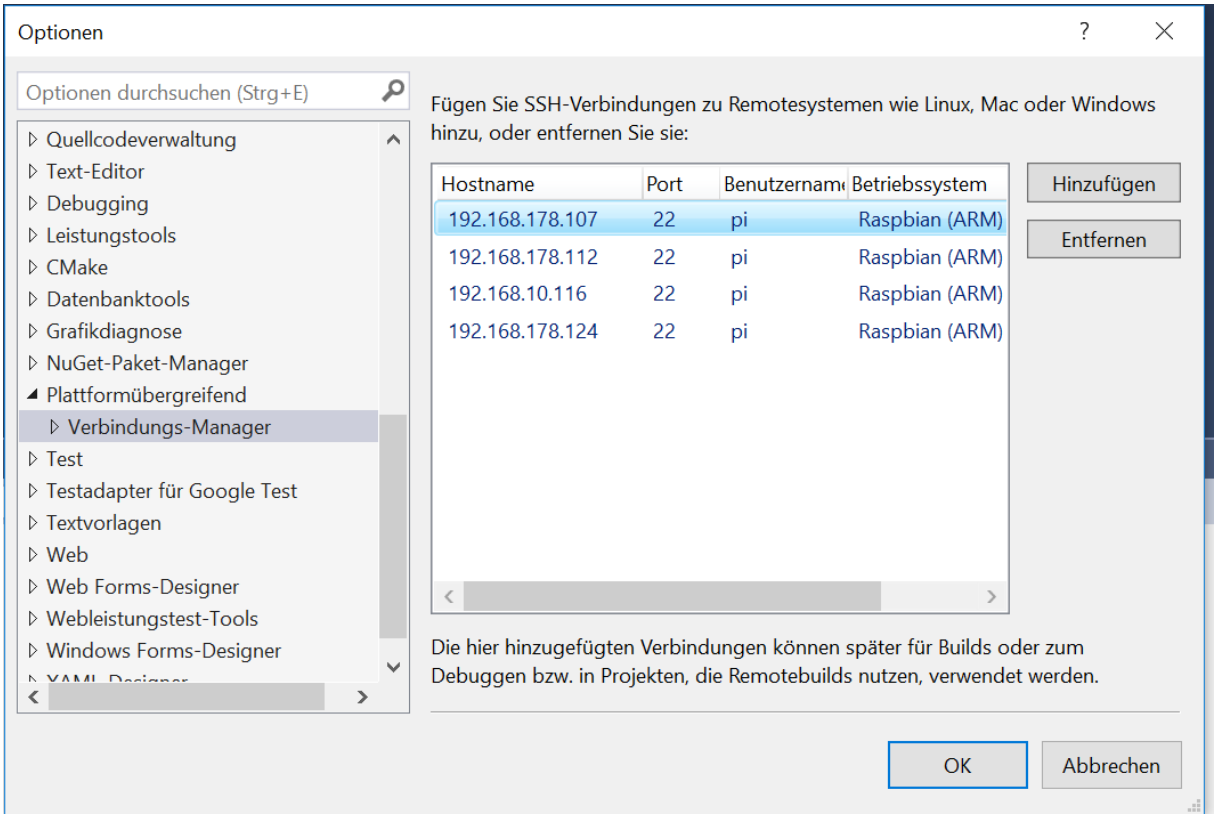
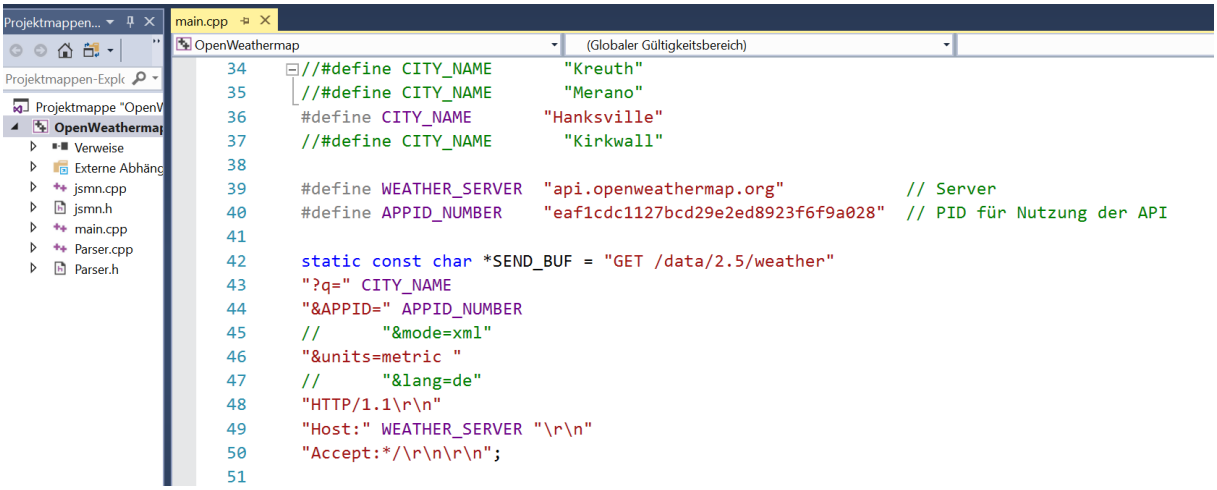
RaspberryPi und Openweathermap

