



WETTERSTATION TEIL 2

22. Februar 2022:

INTRO

JSON DATEN IM WEB

DARC.DE/Q01/TREFF

DEIN CLUB - AMATEURFUNK VERBINDET

Das ganze ist ein wachsendes Projekt, so dass sich diese Dokumentation ändert und dazugehörige Hand-outs stetig erweitern werden, um so im Laufe des Jahres zu einem ausgiebigen Handbuch mit Ausbaubeschreibung zu kommen.

Wir betreiben die Clubstation

DL0GS

Im Rahmen der Modernisierung wird diese um diverse Komponenten und Services ergänzt, wie z.B. die

virtuelle Clubstation unter <http://q01.darc.de/apps> mit **Wetter**

Letztes Mal:

- Wetterstation
- RTL-SDR Stick
- Raspberry Pi
- rtl_433 kompiliert
- Wetterdatenerfassung

Wetter – Wetter – Wetter

- HURRA !!!!

```
rtl_433 version 20.02-16-g06e8746 branch master at 202003161612 inputs file rtl_tcp RTL-SDR
Use -h for usage help and see https://triq.org/ for documentation.
Trying conf file at "rtl_433.conf"...
New defaults active, use "-Y classic -s 250k" for the old defaults!
```

```
Registered 122 out of 150 device decoding protocols [ 1-4 8 11-12 15-17 19-21 23 25-26 29-36 38-60 63 67-71 73-100 102-105 108-116 119 121 124-128 130-149 ]
Found Rafael Micro R828D tuner
[R82XX] PLL not locked!
Sample rate set to 1024000 S/s.
Tuner gain set to Auto.
Tuned to 868.000MHz.
```

```
-----
time      : 2022-01-19 17:31:12
model    : Bresser-5in1 id      : 12
Temperature: 21.2 C      Humidity : 38      Wind Gust : 0.0 m/s      Wind Speed: 0.0 m/s      Direction : 247.5 °      Rain       : 0.0 mm      Integrity : CHECKSUM
```

```
-----
time      : 2022-01-19 17:31:48
model    : Bresser-5in1 id      : 12
Temperature: 21.3 C      Humidity : 38      Wind Gust : 0.8 m/s      Wind Speed: 0.2 m/s      Direction : 270.0 °      Rain       : 0.0 mm      Integrity : CHECKSUM
```

- Webserver am Pi (lighttpd)
- kleine Einführung: HTML & JavaScript
- JSON-Daten im Webbrowser

- Fragen

Vorbereiten des Raspberry PI

- Webserver (lighttpd)
- Super-User werden (sudo su)

```
pi@raspberrypi:~$ sudo apt-get install lighttpd
```

```
.....
```

```
pi@raspberrypi:~$ sudo su
```

```
root@raspberrypi:/home/pi#
```

Webbrowser /

403 Forbidden

http://raspberrypi/

403 Forbidden

Webseite am Pi

- Unterverzeichnisse wechseln (cd), erstellen (mkdir)
- Inhalt auflisten (ls)

```
root@raspberrypi:/home/pi# cd /var/www/html/  
root@raspberrypi:/var/www/html# mkdir wetter  
root@raspberrypi:/var/www/html# cd wetter  
root@raspberrypi:/var/www/html/wetter# ls
```

