

Im Seriellen Monitor sollen verschiedene Information zu Städten angezeigt werden.

Bei dieser Aufgabe lernst du einen neuen Datentyp kennen. Arrays erweisen sich als geeignet, wenn eine Vielzahl von Daten eines gleichen Variablentyps verarbeitet werden sollen.

Strukturen (struct) sind hingegen eine Sammlung und Beschreibung verschiedener Daten. Sie können beliebige Datentypen verwenden.

Die Struktur ist die Vorlage für die Definition der konkreten Daten.

In den Beispielen werden verschiedene Informationen zu den einzelnen Städten mit Hilfe unterschiedlicher Datentypen zusammengefasst.

Zunächst wird eine Schablone definiert:

```
struct Stadt
{
    int Platz;
    String Name;
    String KFZ_Kennzeichen;
    float Einwohner;
    String Vorwahl;
    int Hoehe;
};
```

Definition der Daten:

```
const int MAX = 3;
Stadt Namen[MAX] =
{
    {1, "Berlin", "B", 3.64, "030", 30},
    {2, "Hamburg", "HH", 1.84, "040", 6},
    {3, "München", "M", 1.47, "089", 519}
};
```

Der Zugriff auf die Elemente erfolgt über die Punkt-Schreibweise und die Nummer des Elements.

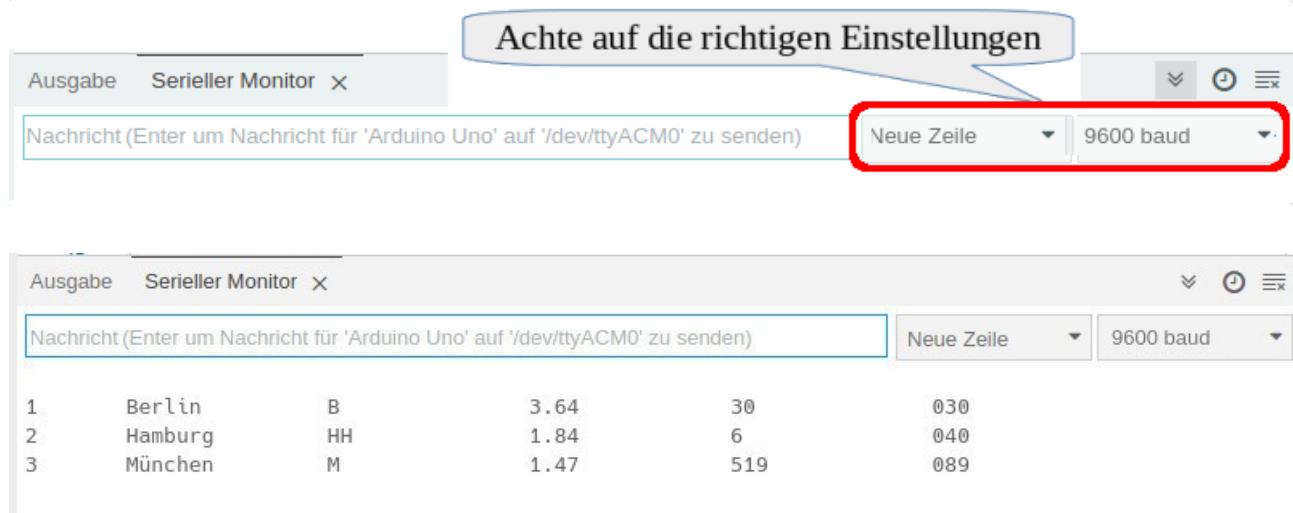
Beispiel:

```
Namen[0].Name = "Berlin";
Namen[1].Platz = 4;
Namen[2].Hoehe = 520;
```

Eine Methode zeigt die Daten an:

```
for (int i = 0; i < MAX; i++ )
{
  Serial.print(String(Namen[i].Platz) + "\t");
  Serial.print(Namen[i].Name + "\t\t");
  Serial.print(Namen[i].KFZ_Kennzeichen + "\t\t");
  Serial.print(String(Namen[i].Einwohner) + "\t\t");
  Serial.print(String(Namen[i].Hoehe) + "\t\t");
  Serial.println(Namen[i].Vorwahl);
}
```

So sieht es aus:



Nach Eingabe einer Zahl zwischen 1 und 4 im Seriellen Monitor soll die Information zur entsprechenden Stadt des Arrays angezeigt werden.

Ergänze die Daten für die jeweiligen Bundesländer und für Köln.

```
{4, "Köln", "K", 1.04, "0221", 53, "NRW"}
```

