         // return stehende Ausdruck zurückgegeben.
}                                                           // Ende der Klasse
```