Webseite am Pi

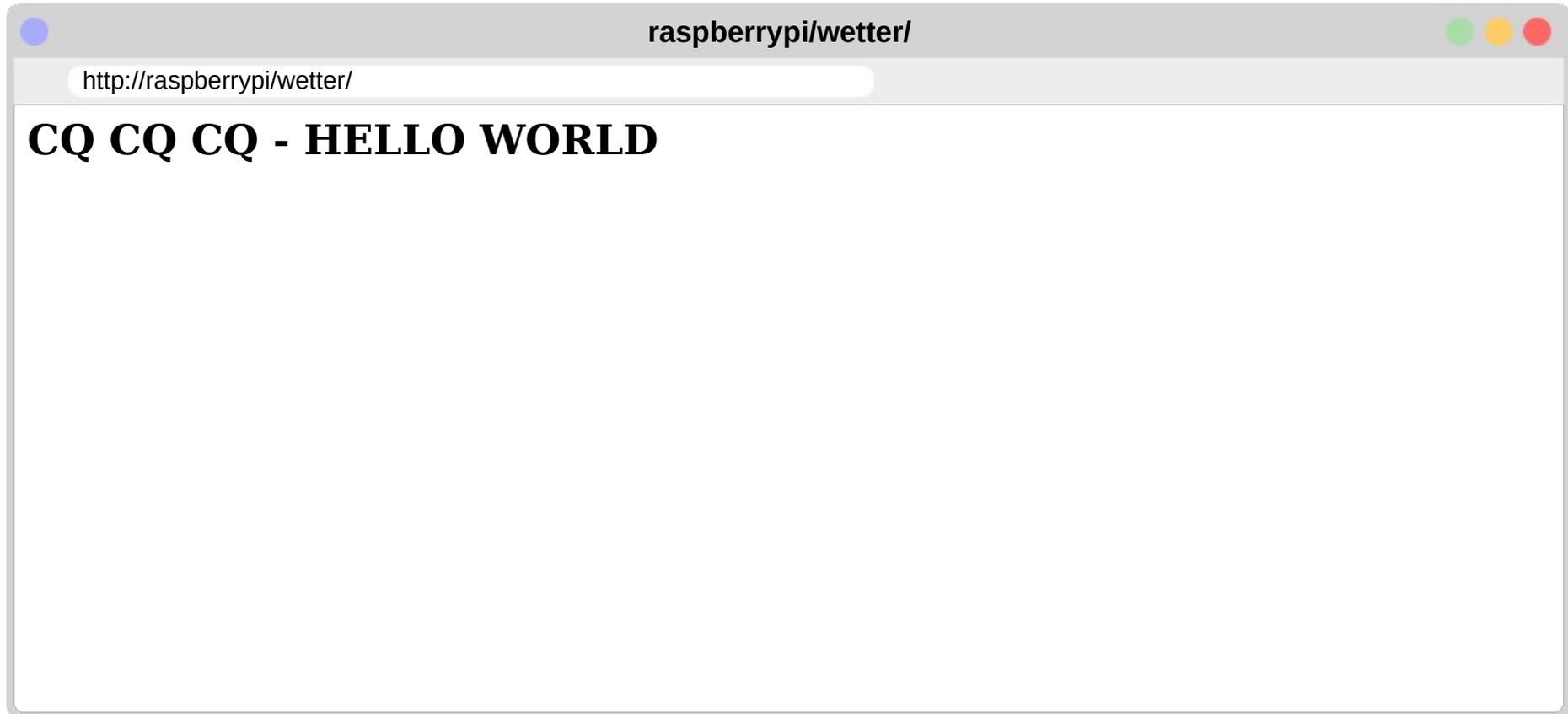
- index.html schreiben (cat > index.html)
- index.html anzeigen (cat index.html)

```
root@raspberrypi:/var/www/html/wetter# cat > index.html
<html>
  <body>
    <h1>CQ CQ CQ - HELLO WORLD</h1>
  </body>
</html>
^C   + 

root@raspberrypi:/var/www/html/wetter# cat index.html
```

.....

Webbrowser /wetter/



HTML Syntax

HTML ANFANG
KOPF ANFANG
 ZEICHENSATZ (ÄÖÜ)
 TITEL DER SEITE
KOPF ENDE
KÖRPER ANFANG
KOMMENTAR
ZENTRIEREN
 ÜBERSCHRIFT erster Kategorie
 ÜBERSCHRIFT zweiter Kategorie
 INHALTSEBENE
 INHALT
 INHALTSEBENE ENDE
ZENTRIEREN ENDE
KÖRPER ENDE
HTML ENDE

```
<html>  
  <head>  
    <meta charset="UTF-8">  
    <title>HOMEPAGE mit Wetter</title>  
  </head>  
  <body>  
    <!-- wird nicht dargestellt -->  
    <center>  
      <h1>CQ CQ CQ - Hello World</h1>  
      <h2>Wetter</h2>  
      <div id="weatherdata">  
        WETTERDATEN  
      </div>  
    </center>  
  </body>  
</html>
```

