

Versuchen Sie immer, Ihren Code zu kommentieren!

Aufgabe 1

Schreibe ein Programm, das eine Methode enthält, welche zwei ganze Zahlen akzeptiert und ihre Summe zurückgibt.

Aufgabe 2

Schreibe ein Programm, das eine Methode enthält, welche drei Gleitkommazahlen akzeptiert und ihr Produkt zurückgibt.

Aufgabe 3

Schreibe ein Programm, das eine Methode enthält, welche eine Zeichenkette akzeptiert und die Anzahl der Buchstaben darin zurückgibt.

Aufgabe 4

Schreibe ein Programm, das den Benutzer nach zwei Zahlen fragt und dann die größere der beiden Zahlen ausgibt. Verwende dazu die folgenden Methoden:

1. **int GetMaxNumber(int num1, int num2)**

Aufgabe 5

Schreibe ein Programm, das den Benutzer nach einer Zahl fragt und dann überprüft, ob die Zahl gerade oder ungerade ist. Verwende dazu die folgenden Methoden:

1. **bool IsEven(int number)**

Aufgabe 6

Schreibe ein Programm, das den Benutzer nach seinem Namen fragt und dann eine personalisierte Begrüßung ausgibt. Erstelle dazu die folgenden Methoden:

1. **string ReadNameFromUser()**
2. **void PrintGreeting(string name)**

Aufgabe 7

Erweitere das Programm aus Aufgabe 6, indem du eine Methode hinzufügst, die den Benutzer nach seinem Alter fragt. Die Methode PrintGreeting soll nun auch das Alter als Parameter akzeptieren.

1. **int ReadAgeFromUser()**
2. **void PrintGreeting(string name, int age)**

Aufgabe 8

Schreibe ein Programm, das den Benutzer nach drei Zahlen fragt, diese addiert und das Ergebnis ausgibt. Erstelle dazu die folgenden Methoden:

1. **int ReadNumberFromUser(string message)**
2. **int CalculateSum (int numberOne, int numberTwo, int numberThree)**
3. **void PrintResult(int result)**

Aufgabe 9

Erweitere das Programm aus Aufgabe 8, indem du eine Methode hinzufügst, die das Produkt der drei Zahlen berechnet. PrintResult soll nun auch das Produkt als Parameter akzeptieren. Erstelle dazu die folgenden Methoden:

1. **int CalculateProduct (int numberOne, int numberTwo, int numberThree)**
2. **void PrintResult(int sumResult, int productResult)**

Aufgabe 10

Schreibe ein Programm, das zwei Überladungen derselben Methode enthält. Eine für die Addition von zwei Ganzzahlen und eine für die Addition von zwei Gleitkommazahlen.

Aufgabe 11

Schreibe ein Programm, das eine Methode implementiert, welche zwei Ganzzahlen und eine optionale dritte Ganzzahl mit Standardwert akzeptiert. Gib die Summe der drei Zahlen zurück.

Aufgabe 12

Schreibe ein Programm, das eine Methode enthält, welche eine variable Anzahl von Ganzzahlen akzeptiert, und ihre Summe zurückgibt (params keyword).

Aufgabe 13

Erstellen Sie ein Programm, welches vom Anwender erfragt wie viele Zahlen er angeben möchte (1 bis 20). Dann werden die Zahlen in ein Array passender Größe eingelesen und Ihr Programm ermittelt die kleinste und größte eingegebene Zahl.

Ihr Programm muss (außer der Main-Methode) mindestens die folgenden Methoden aufweisen:

1. **int ReadNumberFromUser(string text, int min, int max)**
2. **int[] ReadInArray(int numOfElements)**
3. **int FindMin(int[] searchArray)**
4. **int FindMax(int[] searchArray)**

Aufgabe 14

Erstellen Sie ein Programm, welches vom Anwender erfragt wie viele Zahlen (1 bis 100) er in einem Array haben möchte. Dann wird das Array mit Zufallszahlen gefüllt und ausgegeben. Schließlich werden die Summe und der Mittelwert aller Zahlen im Array ausgegeben.