Über einen RaspberryPi soll der Zugriff auf den Server Openwaeathermap erfolgen.
Das vorbereitete Projekt finden Sie auf der Homepage als zip-Datei.
Über eine API können umfangreiche Datensätze von einer Vielzahl von Wetterstationen geholt werden zugegriffen werden.

Für das Programmieren wird VisualStudio verwendet



Der folgende Ausschnitt aus main.cpp zeigt die GET Methode mit der auf den Server zugegriffen werden kann.



Wenn Sie das Projekt übersetzen und auf dem RaspberryPi starten erhalten Sie die folgende Ausgabe:

```
root@raspberrypi_d204:/home/pi#
./projects/OpenWeathermap/bin/ARM/Debug/OpenWeathermap.out
Hostname    : api.openweathermap.org
Aliase      :
IP-Adressen :
     37.139.20.5
     37.139.1.159
     82.196.7.246
host: api.openweathermap.org IP - Addy: 82.196.7.246
Socket wurde angelegt
Verbindung mit dem Server 82.196.7.246 hergestellt
Ausgabe der Wetterdaten als String :
HTTP/1.1 200 OK
Server: openresty
Date: Fri, 10 May 2019 09:16:36 GMT
Content-Type: application/json; charset=utf-8
Content-Length: 448
Connection: keep-alive
X-Cache-Key:
/data/2.5/weather?APPID=eaf1cdc1127bcd29e2ed8923f6f9a028&q=hanksvill
e&units=metric
Access-Control-Allow-Origin: *
Access-Control-Allow-Credentials: true
Access-Control-Allow-Methods: GET, POST

{"coord":{"lon":-
110.71,"lat":38.37},"weather":[{"id":804,"main":"Clouds","descriptio
n":"overcast
clouds","icon":"04n"}],"base":"stations","main":{"temp":7.49,"pressu
re":1015,"humidity":81,"temp_min":5.56,"temp_max":10},"visibility":1
6093,"wind":{"speed":2.6,"deg":30},"clouds":{"all":90},"dt":15574797
22,"sys":{"type":1,"id":4443,"message":0.0043,"country":"US","sunris
e":1557490583,"sunset":1557541330},"id":5540148,"name":"Hanksville",
"cod":200}

Wetterdaten (Auswahl) als JSON Variablen ausgeben:

          Ort: Hanksville
          Temp: 7.49 Grad
          Feuchtigkeit: 81 %
          Luftdruck: 1015 mbar
          Sonnenaufgang: 2019-05-11 04:22:10 CEST
          Sonnenuntergang: 2019-05-10 14:16:23 CEST
          Wetter: overcast clouds

root@raspberrypi_d204:/home/pi# ^C
```

- Welches Protokoll wird verwendet?
- Welcher Teil des ausgedruckten Textes gehört zur Anfrage und ab welcher Zeile beginnt die Antwort
- Wie lautet die Methode für die Anfrage?

Das Format der Serverantwort lautet JSON. Was bedeutet dieser Begriff?

Ein Parser konvertiert diese JSON Daten dann in leicht lesbares Format.

Vervollständigen Sie die Ausgabe:

OpenWeathermap (Globaler Gültigkeitsbereich)

```
105     parser(weatherJSON, &weather_info);             /*Starten des Parsers*/
106     sprintf(str,
107         "\
108         Ort: %s \n\
109         Temp: %s Grad\n\
110         Feuchtigkeit: %s %\n\
111         Luftdruck: %s mbar\n\
112         Sonnenaufgang: %s \n\
113         Sonnenuntergang: %s\n\
114         Wetter: %s\n",
115         weather_info.city.name, weather_info.temperature.value,
116         weather_info.humidity.value, weather_info.pressure.value,
117         weather_info.city.sun.sunset, weather_info.city.sun.sunrise,
118         weather_info.weather.value);
119
120     cout << "\nWetterdaten (Auswahl) als JSON Variablen ausgeben:\n\n" << str << endl;
121
```

Die Struktur finden Sie in der Datei parser.h: