

2.1

-array hat eckige klammern
-benötigt new operator bei dekleration
Syntax :
-Datentyp [] Array = new datentyp[length];
int array[] = new array[length];

2 möglichkeiten für initialisierung :

-new operator
-oder geschweife klammern, als zb int[] array = { .. } ;

beispiel:

```
int[] zahl = new int[6]
int[] zahl = {1,2,3,4,5,6}
```

2.2 ein wert kann an einem bestimmten index geändert werden, indem man einen wert an index i zuweist, also zb zahl[0] = 1;

2.3

der wert kann an der i-ten stelle ausgelesen werden, indem man i einfach in die eckige klammer reinschreibt

bsp:

```
i = 0
```

```
array[i] = 1;
```

```
System.out.println(Arrays.toString(array)); ( gibt den array aus, output :[1])
```

2.4

es wird ein objekt übergeben, was in der methode verändert wird. Die referenz auf dieses objekt ist jedoch immernoch gespeichert, da ein übergebenes objekt verändert wird

```
//@author Maximilian Raspe
import java.util.Arrays;
import java.util.Scanner;
public class Array {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        System.out.println("Wie viel Zahlen möchten sie eingeben");
        int anzahl = scan.nextInt();
        int[] zahlen = new int [anzahl];
        int produkt = 1;
        int summe = 0;

        for(int i = 0; i < zahlen.length; i++) {          //scanns ints ein und speichert diese am i-
ten index
            System.out.println("Zahl eingeben");
            zahlen[i] = scan.nextInt();
            System.out.println("Bisherige Zahlen : " + Arrays.toString(zahlen));
        }

        scan.close();
    }
}
```

```

        for(int i = 0; i < zahlen.length; i++) {      // berechnet produkt mit produkt =
produkt*zahlen[i], das gleiche bei summe
            produkt *= zahlen[i];
            summe += zahlen[i];
        }
        Arrays.sort(zahlen); // sortiere zahlen um größte, kleinste auszugeben
        System.out.println(Arrays.toString(zahlen));
        System.out.println("Kleinste : " + zahlen[0]);
        System.out.println("Größte : " + zahlen[anzahl - 1]);
        System.out.println("Produkt : " + produkt);
        System.out.println("Summe : " + summe);
        System.out.print("Durchschnitt : " + (double)summe / anzahl);
    }
}

//@author Maximilian Raspe
import java.awt.Color;

public class Feuer {

    public static final int R = 0;
    public static final int G = 1;
    public static final int B = 2;

    public static final int WIDTH = 200;
    public static final int HEIGHT = 150;

    public static void main(String[] args) {

        ImageFrame frame = new ImageFrame("Feuer");

        Color[][] colors = new Color[WIDTH][HEIGHT];

        // alle Pixel auf Schwarz setzen
        for (int i = 0; i < WIDTH; i++) {
            for (int j = 0; j < HEIGHT; j++) {
                colors[i][j] = Color.BLACK;
            }
        }

        while (true) {
            //erzeugt die Keimpunkte
            for(int i = 0; i <= 10; i++) {
                int x = (int)(Math.random() * 200);
                int y = 1;
                colors[x][y] = Color.yellow;
            }

            for(int i = 0; i <= 20; i++) {
                int x = (int)(Math.random()*200);

