V. SOFTWAREKONZEPT

V.I. BIBLIOTHEKEN

|  |  |
| --- | --- |
| **Name** | **Funktion** |
| Wire.h | Für die Kommunikation mit I2C Geräten |
| LiquidCrystal\_I2C.h | Um über i2c mit dazugehörigem LCD zu kommunizieren |
| SPI.h | Um mit SPI Geräten zu kommunizieren |
| SD.h | Um auf die SD zu schreiben bzw. von dieser zu lesen |

V.II. VARIABLEN

|  |  |  |
| --- | --- | --- |
| Name | Typ | Funktion |
| maximaleAnzahlPersonen | int | Maximal im Raum erlaubte Anzahl an Personen |
| aktuelleAnzahlPersonen | int | Aktuelle Personenzahl im Raum |
| zeitMesser | float | Wird als Zeitstempel genutzt |
| zeitSprung | int | Zeitabstand zwischen Durchläufen. In Millisekunden |
| raumNummer | String | Gibt Raumnummer wieder |
| Status | String | Gibt aus, ob noch Platz ist oder sich jemand vor den mittleren Sensoren stellt. |
| anzahlAenderung | char | Gibt im Datensatz an, ob jemand hinzugekommen ist oder rausgegangen ist |
| personenZahlGesamt | int | Insgesamt vorbeigegangene Personen |
| personenZahlRein | int | Insgesamt reingegangene Personen |
| personenZahlRaus | int | Insgesamt rausgegangene Personen |
| modulSDPin | int | Variable für SD Datenpin |
| myFile | File | Für Zugriff auf Dateien |
| Dateiname | String | Um auf bestimmte Datei zuzugreifen |
| pinSensorRein | int | Pin für Sensor fürs reingehen |
| pinSensorRaus | int | Pin für Sensor fürs rausgehen |
| pinSensorMitte | int | Pin für Sensor in der Mitte |
| fertig | int | Variable um Änderung der Personenzahl zu bestätigen |

V.III. UNTERFUNKTIONEN

V.III.I. Beschreibung

Neben dem Setup- sowie Loopteils hat das Programm insgesamt zwei Unterfunktionen. Die erste, vom Typ void und dem Namen Speichern(), dient der Speicherung der Datensätze. Eine zweite Unterfunktion, vom Typ void und dem Namen zeitMesserErhoehung(), dient zur Erhöhung des Zeitstempels.

V.III.II.I. Speichern

void Speichern() //Speicherfunktion  
{  
 myFile = SD.open(Dateiname, FILE\_WRITE); //SD objekt zuweisen  
 myFile.print(zeitMesser, 3); //zeitMesser mit 3 nachkommastellen in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(maximaleAnzahlPersonen); //maximaleAnzahlPersonen in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(aktuelleAnzahlPersonen); //aktuelleAnzahlPersonen in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(anzahlAenderung); //anzahlAenderung in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(personenZahlGesamt); //personenZahlGesamt in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(personenZahlRaus); //personenZahlRaus in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(personenZahlRein); //personenZahlRein in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(Status); //Status in Datei schreiben  
 myFile.println(";"); //Semilkolon in Datei schreiben  
 myFile.close(); //Datei schließen  
}

