

**Bereich: Klassen (3)****Nimmspiel****Musterlösung****Package:** de.dhbwka.java.exercise.classes**Klasse:** Nimmspiel

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

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

    String[] spielerName = new String[2];
    int[] haufen = new int[2];
    java.util.Scanner scan;

    public Nimmspiel(String spieler1, String spieler2) {
        spielerName[0] = spieler1;
        spielerName[1] = spieler2;
        haufen[0] = (int) (Math.random() * 10) + 2;
        haufen[1] = (int) (Math.random() * 10) + 2;
    }

    public void gesamtesSpiel() {
        scan = new java.util.Scanner(System.in);
        int spieler = 1;
        do {
            if (spieler == 0) {
                spieler = 1;
            } else {
                spieler = 0;
            }
            spielzug(spieler);
        } while (!alleKugelnEntfernt());
        System.out.println("Spiel beendet.");
        System.out.println("Gewonnen hat Spieler " + spielerName[spieler]
            + ".");
    }
}
```

```
public void spielzug(int spielerNr) {
    int haufenNr;
    int gezogeneKugeln;
    boolean eingabeFehler = true;
    do {
        System.out.print("Spieler " + spielerName[spielerNr]
            + ": Von welchem Haufen ziehen Sie Kugeln? ");
        haufenNr = scan.nextInt() - 1;
        if ((haufenNr < 0) || (haufenNr > 1) || (haufen[haufenNr] == 0)) {
            System.out.println("Eingabefehler! Erneute Eingabe.");
        } else {
            eingabeFehler = false;
        }
    } while (eingabeFehler);

    eingabeFehler = true;
    do {
        System.out.print("Spieler " + spielerName[spielerNr]
            + ": Wieviele Kugeln ziehen Sie? ");
        gezogeneKugeln = scan.nextInt();
        if ((gezogeneKugeln < 1) || (gezogeneKugeln > haufen[haufenNr])) {
            System.out.println("Eingabefehler! Erneute Eingabe.");
        } else {
            eingabeFehler = false;
        }
    } while (eingabeFehler);
    haufen[haufenNr] -= gezogeneKugeln;
    System.out.println(this);
}

public boolean alleKugelnEntfernt() {
    return (haufen[0] == 0) && (haufen[1] == 0);
}

public String toString() {
    return "Spieler: " + spielerName[0] + " und " + spielerName[1]
        + ", Haufen 1: " + haufen[0] + " Kugel(n), Haufen 2: "
        + haufen[1] + " Kugel(n)";
}

