

Bereich: Methoden-Spezial**Potenzen****Musterlösung****Package:** de.dhbwka.java.exercise.methods**Klasse:** Exponentiation

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

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. Suess
 */
public class Exponentiation {

    public static double xPowerN(double x, int n) {
        if (n==0)
            return 1.0;
        return x * xPowerN(x, n-1);
    }

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("Geben Sie bitte die Basis ein: ");
        double x = scan.nextDouble();
        System.out.print("Geben Sie bitte den positiven " +
            "ganzzahligen Exponenten ein: ");
        int n = scan.nextInt();
        System.out.println(x + "^ " + n + " = " + xPowerN(x,n));
    }
}
```

Bereich: Methoden-Spezial**Fibonacci-Zahlen rekursiv****Musterlösung****Package:** de.dhbwka.java.exercise.methods**Klasse:** Fibonacci

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

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. Suess
 */
public class Fibonacci {

    /** Returns the value of the nth fibonacci number
     * @param n
     * @return value of the nth fibonacci number */
    private static int getFib(int n) {
        if (n <= 2)
            return 1;
        return getFib(n-1) + getFib(n-2);
    }

    public static void main(String[] args) {
        Scanner scan = new Scanner(System.in);
        System.out.print("n-te Fibonacci-Zahl bestimmen für n: ");
        int n = scan.nextInt();

        for (int i = 1; i <= n; i++) {
            System.out.println("F(" + i + ") = " + getFib(i));
        }

        scan.close();
    }
}
```

Bereich: Methoden-Spezial**Quicksort*****Musterlösung****Package:** de.dhbwka.java.exercise.methods**Klasse:** Quicksort

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

```
/**
 * @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 Quicksort {

    public static void quicksort(int[] daten, int links, int rechts) {
        if (links < rechts) {
            int teiler = teile(daten, links, rechts);
            quicksort(daten, links, teiler - 1);
            quicksort(daten, teiler + 1, rechts);
        }
    }

    public static int teile(int[] daten, int links, int rechts) {
        int i = links;
        int j = rechts - 1;
        int pivot = daten[rechts];

        do {
            // Suche von links ein Element, das größer als das Pivot ist
            while (daten[i] <= pivot && i < rechts) {
                i++;
            }
            // Suche von rechts ein Element, das kleiner als das Pivot ist
            while (daten[j] >= pivot && j > links) {
                j--;
            }
            if (i < j) {
                // tausche daten[i] mit daten[j]
                int temp = daten[i];
                daten[i] = daten[j];
                daten[j] = temp;
            }
        } while (i < j); // solange i an j nicht vorbeigelaufen ist

        // Tausche das Pivotelement mit seiner endgültigen Position i
        if (daten[i] > pivot) {
            // tausche daten[i] mit daten[rechts]
            int temp = daten[i];
            daten[i] = daten[rechts];
            daten[rechts] = temp;
        }
        // gib die Position des Pivotelements zurück
    }
}
```

```
        return i;
    }

    public static void sort(int[] array) {
        quicksort(array, 0, array.length - 1);
    }

    public static void main(String[] args) {
        // Testlauf
        int[] array = {9, 2, 5, 7, 8, 3, 4, 10, 1, 6};
        System.out.println("Unsortierte Zahlenfolge:");
        for (int a : array) {
            System.out.print(a + " ");
        }
        System.out.println("\n\nSortierte Zahlenfolge:");
        sort(array);
        for (int a : array) {
            System.out.print(a + " ");
        }
        System.out.println("");
    }
}
```