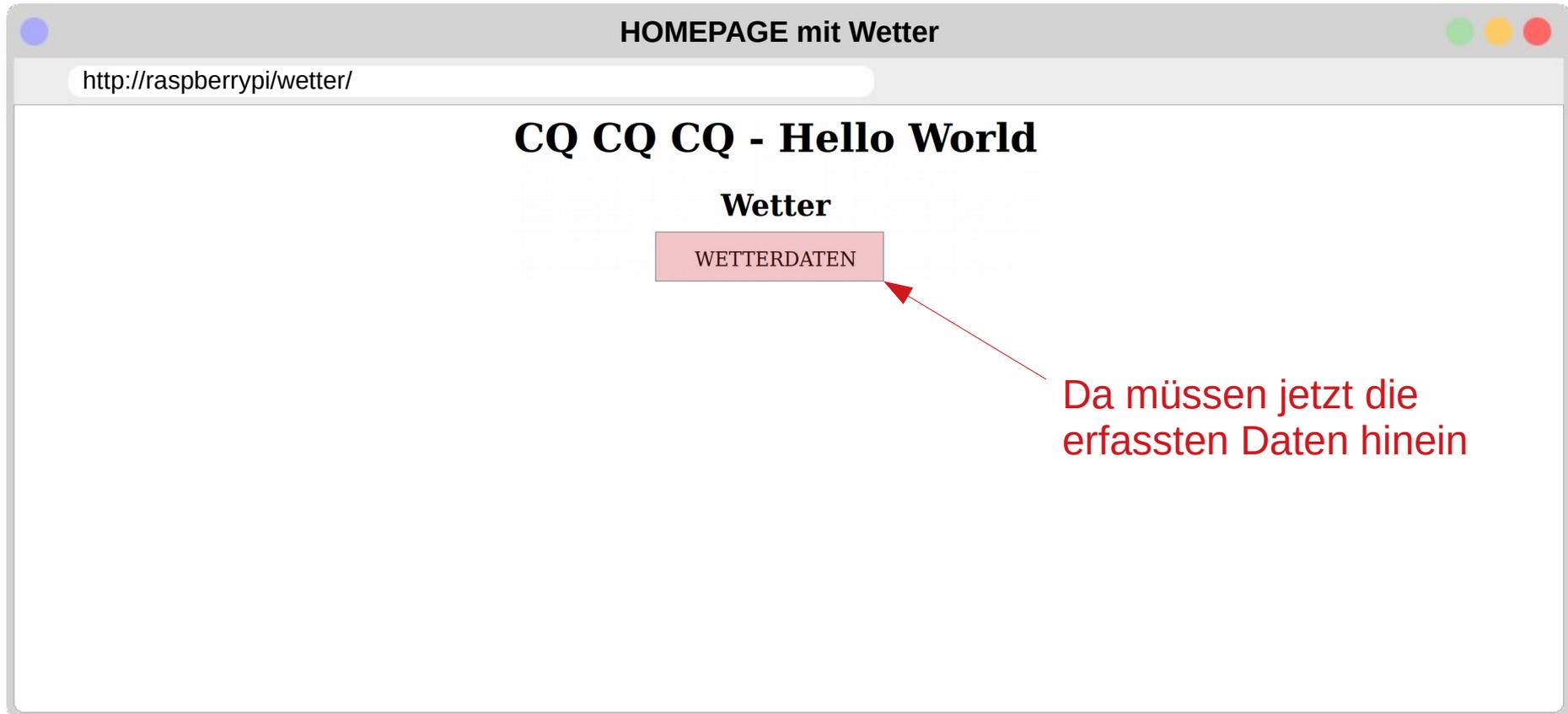
HTML (für copy & paste)

```
<html>
  <head>
    <meta charset="UTF-8">
    <title>HOMEPAGE mit Wetter</title>
  </head>
  <body>
    <!-- wird nicht dargestellt -->
    <center>
      <h1>CQ CQ CQ - Hello World</h1>
      <h2>Wetter</h2>
      <div id="weatherdata">
        WETTERDATEN
      </div>
    </center>
  </body>
</html>
```

Webbrowser /wetter/



Webbrowser /wetter/



The screenshot shows a web browser window titled "HOMEPAGE mit Wetter". The address bar contains "http://raspberrypi/wetter/". The main content area displays "CQ CQ CQ - Hello World" in large bold text. Below this, there is a section titled "Wetter" which contains a light blue rectangular box with the text "WETTERDATEN" inside. A red arrow points from the text "Da müssen jetzt die erfassten Daten hinein" to the "WETTERDATEN" box.

- **JavaScript**

JavaScript (kurz JS) ist eine Skriptsprache, die ursprünglich 1995 von Netscape für dynamisches HTML in Webbrowsern entwickelt wurde, um Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren und so die Möglichkeiten von HTML zu erweitern.

Heute wird JavaScript auch außerhalb von Browsern angewendet, etwa auf Servern und in Microcontrollern.

- JavaScript

JavaScript (kurz JS) ist eine Skriptsprache, die ursprünglich 1995 von Netscape für dynamisches HTML in Webbrowsern entwickelt wurde, um Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren und so die Möglichkeiten von HTML zu erweitern.

Heute wird JavaScript auch außerhalb von Browsern angewendet, etwa auf Servern und in Microcontrollern.



JavaScript != Java
Das sind 2 völlig unterschiedliche Dinge !!