public static void main(String[] args) {
    Nimmspiel spiel = new Nimmspiel("Alf", "Ben");
    // Test-Ausgabe
    System.out.println(spiel);
    spiel.gesamtesSpiel();
}
}
```

**Bereich: Klassen (3)****Lotto****Musterlösung****Package:** de.dhbwka.java.exercise.classes**Klasse:** Lotto

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

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

    int anzahlZuZiehendeKugeln;
    int gesamtzahlKugeln;
    int[] tipp;
    int[] gezogeneZahlen;
    java.util.Scanner scan;

    Lotto(int m, int n) {
        anzahlZuZiehendeKugeln = m;
        gesamtzahlKugeln = n;
        tipp = new int[anzahlZuZiehendeKugeln];
        gezogeneZahlen = new int[anzahlZuZiehendeKugeln];
        scan = new java.util.Scanner(System.in);
    }

    void spiel() {
        tippen();
        sortiere(tipp);
        ziehen();
        sortiere(gezogeneZahlen);
        // Test-Ausgabe
        System.out.println(this.toString());
        System.out.println("Richtige: " + richtige());
    }
}
```

```
void tippen() {
    boolean eingabeFehler;
    for (int i = 0; i < anzahlZuZiehendeKugeln; i++) {
        do {
            System.out.print("Geben Sie bitte Ihren Tipp für die " + (i + 1)
                + ". Zahl ein: ");
            eingabeFehler = false;
            tipp[i] = scan.nextInt();
            if ((tipp[i] < 1) || (tipp[i] > 49)) {
                System.out.println("Eingabefehler: Zahl zu groß oder"
                    + " zu klein.");
                eingabeFehler = true;
            }
            for (int k = 0; k < i; k++) {
                if (tipp[k] == tipp[i]) {
                    System.out.println("Eingabefehler: Zahl schon "
                        + "getippt.");
                    eingabeFehler = true;
                }
            }
        } while (eingabeFehler);
    }
}

void sortiere(int[] feld) {
    int temp;
    boolean tausch = true;
    while (tausch) {
        tausch = false;
        for (int i = 0; i < anzahlZuZiehendeKugeln - 1; i++) {
            if (feld[i] > feld[i + 1]) {
                temp = feld[i];
                feld[i] = feld[i + 1];
                feld[i + 1] = temp;
                tausch = true;
            }
        }
    }
}

void ziehen() {
    boolean schonVorhanden;
    for (int i = 0; i < anzahlZuZiehendeKugeln; i++) {
        do {
            schonVorhanden = false;
            gezogeneZahlen[i] = (int) (Math.random() * 49 + 1);
            for (int k = 0; k < i; k++) {
                if (gezogeneZahlen[k] == gezogeneZahlen[i]) {
                    schonVorhanden = true;
                }
            }
        } while (schonVorhanden);
    }
}
```

```
int richtige() {
    int anzRichtige = 0;
    for (int i = 0; i < anzahlZuZiehendeKugeln; i++) {
        for (int k = 0; k < anzahlZuZiehendeKugeln; k++) {
            if (tipp[i] == gezogeneZahlen[k]) {
                anzRichtige++;
            }
        }
    }
    return anzRichtige;
}

public String toString() {
    String str = "";
    if (tipp[0] != 0) {
        str += "Tipp: ";
        for (int i = 0; i < anzahlZuZiehendeKugeln; i++) {
            str = str + tipp[i] + " ";
        }
        str += "\n";
    }
    if (gezogeneZahlen[0] != 0) {
        str += "Gezogene Zahlen: ";
        for (int i = 0; i < anzahlZuZiehendeKugeln; i++) {
            str = str + gezogeneZahlen[i] + " ";
        }
        str += "\n";
    }
    return str;
}

public static void main(String[] args) {
    Lotto deutschesLotto = new Lotto(6, 49);
    deutschesLotto.spiel();
}
}
```

**Bereich: Klassen (3)****Schema MasterMind****Musterlösung****Package:** de.dhbwka.java.exercise.classes**Klasse:** MasterMind

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

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

    char[] computerKombination;
    char[][] spielerKombination;
    int[] richtigeStellen;
    int[] falscheStellen;
    java.util.Scanner scan;
    int versuch;

    MasterMind() {
        computerKombination = new char[5];
        spielerKombination = new char[50][5];
        richtigeStellen = new int[50];
        falscheStellen = new int[50];
        scan = new java.util.Scanner(System.in);
    }

    void spiel() {
        bestimmeComputerKombination();
        rateSchleife();
    }

    void bestimmeComputerKombination() {
        for (int i = 0; i < 5; i++) {
            computerKombination[i] = (char) ('A' + (int) (Math.random() * 8));
        }
        // Ausgabe für das Testen
        System.out.print("Computer-Kombination: ");
        for (int i = 0; i < 5; i++) {
            System.out.print(computerKombination[i]);
        }
        System.out.println("");
    }
}
```



```
void richtigeBuchstaben() {
    richtigeStellen[versuch] = 0;
    falscheStellen[versuch] = 0;
    for (char ch = 'A'; ch <= 'H'; ch++) {
        int buchstabeInComputerKombination = 0;
        int buchstabeInSpielerKombination = 0;
        int buchstabeUebereinstimmend = 0;
        for (int i = 0; i < 5; i++) {
            if (computerKombination[i] == ch) {
                buchstabeInComputerKombination++;
                if (spielerKombination[versuch][i] == ch) {
                    buchstabeUebereinstimmend++;
                }
            }
        }
        for (int i = 0; i < 5; i++) {
            if (spielerKombination[versuch][i] == ch) {
                buchstabeInSpielerKombination++;
            }
        }
        richtigeStellen[versuch] += buchstabeUebereinstimmend;
        falscheStellen[versuch]
            += Math.min(buchstabeInSpielerKombination,
                buchstabeInComputerKombination)
            - buchstabeUebereinstimmend;
    }
}

public static void main(String[] args) {
    MasterMind mm = new MasterMind();
    mm.spiel();
}
}
```