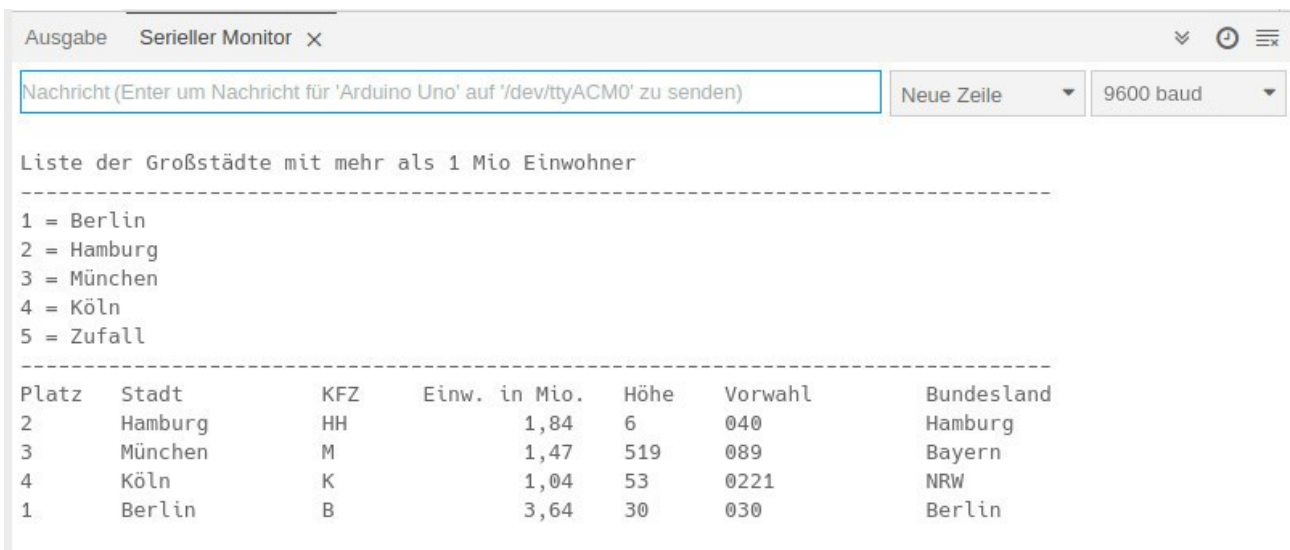
Du musst auch die Größe des Arrays (MAX) ändern.

```
struct Stadt
{
  int Platz;
  String Name;
  String KFZ_Kennzeichen;
  float Einwohner;
  String Vorwahl;
  int Hoehe;
  String Land;
};
```

```
const int MAX = 4;

// Array definieren
Stadt Namen[MAX] =
{
  {1, "Berlin", "B", 3.64, "030", 30, "Berlin"},
  {2, "Hamburg", "HH", 1.84, "040", 6, "Hamburg"},
  {3, "München", "M", 1.47, "089", 519, "Bayern"},
  {4, "Köln", "K", 1.04, "0221", 53, "NRW"}
};
```

So soll es aussehen:



Ausgabe Serieller Monitor x

Nachricht (Enter um Nachricht für 'Arduino Uno' auf '/dev/ttyACM0' zu senden) Neue Zeile 9600 baud

Liste der Großstädte mit mehr als 1 Mio Einwohner

1 = Berlin
2 = Hamburg
3 = München
4 = Köln
5 = Zufall

Platz	Stadt	KFZ	Einw. in Mio.	Höhe	Vorwahl	Bundesland
2	Hamburg	HH	1,84	6	040	Hamburg
3	München	M	1,47	519	089	Bayern
4	Köln	K	1,04	53	0221	NRW
1	Berlin	B	3,64	30	030	Berlin

Im setup-Teil wird der Serielle Monitor gestartet und über den Ablauf des Programms informiert. Mit Hilfe des Tabulators (\t) wird die Ausgabe formatiert.

```
struct Stadt
{
  int Platz;
  String Name;
  String KFZ_Kennzeichen;
  float Einwohner;
  String Vorwahl;
  int Hoehe;
  String Land;
};

const int MAX = 4;

// Array definieren
Stadt Namen[MAX] =
{
  {1, "Berlin", "B", 3.64, "030", 30, "Berlin"},
  {2, "Hamburg", "HH", 1.84, "040", 6, "Hamburg"},
  {3, "München", "M", 1.47, "089", 519, "Bayern"},
  {4, "Köln", "K", 1.04, "0221", 53, "NRW"}
};
```

Der setup-Teil.

```
void setup()
{
    Serial.begin(9600);

    // auf serielle Verbindung warten
    while (!Serial);

    // Info
    Serial.println("Liste der Gro\u00dfst\u00e4dte mit mehr als 1 Mio Einwohner");
    Serial.println("-----");
    Serial.println("1 = Berlin");
    Serial.println("2 = Hamburg");
    Serial.println("3 = München");
    Serial.println("4 = Köln");
    Serial.println("-----");
    Serial.println("Platz\tStadt\t\tKFZ\tEinw. in Mio.\tHöhe\tVorwahl\t\tBundesland");
}
```

Der loop-Teil. Beachte die Kommentare.

```
void loop()
{
    // auf serielle Eingabe warten
    while (Serial.available() > 0)
    {
        // Eingabe im Seriellen Monitor lesen
        String Eingabe = Serial.readStringUntil(char(13));

        // String Eingabe nach int umwandeln
        // die Funktion ZeigeDaten kann dann direkt aufgerufen werden
        int Auswahl = Eingabe.toInt();

        // wenn Auswahl > 0 und < 5
        // weil das Array mit 0 beginnt, 1 abziehen
        if (Auswahl > 0 && Auswahl < 5) ZeigeDaten(Auswahl - 1);

        // zufällige Auswahl
        if (Auswahl == 5)
        {
            int Minimum = 0;
            int Maximum = 4;
            int Zahl = random(Minimum, Maximum);
            ZeigeDaten(Zahl);
        }
    }
}
```

Die Funktion ZeigeDaten:

```
void ZeigeDaten(int i)
{
    // bei der Einwohnerzahl . durch , ersetzen
    String Einwohner = String(Namen[i].Einwohner);
    Einwohner.replace(".", ",");

    // Daten anzeigen, mit Tabulator (\t) formatieren
    Serial.print(String(Namen[i].Platz) + "\t");
    Serial.print(Namen[i].Name + "\t\t");
    Serial.print(Namen[i].KFZ_Kennzeichen + "\t\t");
    Serial.print(Einwohner + "\t");
    Serial.print(String(Namen[i].Hoehe) + "\t");
    Serial.print(Namen[i].Vorwahl + "\t\t");
    Serial.println(Namen[i].Land);
}
```

Hartmut Waller (hartmut-waller.info/arduino-blog) Letzte Änderung: 16.05.24