

TTT: pmwf – háromnapos időjárás-előrejelzés

PCLinuxOS Magazine – 2018. július

• A szerkesztő megjegyzése: a Tip Top Tips a PCLinuxOS Magazine közel havi rovata. Időközönként foglalkozunk – és talán jobban kifejtünk – egy, a PCLinuxOS fórumáról származó ötletet. A Magazin nem fogad el a Tip Top Tips számára önállóan beküldött felvetéseket. Ha van egy tipped, akkor inkább a PCLinuxOS fórumán a Tips & Tricks részében oszd meg. A tippedet kiválaszthatjuk a PCLinuxOS Magazine-ban történő publikálásra.

-
- E havi [tippünk](#) **hakerdefo**-tól származik.
-
- A **pmwf** az **aktuális** időjárási helyzetet **háromnapos** előrejelzéssel együtt jeleníti meg, a világ szinte minden **helyére**. A következő időjárási **adatokat** jeleníti meg a **pmwf**:

-
- időjárás
- legalacsonyabb hőmérséklet
- legmagasabb hőmérséklet
- légnyomás
- páratartalom
- szélesebbesség
- széllirány

A **pmwf** igényel egy **jq** nevű csomagot, A **jq** telepítése igen könnyű. Csak töltsd le a **jq** binárist és rakd el valahova a **\$PATH**-odban, **végrehajtható** paraméterrel.

[jq binary 64-bitre](#)

[jq binary 32-bitre](#)

Ezután mentsd a **pmwf** szkriptjét valahová a **\$PATH**-odban végrehajthatóként. **#!/**

Nyisd meg a **pmwf**-et a kedvenc szövegszerkesztőddel. Jobb kattintás a fájl

```
#!/usr/bin/env bash
#####
local_latitude=
local_longitude=
local_unit=
api_key=
local_rm=$(which rm)
printf "\033c"
wget --user-agent="Mozilla/5.0 Gecko/20100101" --quiet --
timeout=30
```

```
"http://api.openweathermap.org/data/2.5/weather?lat=$local_latitude&lon=$local_longitude&units=$local_unit&mode=json&APPID=$api_key"
--output-document "/tmp/current.json"
wget --user-agent="Mozilla/5.0 Gecko/20100101" --quiet --
timeout=30
"http://api.openweathermap.org/data/2.5/forecast/daily?lat=$local_latitude&lon=$local_longitude&cnt=4&units=$local_unit&mode=json&APPID=$api_key" --output-document "/tmp/forecast.json"
printf "\033c"
if [[ ! -f "/tmp/current.json" ]]; then
    echo ""
    echo -e "Can not connect to openweathermap.org!"
    echo -e "Please check your internet connection!"
    echo ""
    exit 1
else
    place=$(jq -r '.city.name' "/tmp/forecast.json")
    nation=$(jq -r '.city.country' "/tmp/forecast.json")
    time2=$(jq -r '.list[1].dt' "/tmp/forecast.json")
    time3=$(jq -r '.list[2].dt' "/tmp/forecast.json")
    time4=$(jq -r '.list[3].dt' "/tmp/forecast.json")
    time22=$(date -d @"$time2" +%A, %d %B %Y')
    time33=$(date -d @"$time3" +%A, %d %B %Y')
    time44=$(date -d @"$time4" +%A, %d %B %Y')
    tempcurrent=$(jq -r '.main.temp' "/tmp/current.json")
    min2=$(jq -r '.list[1].temp.min' "/tmp/forecast.json")
    min3=$(jq -r '.list[2].temp.min' "/tmp/forecast.json")
    min4=$(jq -r '.list[3].temp.min' "/tmp/forecast.json")
    max2=$(jq -r '.list[1].temp.max' "/tmp/forecast.json")
    max3=$(jq -r '.list[2].temp.max' "/tmp/forecast.json")
    max4=$(jq -r '.list[3].temp.max' "/tmp/forecast.json")
    pressurecurrent=$(jq -r '.main.pressure' "/tmp/current.json")
    pressure2=$(jq -r '.list[1].pressure' "/tmp/forecast.json")
    pressure3=$(jq -r '.list[2].pressure' "/tmp/forecast.json")
    pressure4=$(jq -r '.list[3].pressure' "/tmp/forecast.json")
    humidcurrent=$(jq -r '.main.humidity' "/tmp/current.json")
    humid2=$(jq -r '.list[1].humidity' "/tmp/forecast.json")
    humid3=$(jq -r '.list[2].humidity' "/tmp/forecast.json")
    humid4=$(jq -r '.list[3].humidity' "/tmp/forecast.json")
    windcurrent=$(jq -r '.wind.speed' "/tmp/current.json")
    wind2=$(jq -r '.list[1].speed' "/tmp/forecast.json")
    wind3=$(jq -r '.list[2].speed' "/tmp/forecast.json")
    wind4=$(jq -r '.list[3].speed' "/tmp/forecast.json")
    direcurrent=$(jq -r '.wind.deg' "/tmp/current.json")
    dire2=$(jq -r '.list[1].deg' "/tmp/forecast.json")
    dire3=$(jq -r '.list[2].deg' "/tmp/forecast.json")
```

