

Tutorial 2 – Introduction to Bash Scripting

Disclaimer: Subject to updates as corrections are found

Version 0.12

Scoring: 20 pts maximum

The purpose of this tutorial is to introduce the Bash scripting under the Linux operating system. Complete this tutorial leveraging your Linux environment set up for Tutorial 1. Please review sections 1 – 8 of the online Bash Scripting Tutorial as needed to complete the scripting activity:

Bash Scripting Tutorial:

<https://ryanstutorials.net/bash-scripting-tutorial/>

Tutorial Sections include:

1. What is a BASH script?
2. Variables
3. Input
4. Arithmetic
5. If Statements
6. Loops
7. Functions
8. User Interface

At the conclusion of the online Bash tutorial, please complete the Bash scripting task described below. As needed, search the internet to find BASH examples beyond Ryan’s tutorial to help with the overall programming task. Submit your completed operational Bash script as a file called **weather.sh** online via Canvas.

BASH WEATHER FORECAST TOOL

Write a short Bash script that consumes two web services to produce a weather forecast that is local to where your computer presently receives internet service. To complete the script it is recommended to use the following commands:

Command	Description
curl	Command line http REST client for performing GET PUT POST requests, etc. If curl has not already been installed, it can be installed with: sudo apt install curl When using curl, please use the “-s” flag to request silent output without continuous status information
cut	Cut is simple parsing tool in Bash. Simply pipe text to cut, and specify a custom column delimiter with “-d”, then specify the desired column with “-f”.
jq	Jq is a Bash JSON parser. You will need to install jq as follows: sudo apt install jq

Use the following two web services to obtain a weather forecast using the geolocation (latitude and longitude coordinates) of your internet providers IP address.

IP Address Location API <https://ipinfo.io/developers>

The “IP Address Location API” provides latitude and longitude based on the IP address of your incoming web service request. The API is free to use and does not require registration. The web service never sees your internal IP address inside the “firewall” of your business, school, or home. For example, if at home you might receive an internet IP address from a Comcast modem of “10.0.0.50”. This is an internal IP address. Internal IP addresses are described in the table:

Private IPv4 addresses

RFC1918 name	IP address range	host id size
24-bit block	10.0.0.0 – 10.255.255.255	24 bits
20-bit block	172.16.0.0 – 172.31.255.255	20 bits
16-bit block	192.168.0.0 – 192.168.255.255	16 bits

The “IP Address Location API” will provide metadata regarding your internet service provider in JSON format as follows:

```
{
  "ip": "71.209.4.118",
  "hostname": "c-71-209-4-118.hsd1.wa.comcast.net",
  "city": "Tacoma",
  "region": "Washington",
  "country": "US",
  "loc": "47.2529,-122.4417",
  "postal": "98402",
  "org": "AS33650 Comcast Cable Communications, LLC"
}
```

Using the “jq” function, parse the latitude and longitude and pass this to the Weatherbit API.

Because you’re limited to 1,000 free requests per day, you should cache this information using a local hidden file called “.myipaddr”. In Bash, at the start of your script, check for the existence of a file called “.myipaddr”. If the file does not exist, call the service to obtain the JSON object. Save the JSON object to the disk. Read the JSON object into a Bash variable using the “cat” command. Then process the JSON with “jq”.

Print out the following messages based on whether you're using a cached IP from ".myipaddr" or whether your script had to call the service to obtain the JSON describing your IP and location:

```
echo "CALLING API TO QUERY MY IP"
```

```
echo "IP READ FROM CACHE"
```

See the online documentation for more information: <https://ipinfo.io/developers>

Weatherbit API <https://www.weatherbit.io/api>

Use the Weatherbit API to obtain a 16 weather forecast with High and Low temperature information. The Weatherbit API provides a detailed 16-day forecast on request. This API is also limited to 1,000 calls per day. If you are concerned that debugging your script will exceed 1,000 calls per day, please consider caching the JSON object as described above for the IP Address Location API.

You will need to register and obtain an API Key to use the Weatherbit API. Keys are generally made available within a couple of minutes. Once you have a key, you can check your daily usage quota with the following call:

```
curl -s -g https://api.weatherbit.io/v2.0/subscription/usage?key=[YOUR-API-KEY]
```

Specify your actual apikey (a combination of letters and numbers) in place of "[YOUR-API-KEY]".