V.III.II.II. Zeitstempel erhöhen

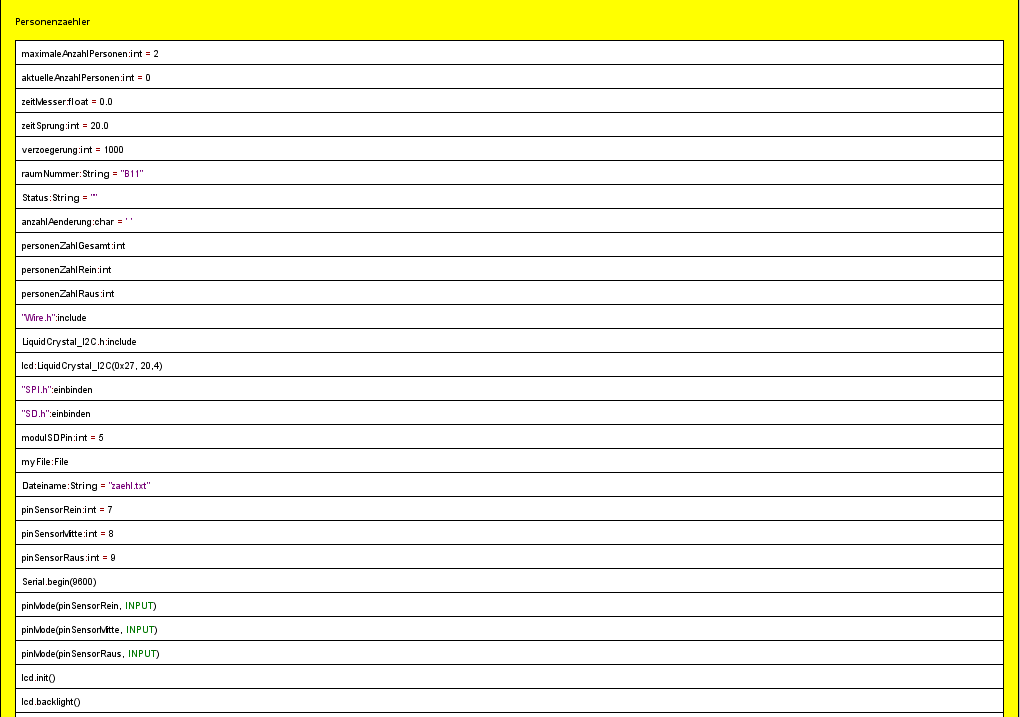
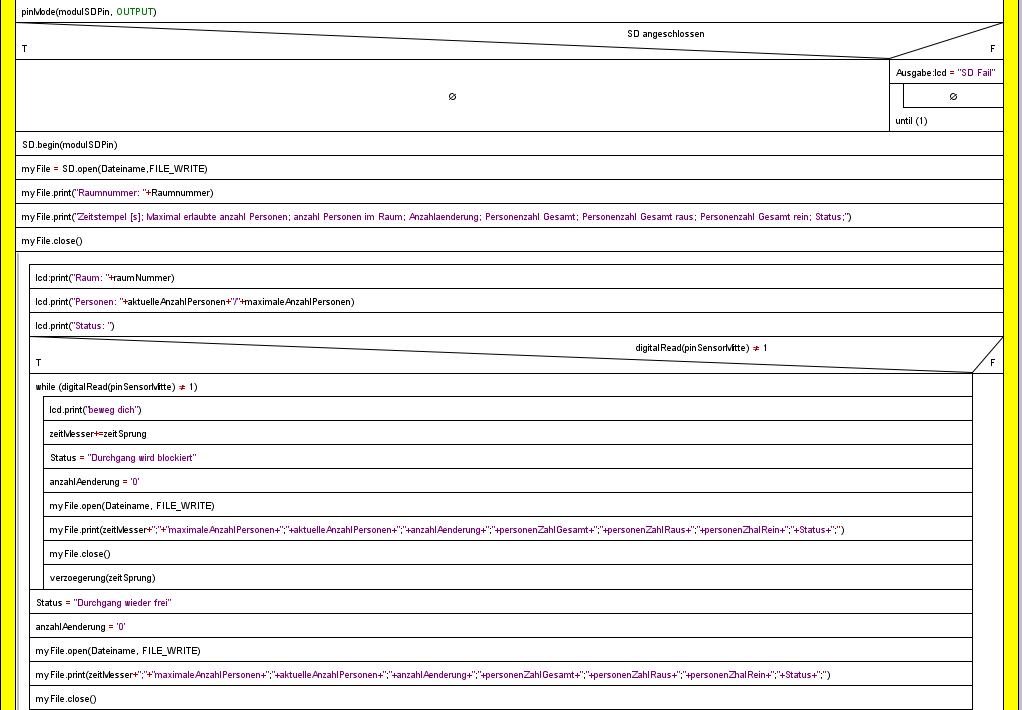
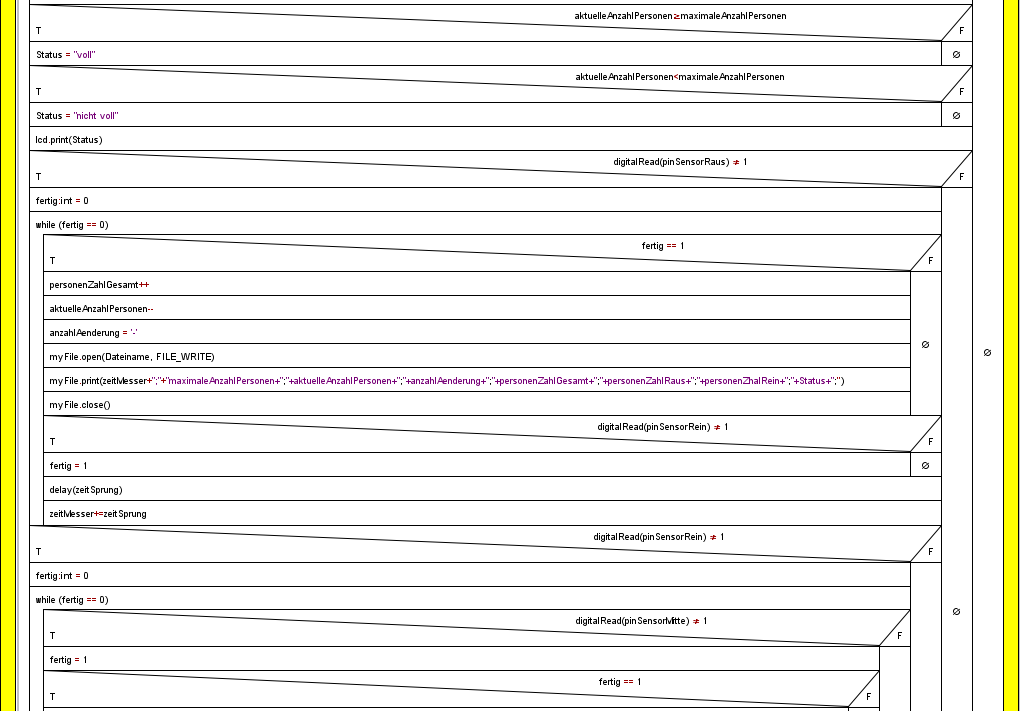
