



Deklaration und Initialisierung

Eine Variable muss zuerst **deklariert** und dann **initialisiert** werden.

1. Deklaration

```
string name = "Peter";
```

2. Initialisierung

```
public static void ZeigInfoPerson()  
{  
    string name; // 1. Deklaration  
    name = "Peter"; // 2. Initialisierung  
  
    Console.WriteLine(name + " ist 19 Jahre alt");  
    Console.WriteLine(name + " wiegt 80,5 kg.");  
    Console.ReadKey();  
}
```

Auch mit einer Anweisung möglich



Zur Wiederholung

Deklaration

Festlegen, welchen **Datentyp** und **Namen** die Variable hat.

▶ Codebeispiel: `string name;`

Initialisierung

Festlegen, welchen **Wert** die Variable hat (Wert zuweisen).

▶ Codebeispiel: `name = "Peter";`



Datentypen festlegen

Bei der Deklaration einer Variable muss ein **Datentyp** festgelegt werden. In C# gibt es 18 **integrierte Datentypen**.

► Wir konzentrieren uns auf 12 davon (3 besonders nützlich)

Wahrheitswert:	<code>bool</code>
Zeichen:	<code>string</code> , <code>char</code>
Kommazahlen:	<code>double</code>
Ganzzahlen:	<code>byte</code> , <code>ushort</code> , <code>uint</code> , <code>ulong</code> , <code>sbyte</code> , <code>short</code> , <code>int</code> , <code>long</code>



Wahrheitswerte

Datentyp	Beispiel	Wertebereich	Größe
bool	true, false	true oder false	8 Bit

Zeichen

Datentyp	Beispiel	Wertebereich	Größe
string	"Hallo", "42"	Text in doppelten Anführungsstrichen	...
char	'H', '2'	Ein Zeichen in einfachen Anführungsstrichen	16 Bit

Gleitkommazahlen *

Datentyp	Beispiel	Wertebereich	Größe
double	3.1415	Kommazahl bis 15/16 Nachkommastellen	64 Bit

* Gleit... bedeutet, dass irgendwann gerundet wird (Nicht unendlich genau).



Ganzzahlen (positiv)

Datentyp	Beispiel	Wertebereich	Größe
byte	42	0 bis 255	8 Bit
ushort	42	0 bis 65.535	16 Bit
uint	42	0 bis ca. 4.29 Mrd.	32 Bit
ulong	42	0 bis ca. 18.44 Trillionen	64 Bit

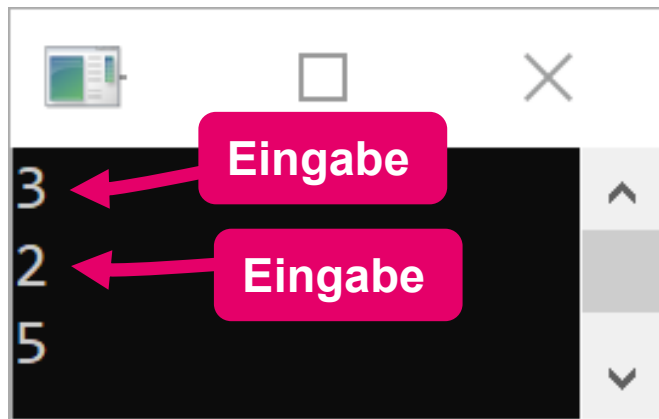
Ganzzahlen (negativ und positiv)

Datentyp	Beispiel	Wertebereich	Größe
sbyte	-42, 42	-128 bis 127	8 Bit
short	-42, 42	-32.768 bis 32.767	16 Bit
int	-42, 42	ca. -2.14 Mrd. bis +2.14 Mrd	32 Bit
long	-42, 42	ca. -9.22 Trillionen bis +9.22 Trillionen	64 Bit



Übung

Schreiben Sie die Methode **Addition()**, die Folgendes umsetzt.