- **JavaScript**

JavaScript (kurz JS) ist eine Skriptsprache, die ursprünglich 1995 von Netscape für dynamisches HTML in Webbrowsern entwickelt wurde, um Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren und so die Möglichkeiten von HTML zu erweitern.

Heute wird JavaScript auch außerhalb von Browsern angewendet, etwa auf Servern und in Microcontrollern.

- **JSON**

Die JavaScript Object Notation (JSON ['dʒeɪsən]) ist ein kompaktes Datenformat in einer einfach lesbaren Textform für den Datenaustausch zwischen Anwendungen. JSON ist von Programmiersprachen unabhängig. Parser und Generatoren existieren in allen verbreiteten Sprachen.

- JavaScript

JavaScript (kurz JS) ist eine Skriptsprache, die ursprünglich 1995 von Netscape für dynamisches HTML in Webbrowsern entwickelt wurde, um Benutzerinteraktionen auszuwerten, Inhalte zu verändern, nachzuladen oder zu generieren und so die Möglichkeiten von HTML zu erweitern.

Heute wird JavaScript auch außerhalb von Browsern angewendet, etwa auf Servern und in Microcontrollern.

- JSON

Die JavaScript Object Notation (JSON ['dʒeɪsən]) ist ein kompaktes Datenformat in einer einfach lesbaren Textform für den Datenaustausch zwischen Anwendungen. JSON ist von Programmiersprachen unabhängig. Parser und Generatoren existieren in allen verbreiteten Sprachen.

```
{  
  "time" : "2022-02-09 18:42:51", "model" : "Bresser-5in1", "id" : 113, "battery_ok" : 1,  
  "temperature_C" : 7.500, "humidity" : 70, "wind_max_m_s" : 0.000, "wind_avg_m_s" : 0.000, "wind_dir_deg" : 292.500, "rain_mm" : 1.200,  
  "mic" : "CHECKSUM"  
}
```

JavaScript in index.html

- intern im <head>

```
<head>  
.....  
    <script>  
        alert('Hello world');  
    </script>  
</head>
```

JavaScript in index.html

- intern im <head>

```
<head>
.....
  <script>
    alert('Hello world');
  </script>
</head>
```

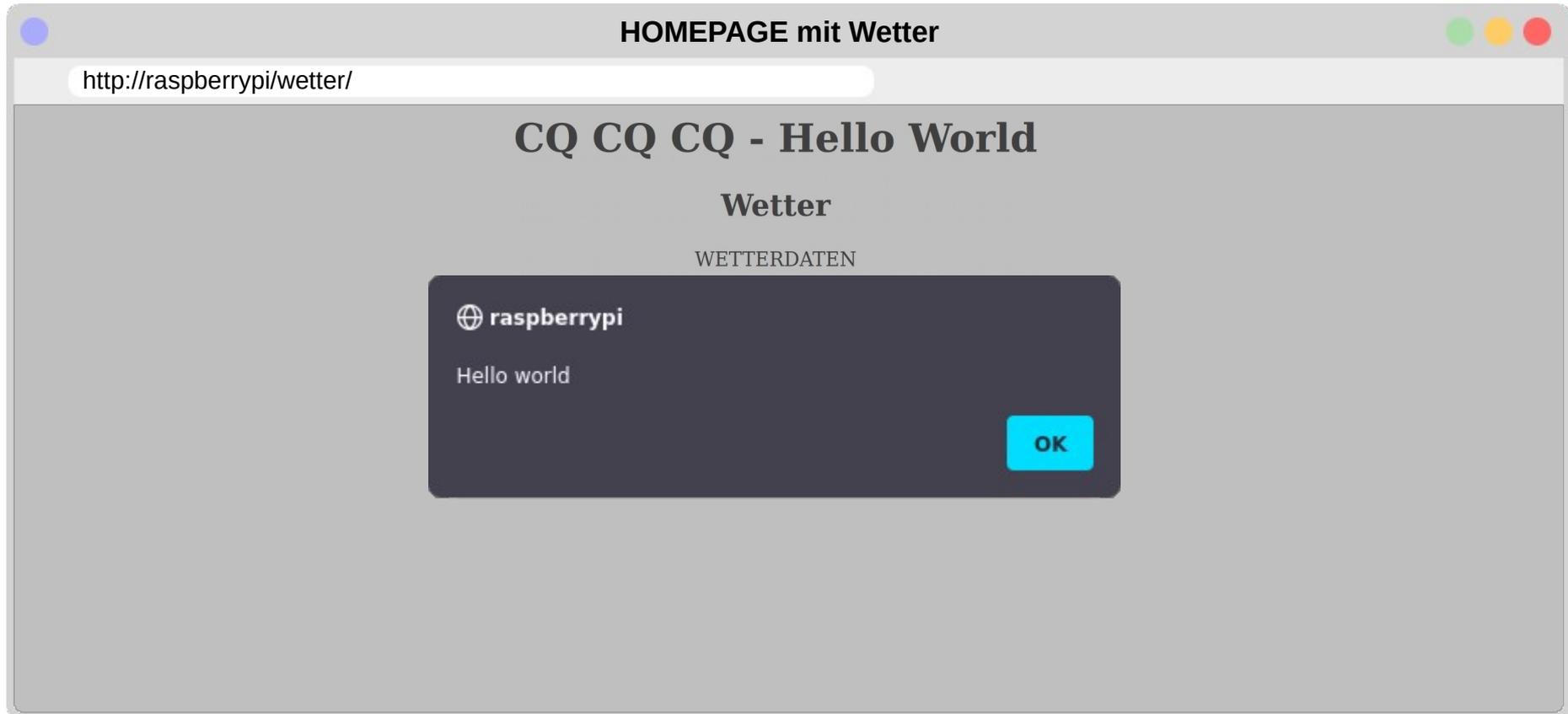
- externe Datei

```
<head>
.....
  <script src="script.js"></script>
</head>
```