Ihr Programm muss (außer der Main-Methode) mindestens die folgenden Methoden aufweisen:

1. **int ReadNumberFromUser(string text, int min, int max)**
2. **void FillArrayWithRandomNumbers(int[] array)**
3. **int Sum(int[] array)**
4. **double Avg(int[] array)**
5. **void PrintArray(int[] array)**

Aufgabe 15

Erstellen Sie ein Programm, welches vom Anwender erfragt wie viele Zahlen (z.B. 1 bis 100) er in einem Array haben möchte. Dann wird gefragt, in welchem Bereich die Zahlen in dem Array liegen sollen (z.B. 7-14). Anschließend wird das Array mit den Zufallszahlen aus diesem Bereich gefüllt. Ihr Programm soll nun die Häufigkeit jeder Zahl im Array ermitteln und diese in einem 2D Array speichern (Zahl + Häufigkeit). Anschließend werden die Zahlen und ihre Häufigkeit ausgegeben.

Ihr Programm muss (außer der Main-Methode) mindestens die folgenden Methoden aufweisen.

1. **int ReadNumberFromUser(string text, int min, int max)**
2. **int[] FillArrayWithRandomNumbers(int length, int min, int max)**
3. **int[,] GetFrequency(int[] array, int min, int max)**
4. **void PrintResult(int[] arr, int[,] frequency)**

Beispiel:

Geben Sie die Anzahl der Zahlen im Array ein (1 bis 100): 5

Geben Sie das Minimum des Intervalls ein: 2

Geben Sie das Maximum des Intervalls ein: 4

Zufallszahlen im Array:

3 2 4 4 4

Häufigkeiten:

Die Zahl 2 kommt 1 Mal vor.

Die Zahl 3 kommt 1 Mal vor.

Die Zahl 4 kommt 3 Mal vor.

Aufgabe 16

Erstellen Sie ein Programm, das den Benutzer nach einer Zahl fragt. Erstelle dazu eine Methode mit dem Namen TryReadNumber. Wenn die Eingabe des Benutzers eine gültige Zahl war, sollte die Methode true zurückgeben und die Zahl über den out Parameter setzen. Wenn die eingegebene Nummer ungültig war, wird false zurückgegeben. Wenn die Zahl gültig ist, geben Sie sie auf der Konsole aus.

1. **bool TryReadNumber(string text, out int x)**

Aufgabe 17

Der antike Mathematiker Eratosthenes von Kyrene (ca. 275 - 194 v.Chr.) hat einen Algorithmus zur Bestimmung aller Primzahlen, die kleiner oder gleich einer vorgegebenen Zahl sind, entwickelt. Dieses Verfahren ist unter dem Namen „Das Sieb des Eratosthenes“ bekannt.

Das Verfahren:

Zunächst werden alle Zahlen 2, 3, 4... bis zu einem frei wählbaren Maximalwert MAX in einem Array eingetragen. Die zunächst nicht markierten Zahlen werden alle als potenzielle

Primzahlen betrachtet. Die kleinste nicht markierte Zahl muss immer eine Primzahl sein. Nachdem eine Primzahl gefunden wurde, werden alle Vielfachen dieser Primzahl als Nicht-Primzahl markiert. Dann fährt man mit der nächstgrößeren Primzahl fort und streicht erneut alle Vielfachen. Das wendet man auf alle Zahlen an, bis das Quadrat der aktuellen Primzahl größer als die Schranke MAX ist. Dann sind alle nicht markierten Zahlen im Array Primzahlen.

Zur Anschauung kann man sich eine Online-Implementierung ansehen:

<http://www.arndt-bruenner.de/mathe/scripts/eratosthenes.htm>

Ihre Aufgabe ist es ein Programm zu schreiben, welches nach dem angegebenen Algorithmus die Primzahlen zwischen 2 und 1000 berechnet und ausgibt.