

Serial.read - Vokale zählen

In diesem Programm soll die Eingabe im Seriellen Monitor dazu verwendet werden, in einem Satz die Anzahl der Vokale zu zählen und anzuzeigen. Außerdem erfährst du einiges über Zeichenketten.

Zeichenketten (Strings) können aus einem einzelnen Zeichen (char) oder mehreren Zeichen bestehen.

Sie werden mit den Schlüsselwörtern String und char definiert.

```
String Zeichenkette = "abcd";
char Buchstabe = "a";
```

String und char können auch in einem Array abgelegt werden:

```
// definiert ein char-Array mit 5 Elementen
char Buchstaben[5];

// definiert ein Array mit den Elemente aeiou
char Buchstaben[] = {"aeiou"};

// definiert ein String-Array
String Satzteile[] = {"Programmieren ", "ist ", "toll!"};
```

Eine for-Schleife und **sizeof** ermöglichen den Zugriff auf die Elemente des Arrays.

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

  // Array mit den Buchstaben anzeigen
  // char belegt 1 Byte
  for(int i = 0; i < sizeof(Buchstaben); i++)
  {
    Serial.print(Buchstaben[i]);
  }

  // Array mit den Satzteilen anzeigen
  for(int i = 0; i < sizeof(Satzteile) / sizeof(Satzteile[0]); i++)
  {
    Serial.print(Satzteile[i]);
  }
}
```

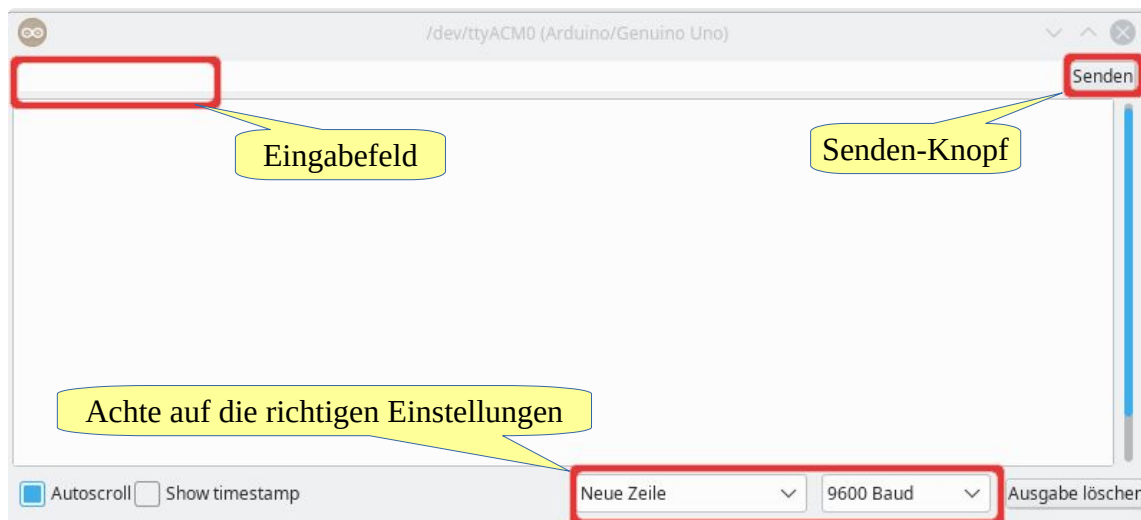


Für diese Aufgabe musst du den Arduino nur über USB verbinden und den Seriellen Monitor öffnen.

So soll es aussehen:



Verwende das Eingabefeld im Seriellen Monitor:



Im setup-Block musst du den Seriellen Monitor starten und einen Hinweis für die Eingabe geben:

```
void setup()
{
  Serial.begin(9600);
  Serial.println("Bitte Text eingeben:");
}
```

Im loop-Teil musst du zunächst die Zähler für die Vokale definieren und auf 0 setzen ...

```
void loop()
{
  // alle Zähler auf 0 setzen
  int ZaehlerA = 0;
  int ZaehlerE = 0;
  int ZaehlerO = 0;
  int ZaehlerU = 0;
  int ZaehlerI = 0;
```


... dann auf die Eingabe warten:


```
while (Serial.available() > 0)
{
  // solange lesen, bis return eingegeben wurde
  String Eingabe = Serial.readStringUntil("\n");

  // das letzte Zeichen ist return → soll entfernt werden (-1)
  Eingabe = Eingabe.substring(0, Eingabe.length() - 1);
```

Damit Groß- und Kleinbuchstaben erfasst werden, wird der Text in Kleinbuchstaben umgewandelt und dann auf die Anzahl der Vokale untersucht:

```
Eingabe.toLowerCase();

for (int i = 0; i < Eingabe.length(); i ++)
{
  // Vokale zählen
  if (Eingabe[i] == 'a') ZaehlerA ++;
  if (Eingabe[i] == 'e') ZaehlerE ++;
  if (Eingabe[i] == 'i') ZaehlerI ++;
  . . . 
  . . .
}

// Ergebnis anzeigen
// \u00e4lt → ä
Serial.println("Der Text enth\u00e4lt " + String(ZaehlerA) + " mal das a");
Serial.println("Der Text enth\u00e4lt " + String(ZaehlerE) + " mal das e");
. . . 
. . .


// Anzahl der Zeichen anzeigen
Serial.println(String(Eingabe.length()) + " Zeichen!")
}
}
```

Zusätzlich kannst du auch alle Vokale durch einen anderen Buchstaben ersetzen.
 Ergänze das Programm nach der Ausgabe der gezählten Vokale.
 Im Beispiel werden mit `replace` alle Vokale durch „i“ ersetzt.

Vorher musst du den ursprünglichen Text sichern:

```
// Original-Text sichern
String OriginalEingabe = Eingabe;

// alle Vokale durch i ersetzen
OriginalEingabe.replace("a", "i");
OriginalEingabe.replace("e", "i");

// i o und u ersetzen
. . .
. . . 
. . .
Serial.println(OriginalEingabe);
```

Du kannst den Satz auch rückwärts anzeigen lassen. Leider funktioniert das nicht mit ß.

Allerdings musst du vorher den String in ein char-Array umwandeln:

```
/*
  Eingabe rückwärts ausgeben
  Array Satz erstellen
  Anzahl der Elemente entspricht der Anzahl der Buchstaben
*/
char Satz[Eingabe.length()];
// char-Array Satz erstellen
Eingabe.toCharArray(Satz, Eingabe.length() + 1);

// Array rückwärts anzeigen
for (int i = sizeof(Satz); i >= 0; i--)
{
  Serial.print(Satz[i]);
}
```