script.js

```
alert('Hello world');
```

Webbrowser mit JS (alert)



JavaScript Syntax

WENN FENSTER GELADEN
DIALOGBOX mit "Hello world"
schreibe "NEU" in INHALTSEBENE

```
window.onload = function() {  
    alert('Hello world');  
    document.getElementById('weatherdata').innerHTML='NEU';  
}
```

JavaScript (für copy & paste)

```
window.onload = function() {  
  alert('Hello world');  
  document.getElementById('weatherdata').innerHTML='NEU';  
}
```

JavaScript mal ernsthaft

```
var intervalTime=5;
var fileName='data.json';

window.onload = function() {
  start_weather();
}

function weather_show(param) {
  document.getElementById('weatherdata').innerHTML=param;
}

function weather_fetch(param) {
  console.log('Hole: '"+param+"'");
  fetch(param)
    .then(response => response.json())
    .then(data => weather_show(data));
}

function start_weather() {
  weather_fetch(fileName+'?utime='+Date.now());
  setTimeout(start_weather, intervalTime*1000);
}
```

Wetteranzeige-Funktion

Zur Erinnerung: unser JSON-Datensatz:

```
{  
  "time" : "2022-02-09 18:42:51", "model" : "Bresser-5in1", "id" : 113, "battery_ok" : 1,  
  "temperature_C" : 7.500, "humidity" : 70, "wind_max_m_s" : 0.000, "wind_avg_m_s" : 0.000, "wind_dir_deg" : 292.500, "rain_mm" : 1.200,  
  "mic" : "CHECKSUM"  
}
```

```
function weather_show(param) {  
  var elem=document.getElementById('weatherdata');  
  elem.innerHTML+"<b>" + param.time+"</b>";  
  elem.innerHTML+="  
>Temp: " + param.temperature_C+" °C";  
  elem.innerHTML+="  
>Feucht: " + param.humidity+" %";  
  elem.innerHTML+="  
>Wind: " + param.wind_dir_deg+" °";  
}
```

Wettererfassung/ JSON schreiben

- SuperUser beenden (exit)
- rtl_433 im Hintergrund starten
- Daten in Schleife nach /var/www/html/wetter/data.json

```
root@raspberrypi:/var/www/html# exit
pi@raspberrypi:~$ rtl_433 -f 868M -s 1024k -F json:wetterdaten.json &
.....
pi@raspberrypi:~$ while true; do sudo /bin/bash -c "/bin/tail -n 1
wetterdaten.json > /var/www/html/wetter/data.json" && sleep 5 ; done
```

Gestalte die Seite

- nach Deinen Farbwünschen
- mit einem vollflächigen Hintergrundbild
- ändere die Schriftarten
- positioniere das Inhaltselement “weatherdata” an anderer Stelle

Benutze dazu CSS

- entweder inline,
- im <head>
- oder als externe Datei

! CSS-Dateien werden anders eingebunden als JavaScript-Dateien

- Schau im SelfHTML-Wiki
- Sag mir wie !

FRAGEN



DANKE FÜR EURE AUFMERKSAMKEIT



DAS WAR:

WETTERSTATION Teil-2

JSON DATEN IM WEB

 **DARC.DE/Q01**

SAARLAND

Großes entsteht immer im Kleinen.

