

Aufgaben Prozeduren:

Aufgabe 1: Ausgabe1

Schreiben Sie ein Programm, das eine Prozedur `AusgabeGutenMorgen()` enthält. Die Prozedur soll beim Aufruf immer den Text „Guten Morgen!“ auf der Konsole ausgeben.

Beispiel-BildschirmAusgabe bei Aufruf `AusgabeGutenMorgen()`:

```
Guten Morgen!           // Methode
```

Aufgabe 2: Ausgabe2

Schreiben Sie ein Programm, das eine Prozedur `AusgabeText()` enthält. Die Prozedur soll einen Parameter `text` vom Typ `string` haben. Bei Aufruf der Prozedur soll der übergebene Text auf der Konsole ausgegeben werden.

Beispiel-BildschirmAusgabe bei Aufruf `AusgabeText("Hallo Semus")`:

```
Hallo Semus             // Methode
```

Aufgabe 3: Ausgabe3

Schreiben Sie ein Programm, das eine Prozedur `AusgabeText()` enthält. Die Prozedur soll einen Parameter `text` vom Typ `string` haben. Der Benutzer soll diesmal im Hauptprogramm einen Text eingeben, und die Prozedur gibt diesen Text bei Aufruf wieder auf der Konsole aus.

Beispiel-BildschirmAusgabe:

```
Eingabe: Hallo           // Hauptprogramm  
Sie haben 'Hallo' eingegeben. // Methode
```

Aufgabe 4: Addition1

Schreiben Sie ein Programm, das eine Prozedur `Addition()` enthält. Die Prozedur soll zwei Parameter `zahl1` und `zahl2` vom Typ `int` haben. Der Benutzer soll zwei Zahlen eingeben und die Prozedur gibt bei Aufruf das Ergebnis der Addition auf der Konsole aus.

Beispiel-BildschirmAusgabe:

```
Eingabe Zahl1: 5         // Hauptprogramm  
Eingabe Zahl2: 7         // Hauptprogramm  
Das Ergebnis der Addition ist 12. // Methode
```

Aufgabe 5: Mehrfache Ausgabe

Schreiben Sie ein Programm, das eine Prozedur `AusgabeText()` enthält. Die Prozedur hat einen Parameter `text` vom Typ `string` und einen weiteren Parameter `anzahl` vom Typ `int`. Bei Aufruf soll die Prozedur den übergebenen Text in der gewünschten Anzahl auf der Konsole ausgeben.

Beispiel-BildschirmAusgabe bei Aufruf `AusgabeText("Hallo Semus", 3)`:

```
Hallo Semus             // Methode  
Hallo Semus             // Methode  
Hallo Semus             // Methode
```

Aufgabe 6: Prozeduren-Verkettung

Schreiben Sie ein Programm, das die drei Prozeduren `Prozedur1()`, `Prozedur2()` und `Prozedur3()` enthält. Jede der drei Prozeduren gibt bei Aufruf einen Text auf der Konsole aus. Die `Prozedur1()` ruft nach der Ausgabe des Textes die `Prozedur2()` auf, die `Prozedur2()` ruft nach der Ausgabe die `Prozedur3()` auf.

Beispiel-BildschirmAusgabe bei Aufruf von `Prozedur1()`:

```
Hallo von Prozedur1
Hallo von Prozedur2
Hallo von Prozedur3
```

Aufgabe 7: Taschenrechner

Schreiben Sie ein Programm, das einen simplen Taschenrechner simuliert. Im Programm gibt es die vier Prozeduren `Addition()`, `Subtraktion()`, `Multiplikation()` und `Division()`. Alle Prozeduren haben jeweils zwei Parameter `zahl1` und `zahl2` vom Typ `int` und geben bei Aufruf das Ergebnis der jeweiligen Rechenoperation auf der Konsole aus. Der Benutzer gibt zwei Zahlen und einen Operator (+, -, *, /) ein. Das Programm ruft dann je nach eingegebenem Operator die richtige Prozedur auf und übergibt die beiden eingegebenen Zahlen.

Beispiel-BildschirmAusgabe:

```
Eingabe Zahl1: 5
Eingabe Zahl2: 7
Eingabe Operator: *
Das Ergebnis der Multiplikation ist 35.
```

Aufgabe 8: Array-Ausgabe

Schreiben Sie ein Programm, das eine Prozedur `ArrayAusgabe()` enthält. Die Prozedur hat einen Parameter `array` vom Typ `int[,]`. Die Prozedur bekommt beim Aufruf ein zweidimensionales Array beliebiger Größe übergeben und gibt dieses strukturiert auf der Konsole aus.

Beispiel-BildschirmAusgabe:

```
Inhalt des Arrays:
2 7 9
6 0 1
3 7 4
1 2 3
```

Aufgabe 9: Teiler

Schreiben Sie ein Programm, das eine Prozedur `AnzeigeTeiler()` enthält. Die Prozedur hat einen Parameter `zahl` vom Typ `int`. Der Benutzer kann im Hauptprogramm eine Zahl eingeben und die Prozedur gibt bei Aufruf alle Teiler der übergebenen Zahl auf der Konsole aus, also alle Zahlen, durch die sich die eingegebene Zahl ohne Rest teilen lässt.

Beispiel-BildschirmAusgabe:

```
Eingabe Zahl: 12
Teiler von 12 sind: 1, 2, 3, 4, 6, 12
```