```

```

        int y = 1;
        colors[x][y] = Color.red;

    }

    for(int i = 0; i <= 10; i++) {
        int x = (int)( Math.random()* 200);
        int y = 1;
        colors[x][y] = Color.black;
    }

    for(int i = 1; i < WIDTH - 1; i++) {
        for(int a = 2; a < HEIGHT - 1; a++) {

            //rotwerte rausfinden
            int dRot = colors[i][a - 1].getRed();
            int lRot = colors[i - 1][a].getRed();
            int rRot = colors[i + 1][a].getRed();
            int aRot = colors[i][a].getRed();

            //blauwerte rausfinden
            int dBlau = colors[i][a - 1].getBlue();
            int lBlau = colors[i - 1][a].getBlue();
            int rBlau = colors[i + 1][a].getBlue();
            int aBlau = colors[i][a].getBlue();

            //grünwerte rausfinden
            int dGruen = colors[i][a - 1].getGreen();
            int lGruen = colors[i - 1][a].getGreen();
            int rGruen = colors[i + 1][a].getGreen();
            int aGruen = colors[i][a].getGreen();

            //neue farbwerte berchnen
            int neuRot = ((aRot + rRot + lRot + dRot)*2 / 8);
            int neuBlau = ((aBlau + rBlau + lBlau + dBlau)*2 / 8);
            int neuGruen = ((aGruen + rGruen + lGruen + dGruen)*2 / 8);

            //neue farbe und punkt
            Color neueFarbe = new Color(neuRot, neuGruen, neuBlau);
            colors[i][a] = neueFarbe;
        }
    }

    ImageFrame.drawArray(colors);

}

try {
    Thread.sleep(50);
} catch (InterruptedException e) {
    e.printStackTrace();
}
}
}
```

```
}
```

```
//@author Maximilian Raspe
import java.util.Scanner;
public class Pruefziffer {

    public static void main(String[] args) {

        Scanner scan = new Scanner(System.in);
        System.out.print("Bitte Seriennummer eingeben ");
        String pruefziffer = scan.nextLine();
        scan.close();

        int buchstabenwert = 0;
        int zwischenergebnis = 0;
        int ergebnis;
        boolean wert = false;

        int[] Alphabet = { ' ', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T',
                           'U', 'V', 'W', 'X', 'Y', 'Z' }; // array für zahlenwerte 1-26

        for(int i = 0; wert == false; i++) {
            if(pruefziffer.charAt(0) == Alphabet[i]) { //ermittelt den buchstabenwert (1-26) und bildet die quersumme aus diesem
                buchstabenwert = (i % 10) + (i / 10);
                wert = true; //sobald buchstabenwert ermittelt wird wert auf true gesetzt, was die schleife beendet
            }
            for(int a = 1; wert == true && pruefziffer.length() > a; a++) { //schleife geht erst an, sobald buchstabenwert ermittelt wurde und berechnet dann die restliche quersumme
                zwischenergebnis = (int)pruefziffer.charAt(a) - '0' + zwischenergebnis;
            }
            ergebnis = 8 - ((zwischenergebnis + buchstabenwert)%9); // ist die prüfziffer
            if(ergebnis == 0) { //wenn prüfziffer = 0, dann ist prüfziffer = 9
                System.out.print("Prüfziffer 9");
            }
            else {
                System.out.println("Prüfziffer " + ergebnis);
            }
        }
    }

//@author Maximilian Raspe
import java.util.Arrays;
import java.util.Scanner;
public class Sieb {

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

```

Scanner scan = new Scanner(System.in);
int grenze = scan.nextInt();
int[] zahlen = new int [grenze +1];
int[] fertig = new int[grenze + 1];
boolean [] gesiebt = new boolean[grenze +1];

scan.close();

for(int i = 2; i <= grenze ; i++) { //füllt den array mit zahlen und setzt alle zahlen
in gesiebt auf true
    zahlen[i] = i;
    gesiebt[i] = true;
}
for(int i = 2; i < grenze; i++) {
    for(int a = 1; a <= grenze / i; a++) {
        if (i * a != i) { // wenn i * a ungleich i ist, wird i * a auf false
gesetzt, was die primzahlen übrig lässt
            gesiebt[i * a] = false;
        }
    }
}
for(int i = 0; i <= grenze; i++) { // vergleicht nochmal die gesiebten werten und
wenn diese true sind, wird i als zahl in die liste gespeichert
    if(gesiebt[i] == true) {
        fertig[i] = i;
    }
}

System.out.print(Arrays.toString(fertig));

}

import java.util.Arrays;

//@author Maximilian Raspe

public class Wuerfel {

    public static void main(String[] args) {

        int[] zahlen = new int[6000000];
        int [] ergebnis = new int[6];

        boolean col = false;

        for(int i = 0; i < zahlen.length; i++) {
            zahlen[i] = (int)(Math.random()* 6) + 1; //generiert die würfelzahlen und
prüft ob 1,2,3,4,5 oder 6 und zählt hoch
            if(zahlen[i] == 1) ergebnis[0] += 1;
            if(zahlen[i] == 2) ergebnis[1] += 1;
    }
}

```

```
        if(zahlen[i] == 3) ergebnis[2] += 1;
        if(zahlen[i] == 4) ergebnis[3] += 1;
        if(zahlen[i] == 5) ergebnis[4] += 1;
        if(zahlen[i] == 6) ergebnis[5] += 1;
    }

    System.out.println("1" +"\t"+ "2" + "\t" + "3" + "\t" + "4" + "\t" + "5" + "\t" + "6");
    System.out.println("-----");
    System.out.println( ergebnis[0] + "\t" + ergebnis[1] + "\t" + ergebnis[2] + "\t" +
ergebnis[3] + "\t" + ergebnis[4] + "\t" + ergebnis[5]);
```

```
}
```