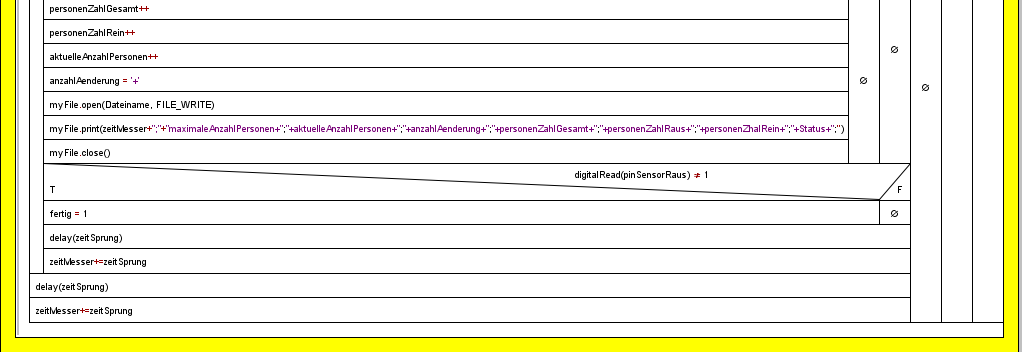
void zeitMesserErhoehung() //Funktion um zeitMesser um zeitSprung Sekunden erhoehen

