

```
1  #!/bin/bash
2
3  # Make the render directory.
4  if [ ! -d "render" ]; then
5      mkdir render
6  fi
7
8  echo 'Creating first json layer with hospital points...'
9
10 # Create a new csv `points.csv` with a header row
11 echo 'i,lat,lng,o,d' > points.csv
12
13 # In the state column, reverse grep for territories,
14 # cut out the following columns:
15 # 1: provider_id
16 # 7: Lat
17 # 8: Lng
18 # 12: observed
19 # 13: days
20 # skip the header,
21 # shorten "Not Available",
22 # and send it all to `points.csv`.
23 csvgrep -c 'state' -r 'GU|MP|PR|VI' -i hospitals_clabsi.csv | csvcut -c 1,7,8,12,13 | tail
    -n +2 | sed -E 's/Not Available/NA/g' >> points.csv
24
25 # Let's create the first json file.
26 csvjson points.csv > render/hospitals.json
27
28 # Un-comment the two lines below if you want to create separate json files for
29 # hospitals in the data that reported central line infections and those that didn't.
30 # csvgrep -c 'o' -r 'NA' -i points.csv | csvjson > render/hospitals_info.json
31 # csvgrep -c 'o' -r 'NA' points.csv | csvjson > render/hospitals_na.json
32
33 echo 'Creating individual hospital json...'
34 # Create the `render/hospitals` folder.
35 if [ ! -d "render/hospitals" ]; then
36     mkdir render/hospitals
37 fi
38
39 # `render.py` creates over 4k json files (for each hospital)
40 python render.py
41
42 # Clean up points.csv
43 rm points.csv
44
45 echo 'Rendering files for web complete. Check `render` directory.'
```