Bereich: Arrays (eindimensional)

Mittelwert und Standardabweichung

Musterlösung

Package: `de.dhbwka.java.exercise.arrays`

Klasse: `StandardDeviation`

```
package de.dhbwka.java.exercise.arrays;

import java.util.Random;

/**
 * @author DHBW lecturer
 * @version 1.0
 *
 * Part of lectures on 'Programming in Java'.
 * Baden-Wuerttemberg Cooperative State University.
 *
 * (C) 2015 by W. Geiger, T. Schlachter, C. Schmitt, W. Süß
 */
public class StandardDeviation {

	public static void main(String[] args) {
		int n = 100;
		Random rnd = new Random();
		int[] x = new int[n];

		// generate random numbers and calculate average
		int sum = 0;
		for (int i = 0; i < x.length; i++) {
			x[i] = rnd.nextInt(11); // 0..10
			sum += x[i]; // sum up the x[i]
		}
		double average = 1. * sum / n;
		System.out.println("Mittelwert: "+average);

		// calculate standard deviation
		double devSum = 0;
		for (int i = 0; i < x.length; i++) {
			devSum += Math.pow(x[i] - average, 2);
		}
		double deviation = Math.sqrt(devSum / (n - 1));
		System.out.println("Standardabweichung: "+deviation);
	}

}
```

Bereich: Arrays (eindimensional)

Fibonacci-Folge	**Musterlösung**
Package: `de.dhbwka.java.exercise.arrays`	**Klasse:** `Fibonacci`

```java
package de.dhbwka.java.exercise.arrays;

/**
 * @author DHBW lecturer
 * @version 1.0
 *
 * Part of lectures on 'Programming in Java'.
 * Baden-Wuerttemberg Cooperative State University.
 *
 * (C) 2015 by W. Geiger, T. Schlachter, C. Schmitt, W. Süß
 */
public class Fibonacci {

	public static void main(String[] args) {
		int n = 20; // or 50
		int[] fib = new int[n];
		fib[0] = fib[1] = 1;
		for (int i = 2; i < fib.length; i++) {
			fib[i] = fib[i - 1] + fib[i - 2];
		}
		// output
		for (int i = 0; i < fib.length; i++) {
			System.out.printf("%2d.: %d",i,fib[i]);
			System.out.println();
		}
	}

}
```

Bereich: Arrays (eindimensional)

Sieb des Eratostenes*	**Musterlösung**
Package: `de.dhbwka.java.exercise.arrays`	**Klasse:** `Eratostenes`

```java
package de.dhbwka.java.exercise.arrays;

/**
 * @author DHBW lecturer
 * @version 1.0
 *
 * Part of lectures on 'Programming in Java'.
 * Baden-Wuerttemberg Cooperative State University.
 *
 * (C) 2015 by W. Geiger, T. Schlachter, C. Schmitt, W. Süß
 */
public class Eratostenes {

	public static void main(String[] args) {
		int max = 100;
		boolean[] prim = new boolean[max];
		// initialize array with true
		for (int i = 2; i < prim.length; i++)
			prim[i] = true;
		// eratostenes' sieve
		for (int i = 2; i < prim.length; i++)
			if (prim[i])
				for (int j = i*2; j < prim.length; j+=i)
					prim[j] = false;
		// output
		for (int i = 0; i < prim.length; i++)
			if (prim[i])
				System.out.println(i);
	}

}
```

Bereich: Arrays (eindimensional)

Betrag eines Vektors	**Musterlösung**
Package: `de.dhbwka.java.exercise.arrays`	**Klasse:** `Norm`

```java
package de.dhbwka.java.exercise.arrays;

import java.util.Scanner;

/**
 * @author DHBW lecturer
 * @version 1.01
 *
 * Part of lectures on 'Programming in Java'.
 * Baden-Wuerttemberg Cooperative State University.
 *
 * (C) 2015-2016 by W. Geiger, T. Schlachter, C. Schmitt, W. Süß
 */
public class Norm {

	public static void main(String[] args) {
		Scanner scan = new Scanner(System.in);
		System.out.print("Bitte Anzahl der Elemente n eingeben: ");
		int n = scan.nextInt();
		int[] x = new int[n];
		int norm = 0;
		for (int i = 0; i < x.length; i++) {
			System.out.print("Bitte x_" + i + " eingeben: ");
			x[i] = scan.nextInt();
		}
		// this loop could be integrated with the preceding one
		for (int i = 0; i < x.length; i++)
			norm += x[i] * x[i];
		System.out.println("Der Betrag von x ist " + Math.sqrt(norm));
		scan.close();
	}

}
```

Bereich: Arrays (eindimensional)

Skalarprodukt zweier Vektoren — Musterlösung

Package: `de.dhbwka.java.exercise.arrays` **Klasse:** `DotProduct`

```
package de.dhbwka.java.exercise.arrays;

import java.util.Scanner;

/**
 * @author DHBW lecturer
 * @version 1.0
 *
 * Part of lectures on 'Programming in Java'.
 * Baden-Wuerttemberg Cooperative State University.
 *
 * (C) 2015 by W. Geiger, T. Schlachter, C. Schmitt, W. Süß
 */
public class DotProduct {

	public static void main(String[] args) {
		Scanner scan = new Scanner(System.in);
		System.out.print("Bitte Anzahl der Elemente n eingeben: ");
		int n = scan.nextInt();
		int[] x = new int[n];
		int[] y = new int[n];
		int dotProduct = 0;
		for (int i = 0; i < x.length; i++) {
			System.out.print("Bitte x_" + i + " eingeben: ");
			x[i] = scan.nextInt();
		}
		for (int i = 0; i < y.length; i++) {
			System.out.print("Bitte y_" + i + " eingeben: ");
			y[i] = scan.nextInt();
		}
		// this loop could be integrated with the preceding one
		for (int i = 0; i < x.length; i++)
			dotProduct += x[i] * y[i];
		System.out.println("Das Skalarprodukt von x und y ist "
				+dotProduct);
		scan.close();
	}

}
```

Bereich: Arrays (eindimensional)

Sortieren mit Bubblesort	**Musterlösung**
Package: `de.dhbwka.java.exercise.arrays`	**Klasse:** `BubbleSort`

```java
package de.dhbwka.java.exercise.arrays;

import java.util.Scanner;

/**
 * @author DHBW lecturer
 * @version 1.0
 *
 * Part of lectures on 'Programming in Java'.
 * Baden-Wuerttemberg Cooperative State University.
 *
 * (C) 2015 by W. Geiger, T. Schlachter, C. Schmitt, W. Süß
 */
public class BubbleSort {

	public static void main(String[] args) {
		Scanner scan = new Scanner(System.in);
		System.out.print("Bitte Anzahl der Elemente n eingeben: ");
		int n = scan.nextInt();
		int[] x = new int[n];
		for (int i = 0; i < x.length; i++) {
			System.out.print("Zahl " + i + " eingeben: ");
			x[i] = scan.nextInt();
		}
		boolean swapped;
		do {
			swapped = false;
			for (int i = 1; i < x.length; i++) {
				if (x[i-1]>x[i]) {
					int tmp = x[i-1];
					x[i-1] = x[i];
					x[i] = tmp;
					swapped = true;
				}
			}
		} while (swapped);
		System.out.print("Sortiert: ");
		for (int i : x) {
			System.out.print(i + " ");
		}
		scan.close();
	}

}
```