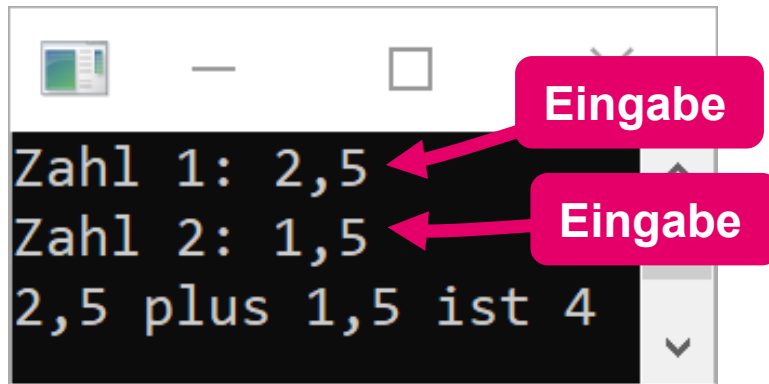


```
public static void Addition()  
{  
    double z1 = Convert.ToDouble(Console.ReadLine());  
    double z2 = Convert.ToDouble(Console.ReadLine());  
    Console.WriteLine(z1 + z2);  
    Console.ReadKey();  
}
```



Übung

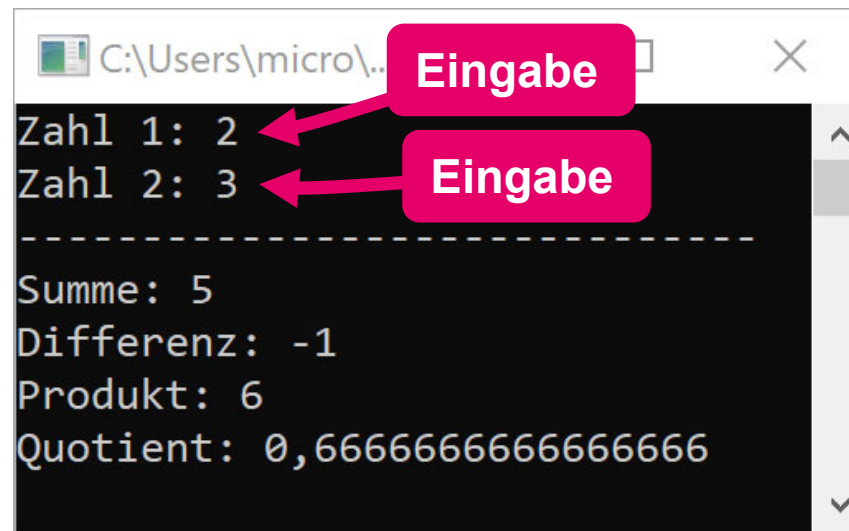
Schreiben Sie die Methode **AdditionMitText()**, die Folgendes umsetzt.



```
public static void AdditionMitText()
{
    Console.Write("Zahl 1: ");
    double z1 = Convert.ToDouble(Console.ReadLine());
    Console.Write("Zahl 2: ");
    double z2 = Convert.ToDouble(Console.ReadLine());
    Console.WriteLine(z1 + " plus " + z2 + " ist " + (z1 + z2));
    Console.ReadKey();
}
```

Übung

Schreiben Sie die Methode **Rechenarten()**, die Folgendes umsetzt.



```
C:\Users\micro\...
Zahl 1: 2
Zahl 2: 3
-----
Summe: 5
Differenz: -1
Produkt: 6
Quotient: 0,6666666666666666
```

```
public static void Rechenarten()
{
    Console.WriteLine("Zahl 1: ");
    double z1 = Convert.ToDouble(Console.ReadLine());
    Console.WriteLine("Zahl 2: ");
    double z2 = Convert.ToDouble(Console.ReadLine());
    Console.WriteLine("-----");
    Console.WriteLine("Summe: " + (z1 + z2));
    Console.WriteLine("Differenz: " + (z1 - z2));
    Console.WriteLine("Produkt: " + (z1 * z2));
    Console.WriteLine("Quotient: " + (z1 / z2));
    Console.ReadKey();
}
```




Übung

Schreiben Sie die Methode **ZeigSumme()**, welche Folgendes umsetzt.

```
C:\Users\mi...  
2,5 plus 1,5 ist 4
```

Die **ersten zwei Zahlen** sollen in Variablen gespeichert werden ($z1$ und $z2$) und das Ergebnis ausgerechnet werden. Hinweis: **($z1 + z2$)**.

```
public static void ZeigSumme()  
{  
    double z1 = 2.5;  
    double z2 = 1.5;  
  
    Console.WriteLine(z1 + " plus " + z2 + " ist " + (z1 + z2));  
    Console.ReadKey();  
}
```