{

zeitMesser = zeitMesser+((float)zeitSprung/1000); //zeitMesser um zeitSprung Sekunden

//erhöhen

}

V.IV. STRUKTOGRAMM  
  
  
  


Siehe

V.VI. PROGRAMMCODE  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//organisatorisches

int maximaleAnzahlPersonen = 2; //gibt maximale anzahl an personen im raum an. ausgabe in 2. zeile. wert nach

//schraegstrich

int aktuelleAnzahlPersonen = 0; //gibt aktuelle anzahl an personen im raum an. ausgabe in 2. zeile. wert vor schraegstrich  
float zeitMesser = 0.0; //zeitstempel in ms  
int zeitSprung = 20; //genauigkeit in ms. nicht auf 0 setzen. arduino wird sonst ueberhitzen weil er zu schnell

//arbeitet

String raumNummer = "B11"; //raumnummerangabe. in 1. Zeile  
String Status = ""; //wert der ausgibt ob im raum noch platz ist. ausgabe in 4. Zeile  
char anzahlAenderung = ' '; //gibt an ob eine person hinzugekommen ist oder rausgegangen ist

int personenZahlGesamt; //insgesamt vorbeigegangene Personen  
int personenZahlRein; //insgesamt reingegangene Personen  
int personenZahlRaus; //insgesamt rausgegangene Personen

//Einrichtung des 4 zeilen lcds  
#include <Wire.h> // Library for I2C communication  
#include <LiquidCrystal\_I2C.h> // bibliothek für i2c lcd  
LiquidCrystal\_I2C lcd = LiquidCrystal\_I2C(0x27, 20, 4); //lcd objekt erstellen  
//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//SD Bezogen  
#include "SPI.h" //erlaubt kommunikation mit spi geräten  
#include "SD.h" //r/w auf sd   
int modulSDPin = 5; //sd modul pin  
File myFile; //fuer zugriff auf dateien  
String Dateiname = "zaehl3.txt"; //um auf bestimmte datei zuzugreifen. txt weil der csv nicht mag  
//\_\_\_\_\_\_\_\_\_\_  
//einrichtung der sensorpins  
int pinSensorRein = 7; //pin Reinsensor fuer daten.  
int pinSensorMitte = 8; //pin mittelsensor fuer bestaetigung. wird genutzt um andere sensorbetaetigungen zu aktivieren  
int pinSensorRaus = 9; //pin Raussensor fuer daten.  
//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

void setup() {  
 //baudrate  
 Serial.begin(9600); //baudrate festlegen  
 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//sensoren einschalten  
 pinMode(pinSensorRein, INPUT); //Reinsensor als eingabegeraet festlegen  
 pinMode(pinSensorMitte, INPUT); //Mittleren Sensor als eingabegeraet festlegen  
 pinMode(pinSensorRaus, INPUT); //Raussensor als eingabegeraet festlegen  
 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//LCD starten  
 lcd.init(); //lcd initiieren  
 lcd.backlight(); //lcd hintergrundbeleuchtung anmachen   
 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//SD einrichtung und erste zeilen mit beschreibung fuellen  
 pinMode(modulSDPin, OUTPUT); //sd als ausgabegeraet festlegen  
 if(!SD.begin(modulSDPin)) //wenn sd nicht dran ist oder aehnlich  
 {  
 Serial.print("SD Fail"); //fehlermeldung ausgeben  
 lcd.setCursor(0,3); //vierte zeile, erste spalte  
 lcd.print("SD Fail"); //kennzeichnung fehlschlag  
 while(1); //dauerschleife  
 }  
 SD.begin(modulSDPin); //sd modul starten  
 myFile = SD.open(Dateiname, FILE\_WRITE); //dateiobjekt oeffnen bzw. zuweisen  
 myFile.print("Raumnummer: ;"); //raumnummer kennzeichnung eintragen  
 myFile.print(raumNummer); //raumnummer eintragen  
 myFile.println(";"); //semikolon eintragen  
 myFile.println("Zeitstempel [s]; Maximal erlaubte anzahl Personen; anzahl Personen im Raum; Anzahlaenderung; Personenzahl Gesamt; Personenzahl Gesamt raus; Personenzahl Gesamt rein; Status;");