To obtain the 16-day weather forecast, use the following API:
<https://www.weatherbit.io/api/weather-forecast-16-day>

***** To call the weatherbit services with CURL put your URL in "quotes" *****

The following online JSON formatter is helpful to paste JSON into a web browser to rapidly make it more readable: <https://jsonlint.com/>

Your weather.sh script should produce output as below:

Sample Output - First Call:

```
CALLING API TO QUERY MY IP
```

```
Forecast for my lat=47.6102°, lon=-122.3040°  
Forecast for 2018-10-05 HI: 12.5°C LOW: 9.6°C  
Forecast for 2018-10-06 HI: 11.4°C LOW: 8.9°C  
Forecast for 2018-10-07 HI: 12.8°C LOW: 7.3°C  
Forecast for 2018-10-08 HI: 14.2°C LOW: 9.9°C  
Forecast for 2018-10-09 HI: 16°C LOW: 10.4°C  
Forecast for 2018-10-10 HI: 16.5°C LOW: 7.7°C  
Forecast for 2018-10-11 HI: 17.6°C LOW: 3.5°C  
Forecast for 2018-10-12 HI: 19°C LOW: 8.6°C  
Forecast for 2018-10-13 HI: 17.6°C LOW: 8.2°C  
Forecast for 2018-10-14 HI: 20.7°C LOW: 9.3°C
```

Forecast for 2018-10-15 HI: 22.8°C LOW: 10.3°C
Forecast for 2018-10-16 HI: 18.9°C LOW: 12.1°C
Forecast for 2018-10-17 HI: 16.7°C LOW: 12.3°C
Forecast for 2018-10-18 HI: 14.6°C LOW: 12.7°C
Forecast for 2018-10-19 HI: 15.5°C LOW: 9°C
Forecast for 2018-10-20 HI: 7.5°C LOW: 7.5°C

Sample Output - Subsequent Calls:

IP READ FROM CACHE

Forecast for my lat=47.6102°, lon=-122.3040°
Forecast for 2018-10-04 HI: 12.2°C LOW: 12.2°C
Forecast for 2018-10-05 HI: 12.3°C LOW: 9.9°C
Forecast for 2018-10-06 HI: 11.1°C LOW: 8.2°C
Forecast for 2018-10-07 HI: 12.8°C LOW: 7.3°C
Forecast for 2018-10-08 HI: 14.2°C LOW: 9.9°C
Forecast for 2018-10-09 HI: 16°C LOW: 10.4°C
Forecast for 2018-10-10 HI: 16.5°C LOW: 7.7°C
Forecast for 2018-10-11 HI: 17.6°C LOW: 3.5°C
Forecast for 2018-10-12 HI: 19°C LOW: 8.6°C
Forecast for 2018-10-13 HI: 17.6°C LOW: 8.2°C
Forecast for 2018-10-14 HI: 20.7°C LOW: 9.3°C
Forecast for 2018-10-15 HI: 22.8°C LOW: 10.3°C
Forecast for 2018-10-16 HI: 18.9°C LOW: 12.1°C
Forecast for 2018-10-17 HI: 16.7°C LOW: 12.3°C
Forecast for 2018-10-18 HI: 14.6°C LOW: 12.7°C
Forecast for 2018-10-19 HI: 15.5°C LOW: 9°C

How to type the degree symbol in Ubuntu/BASH:

To produce the degrees symbol in Ubuntu/BASH simply type [CONTROL]-[SHIFT]-[U] as a three-key combination and then quickly type "b0" and press [ENTER]. The degree symbol should appear: °

Python

Students may choose to implement the weather script in Python. It should produce identical output and the source code should include comments describing any special instructions on how to run the script (e.g. which Python version or list of packages to install)

Scoring

The Bash weather script (weather.sh) will be scored out of 20 points.

10 points for being functionally correct, in other words, it needs to obtain a 16-day weather forecast for the lat/long of the internet service provider.

10 additional points are for proper formatting. Formatting should match the output exactly as described in the tutorial specification.