Wichtig

Wenn eine Zahl ausgerechnet werden soll, dann muss die **Rechnung in Klammern ()** gesetzt werden. Ohne Klammern passiert das hier:

```
public static void ZeigSumme()
{
    double z1 = 2.5;
    double z2 = 1.5;

    Console.WriteLine(z1 + " plus " + z2 + " ist " + z1 + z2);
    Console.ReadKey();
}
```

Ohne Klammern wird nicht gerechnet.

```
C:\Users\mi... 2,5 plus 1,5 ist 2,51,5
```



Übung Fehlersuche

Finde die Syntax-Fehler im Quelltext (4 Zeichen)

```
public static void ZeigInfoPerson();  
{  
    string name = "Peter";  
    int alter = 19;  
    double gewicht = 80.5;  
  
    Console.WriteLine(name + " ist " + alter + " Jahre alt.")  
    Console.WriteLine(name + " wiegt " + gewicht + " kg.")  
    Console.ReadKey()  
}
```



Übung Fehlersuche

Finde die Syntax-Fehler im Quelltext (5 Zeichen)

```
public static void ZeigInfoPerson  
{  
    string name = "Peter";  
    int alter = 19;  
    double gewicht = 80,5;  
  
    Console.WriteLine(name + " ist " + alter + " Jahre alt.");  
    Console.WriteLine(name + " wiegt " + gewicht + " kg.");  
    Console.ReadKey;  
}
```



Übung Fehlersuche

Finde die Syntax-Fehler im Quelltext (4 Zeichen)

```
public static void ZeigInfoPerson()  
{  
    string name = "Peter";  
    int alter = 19;  
    double gewicht = 80.5;  
    console.WriteLine(name + " ist " + alter + " Jahre alt.");  
    console.WriteLine(name + " wiegt " + gewicht + " kg.");  
    console.ReadKey();  
}
```

C →
C →
C →
K →



Übung Fehlersuche

Finde die Syntax-Fehler im Quelltext (3 Zeichen)

```
public static void ZeigInfoPerson()  
{  
    string name = "Peter";  
    int alter = 19;  
    double gewicht = 80.5;  
  
    Console.WriteLine(name + " ist " + Alter + " Jahre alt.");  
    Console.WriteLine(name + " wiegt " + Gewicht + " kg.");  
    Console.ReadKey();  
}
```

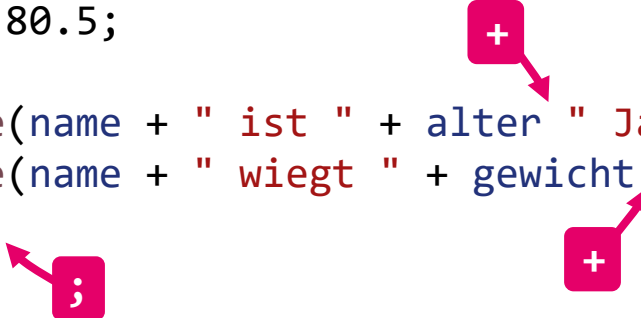




Übung Fehlersuche

Finde die Syntax-Fehler im Quelltext (3 Zeichen)

```
public static void ZeigInfoPerson()  
{  
    string name = "Peter";  
    int alter = 19;  
    double gewicht = 80.5;  
  
    Console.WriteLine(name + " ist " + alter " Jahre alt.");  
    Console.WriteLine(name + " wiegt " + gewicht " kg.");  
    Console.ReadKey()  
}
```





Übung Fehlersuche

Finde die Syntax-Fehler im Quelltext (4 Zeichen)

```
public static void ZeigInfoPerson()  
{  
    string name = "Peter";  
    int alter = 19;  
    double gewicht = 80.5;  
  
    Console.WriteLine(name + " ist " + alter + " Jahre alt.");  
    Console.WriteLine(name + " wiegt " + gewicht + " kg.");  
    Console.ReadKey();  
}
```

Four red boxes containing double quotes (") are placed around the code. Two boxes are above the first string concatenation, and two are below the second. Arrows point from each box to the corresponding opening quote in the code, indicating that the closing quote is missing.