//Datei erste Zeile einfuegen fuer kennzeichnung der folgenden werte

myFile.close(); //dateiobjekt schließen  
}

void loop() { //durchgehend wiederholen bis benutzerdefinierter abbruch  
 //LCD ausgabeeinrichtung  
 //Raumnummer  
 lcd.setCursor(0, 0); // erste zeile, erste spalte  
 lcd.print("Raum: "); // Kennzeichnung Raumnummer  
 lcd.setCursor(10, 0); // erste zeile, 7. Spalte  
 lcd.print(raumNummer); //Raumnummer angeben  
 //\_\_\_\_\_\_\_\_\_\_\_

//Personenanzahl  
 lcd.setCursor(0,1); //zweite zeile, erste Spalte  
 lcd.print("Personen: "); //Kennzeichnung Personenzahl  
 lcd.setCursor(10,1); //zweite zeile, 9. Spalte  
 lcd.print(" "); //feld leer machen   
 lcd.setCursor(10,1); //zweite zeile, 9. Spalte  
 lcd.print(aktuelleAnzahlPersonen); //anzahl personen aktuell  
 lcd.setCursor(12,1); //zweite zeile, 13. Spalte  
 lcd.print("/"); //trennzeichen  
 lcd.setCursor(13,1); //zweite zeile, 14. Spalte  
 lcd.print(maximaleAnzahlPersonen);//anzahl Personen Maximal  
 //\_\_\_\_\_\_\_\_\_\_\_\_

//Ausgabe Status  
 lcd.setCursor(0,3); //vierte zeile, erste spalte  
 lcd.print("Status: "); //kennzeichnung status

//raumfuelle abfrage Sensor  
 //mitte, falls jemand sich einfach hinstellt  
 if(digitalRead(pinSensorMitte) !=1) //Wenn der mittleresensor ein hindernis hat  
 {  
 while(digitalRead(pinSensorMitte) !=1) //solange der mittlere Sensor ein hindernis hat  
 {  
 lcd.setCursor(0,2); //dritte zeile, erste spalte  
 lcd.print("beweg dich"); //ausgabe zur erinnerung, dass der sensor immernoch in betrieb ist  
 zeitMesserErhoehung(); //zeitstempel um zeitSprung Millisekunden erhoehen.   
 Status = "Durchgang wird blockiert"; //Status auf wird blockiert setzen  
 anzahlAenderung = '0'; //Status char auf 0 abändern  
 Speichern(); //Speicherfunktion abrufen  
 delay(zeitSprung); //abfrage um zeitSprung millisekunden verzoegern  
 }

lcd.setCursor(0,2); //dritte zeile, erste spalte  
 lcd.print(" "); //lcd bereich leeren  
 Status = "Durchgang wieder frei"; //status setzen  
 anzahlAenderung = '0'; //status cahr auf 0 abändern  
 Speichern(); //Speicherfunktion abrufen  
 }

//raumfuelle abfrage  
 if(aktuelleAnzahlPersonen>=maximaleAnzahlPersonen) //wenn mehr im raum sind als erlaubt  
 {  
 Status = "voll"; //Statusaenderung auf voll  
 }

if(aktuelleAnzahlPersonen<maximaleAnzahlPersonen) //wenn weniger im raum sind als erlaubt  
 {  
 Status = "nicht voll"; //Statusaenderung auf nicht voll  
 }  
 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

//Status ausgeben  
 lcd.setCursor(10,3); //vierte zeile, 11. Spalte  
 lcd.print(" "); //Leeren  
 lcd.setCursor(10,3); //vierte zeile, 11. Spalte  
 lcd.print(Status); //ausgabe Status  
 //\_\_\_\_\_\_\_\_

//aussen  
 if(digitalRead(pinSensorRaus) !=1) //wenn der aussensensor ausgeloest wurde beginne prozess zur

