**Activity 5: Building an IP Lookup Tool**

* The first step is to create and navigate into the /03-student/day3/IP\_Lookup\_Tool folder on your VM. To do this, run the following command:
  + cd /03-student/day3/IP\_Lookup\_Tool
* Begin with running the new command to see what data it returns on an IP address:
  + curl -s http://ipinfo.io/104.223.95.86
  + This returns:
* "ip": "104.223.95.86",
* "hostname": "r-86-95-223-104.consumer-pool.prcdn.net",
* "city": "Atlanta",
* "region": "Georgia",
* "country": "US",
* "loc": "33.7490,-84.3880",
* "org": "AS8100 QuadraNet Enterprises LLC",
* "postal": "30302",
* "timezone": "America/New\_York",

"readme": "https://ipinfo.io/missingauth"

* Next, we are tasked with using grep, pipes, and awk to extract only the country from these results.
* To do this, we will first modify the script to grep the line that has the country:
  + curl -s http://ipinfo.io/104.223.95.86 | grep country
* When you run this command, it will return the following:

"country": "US",

* We need to now use awk to isolate out the value of "US" from this line.
* To do this, we will filter by : as this is what separates the title of "country" and the value:
  + curl -s http://ipinfo.io/104.223.95.86 | grep country | awk -F: '{print $2}'
  + We also print out the second value since that field contains the country.
* The above command returns the following:

"US",

* Next we will place this script into a file called IP\_lookup.sh:
  + nano IP\_lookup.sh
  + We will place the command within this script.
* We need to make the IP address an argument that is passed, so we will replace the IP with $1.
  + curl -s http://ipinfo.io/$1 | grep country | awk -F: '{print $2}'
* Save the Nano file by pressing Ctrl+x together, and y to keep the name as IP\_lookup.sh.
* Run the following three commands to confirm the script can identify the country from the IP addresses:
  + sh IP\_lookup.sh 133.18.55.255
  + sh IP\_lookup.sh 41.34.55.255
  + sh IP\_lookup.sh 187.54.23.8
* The results should show:
* - "JP", (Japan)
* - "EG", (Egypt)

- "BR", (Brazil)

* We can clean this up a little by getting rid of the extraneous quotes and comma. We do this by telling awk to use a class of characters as field separators, changing the command in our script from awk -F: '{print $2}' to awk -F'[:",]' '{print $5}'