Tip Top Tips: pmwf (Poor Man's Weather Forecast) – háromnapos időjárás-előrejelzés

```
dire4=$(jq -r '.list[3].deg' "/tmp/forecast.json")
condicurrent=$(jq -r '.weather[].description'
"/tmp/current.json")
condi2=$(jq -r '.list[1].weather[].description'
"/tmp/forecast.json")
condi3=$(jq -r '.list[2].weather[].description'
"/tmp/forecast.json")
condi4=$(jq -r '.list[3].weather[].description'
"/tmp/forecast.json")
if [[ "$local_unit" == imperial ]]; then
    windcurrent1="$windcurrent"
    wind22="$wind2"
    wind33="$wind3"
    wind44="$wind4"
    symb=$(echo -e "\u2109")
    symb2="mph"
else
    windcurrent1=$(awk "BEGIN {print $windcurrent * 3.6}")
    wind22=$(awk "BEGIN {print $wind2 * 3.6}")
    wind33=$(awk "BEGIN {print $wind3 * 3.6}")
    wind44=$(awk "BEGIN {print $wind4 * 3.6}")
    symb=$(echo -e "\u2103")
    symb2="kph"
fi
printf "\033c"
echo -e '\e[01m'""
echo -e "Location - $place, $nation:"
echo ""
echo -e "Current weather conditions:"
echo -e '\e[0m'""
echo -e "Temperature - $tempcurrent $symb"
echo -e "Atmospheric Pressure - $pressurecurrent hPa"
echo -e "Humidity - $humidcurrent %"
echo -e "Wind speed - $windcurrent1 $symb2"
echo -e "Wind direction - $direcurrent°"
echo -e "Weather condition - $condicurrent"
echo -e '\e[01m'""
echo -e "Forecast for $time22:"
echo -e '\e[0m'""
echo -e "Minimum Temperature - $min2 $symb"
echo -e "Maximum Temperature - $max2 $symb"
echo -e "Atmospheric Pressure - $pressure2 hPa"
echo -e "Humidity - $humid2 %"
echo -e "Wind speed - $wind22 $symb2"
echo -e "Wind direction - $dire2°"
echo -e "Weather condition - $condi2"
echo -e '\e[01m'""
echo -e "Forecast for $time33:"
echo -e '\e[0m'""
echo -e "Minimum Temperature - $min3 $symb"
echo -e "Maximum Temperature - $max3 $symb"
```

```
echo -e "Atmospheric Pressure - $pressure3 hPa"
echo -e "Humidity - $humid3 %"
echo -e "Wind speed - $wind33 $symb2"
echo -e "Wind direction - $dire3°"
echo -e "Weather condition - $condi3"
echo -e '\e[01m'""
echo -e "Forecast for $time44:"
echo -e '\e[0m'""
echo -e "Minimum Temperature - $min4 $symb"
echo -e "Maximum Temperature - $max4 $symb"
echo -e "Atmospheric Pressure - $pressure4 hPa"
echo -e "Humidity - $humid4 %"
echo -e "Wind speed - $wind44 $symb2"
echo -e "Wind direction - $dire4°"
echo -e "Weather condition - $condi4"
echo -e ""
"$local_rm" -f "/tmp/forecast.json"
"$local_rm" -f "/tmp/current.json"
fi
exit 0
#####
```

elejére, ahol négy üres változót találsz **local_latitude** (szélesség), **local_longitude** (hosszúság), **local_unit** (mértékegység) és **api_key**. Írd be az lakhelyed szélességi és hosszúsági fokát a változókba. Lehet, hogy nem tudod a lakhelyed szélességi és hosszúsági paramétereit. Ne aggódj! Könnyű. A Tago több mint 193 ország 2,667,417 helységének koordinátáit ismeri. Lépj be a Tago [oldalra](#) és keress rá a helyed hosszúsági és szélességi adataira.

Ezután meg kell határoznod a használni kívánt mértékegységeket a **local_unit**-ban. Két nagyobb mértékegység-rendszer van használatban, a metrikus és az angol-szász. Melyiket használd? Ha hőmérsékletet Celsiusban (°C) és a távolságot kilométerben (km) szereted, használd a metrikust. Ám ha Fahrenheitben szereted mérni a hőmérsékletet és mérföldben (mi) a távolságot, használd az Imperial-t.

Az utolsó lépés az OpenWeatherMap-ról az API-kulcs kinyerése. Nyugi! Egyszerű és az ingyenes felületük elég jó a mi számunkra. Íme, itt van a [hivatkozás](#) a OpenWeatherMap-re feliratkozáshoz.

Regisztrálás után megkapod a saját, egyedi **API-key**-t. Ezt az kulcsot add meg az **api_key** változónak a **pmwf** szkriptjében.

Ekkor a változók rész a **pmwf** szkriptjé felső részében, egy Londonban (UK) lakó számára valahogy így fog kinézni:

local_latitude=51.500
local_longitude=-0.117
local_unit=metric
api_key=your_api_key_here

És egy New York-i számára valahogy így:

local_latitude=40.714
local_longitude=-74.006
local_unit=imperial
api_key=your_api_key_here

Mentsd a **pmwf** szkriptfájlját a változók meghatározását követően. Nyiss terminált és írd be **pmwf**, majd Enter és megkapod a lakhelyed időjárását háromnapos előrejelzéssel.

Ahogy mindig is, az egyéni körülmények sok mindent meghatároznak. Noha a legtöbb embernek működik, (sok esetben) nálad nem működhet.

Kérdés? Érdeklődés? Javaslat? Hiba? Ne kímélj!



The PCLinuxOS Magazine Special Editions!

Get Your Free Copies Today!