//personenzahlreduzierung

{  
 int fertig = 0; //variable um personenreduzierung zu bestaetigen  
 while(fertig == 0) //solange bestaetiger 0 ist mache schleife  
 {  
 if(digitalRead(pinSensorMitte) != 1) //wenn bestaetigung mittlerer Sensor  
 {  
 fertig = 1; //schleifenbedingung erfuellt  
 if(fertig == 1) //wenn schleifenbedingung erfuellt  
 {  
 personenZahlGesamt++; //globalen personenzaehler erhoehen  
 if(aktuelleAnzahlPersonen>0) //wenn aktuelle personenzahl groesser als 0  
 {  
 personenZahlRaus++; //insgesamt rausgegangene Personenzahl erhoehen.  
 aktuelleAnzahlPersonen--; //reduziere personenzahl  
 anzahlAenderung = '-';  
 Speichern();  
 }  
 }  
 }  
 if(digitalRead(pinSensorRein) != 1)//wenn statt dem mittleren sensor der reingehsensor betaetigt wird  
 {  
 fertig = 1; //schleifenbedingung erfuellt  
 }  
 delay(zeitSprung); //abfrage um zeitSprung millisekunden verzoegern  
 zeitMesserErhoehung(); //zeitstempel um zeitSprung sekunden erhoehen.  
 }

}

//innen  
 if(digitalRead(pinSensorRein) !=1) //wenn der innensensor ausgeloest wurde beginne prozess zur

//personenzahlerhoehung

{  
 int fertig = 0; //variable um personenzahlsteigung zu bestaetigen  
 while(fertig == 0) //solange bestaetiger 0 ist mache schleife  
 {  
 if(digitalRead(pinSensorMitte) != 1) //wenn bestaetigung mittlerer Sensor  
 {  
 fertig = 1; //schleifenbedingung erfuellt  
 if(fertig == 1) //wenn schleifenbedingung erfuellt  
 {  
 personenZahlGesamt++; //globalen personenzaehler erhoehen  
 personenZahlRein++; //insgesamt reingegangene Personenzahl erhoehen  
 aktuelleAnzahlPersonen++; //erhoehe personenzahl  
 anzahlAenderung = '+'; //Statuschar auf + abaendern  
 Speichern(); //Speicherfunktion abrufen  
 }  
 }  
 if(digitalRead(pinSensorRaus) != 1)//wenn statt dem mittleren sensor der reingehsensor betaetigt wird  
 {  
 fertig = 1; //schleifenbedingung erfuellt  
 }  
 delay(zeitSprung); //abfrage um zeitSprung millisekunden verzoegern  
 zeitMesserErhoehung(); //zeitstempel um zeitSprung sekunden erhoehen.  
 }  
 }  
 //\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
 delay(zeitSprung); //verzoegerung um zeitSprung Millisekunden  
 zeitMesserErhoehung(); //zeitMesser um zeitSprung erhoehen;  
}  
void zeitMesserErhoehung() //Funktion um zeitMesser um zeitSprung Sekunden erhoehen  
{  
 zeitMesser = zeitMesser+((float)zeitSprung/1000);//zeitMesser um zeitSprung Sekunden erhöhen  
}  
void Speichern() //Speicherfunktion  
{  
 myFile = SD.open(Dateiname, FILE\_WRITE); //SD objekt zuweisen  
 myFile.print(zeitMesser, 3); //zeitMesser mit 3 nachkommastellen in Datei schreiben  
 myFile.print(";"); //Semilkolon in Datei schreiben  
 myFile.print(maximaleAnzahlPersonen); //maximaleAnzahlPersonen in Datei schreiben  
 myFile.print(";"); //Semikolon in Datei schreiben  
 myFile.print(aktuelleAnzahlPersonen); //aktuelleAnzahlPersonen in Datei schreiben  
 myFile.print(";"); //Semikolon in Datei schreiben  
 myFile.print(anzahlAenderung); //anzahlAenderung in Datei schreiben  
 myFile.print(";"); //Semikolon in Datei schreiben  
 myFile.print(personenZahlGesamt); //personenZahlGesamt in Datei schreiben  
 myFile.print(";"); //Semikolon in Datei schreiben  
 myFile.print(personenZahlRaus); //personenZahlRaus in Datei schreiben  
 myFile.print(";"); //Semikolon in Datei schreiben  
 myFile.print(personenZahlRein); //personenZahlRein in Datei schreiben  
 myFile.print(";"); //Semikolon in Datei schreiben  
 myFile.print(Status); //Status in Datei schreiben  
 myFile.println(";"); //Semikolon in Datei schreiben  
 myFile.close(); //Datei schließen  
}  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_