

CPE 322 – Lab 6

1. First, I installed Node.js:

```
jstefan1@DESKTOP-6SV6DJ0:~$ sudo apt install nodejs
[sudo] password for jstefan1:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  libc-ares2 libjs-highlight.js libnode72 nodejs-doc
Suggested packages:
  npm
The following NEW packages will be installed:
  libc-ares2 libjs-highlight.js libnode72 nodejs nodejs-doc
0 upgraded, 5 newly installed, 0 to remove and 48 not upgraded.
Need to get 13.7 MB of archives.
After this operation, 53.9 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libjs-highlight.js all 9.18.5+dfsg1-1 [367 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 libc-ares2 amd64 1.18.1-1ubuntu0.22.04.2 [45.0 kB]
Get:3 http://archive.ubuntu.com/ubuntu jammy/universe amd64 libnode72 amd64 12.22.9~dfsg-1ubuntu3 [10.8 MB]
Get:4 http://archive.ubuntu.com/ubuntu jammy/universe amd64 nodejs-doc all 12.22.9~dfsg-1ubuntu3 [2409 kB]
Get:5 http://archive.ubuntu.com/ubuntu jammy/universe amd64 nodejs amd64 12.22.9~dfsg-1ubuntu3 [122 kB]
Fetched 13.7 MB in 1s (14.6 MB/s)
Selecting previously unselected package libjs-highlight.js.
(Reading database ... 49394 files and directories currently installed.)
Preparing to unpack .../libjs-highlight.js_9.18.5+dfsg1-1_all.deb ...
Unpacking libjs-highlight.js (9.18.5+dfsg1-1) ...
Selecting previously unselected package libc-ares2:amd64.
Preparing to unpack .../libc-ares2_1.18.1-1ubuntu0.22.04.2_amd64.deb ...
Unpacking libc-ares2:amd64 (1.18.1-1ubuntu0.22.04.2) ...
Selecting previously unselected package libnode72:amd64.
Preparing to unpack .../libnode72_12.22.9~dfsg-1ubuntu3_amd64.deb ...
```

- a.
2. I then ran the following commands and files in the linux terminal (Ubuntu):

```

jstefan1@DESKTOP-6SV6DJ0:~$ node -v
v12.22.9
jstefan1@DESKTOP-6SV6DJ0:~$ npm -v
10.1.0
jstefan1@DESKTOP-6SV6DJ0:~$ node -h
Usage: node [options] [ script.js ] [arguments]
       node inspect [options] [ script.js | host:port ] [arguments]

Options:
  -                  script read from stdin (default if no file name is
                    provided, interactive mode if a tty)
  --                indicate the end of node options
  --abort-on-uncaught-exception
                    aborting instead of exiting causes a core file to
                    be generated for analysis
  -c, --check        syntax check script without executing
  --completion-bash   print source-able bash completion script
  --conditions=...   additional user conditions for conditional exports
                    and imports
  --cpu-prof          Start the V8 CPU profiler on start up, and write
                    the CPU profile to disk before exit. If
                    --cpu-prof-dir is not specified, write the profile
                    to the current working directory.
  --cpu-prof-dir=... Directory where the V8 profiles generated by
                    --cpu-prof will be placed. Does not affect --prof.
  --cpu-prof-interval=... specified sampling interval in microseconds for the
                    V8 CPU profile generated with --cpu-prof. (default:
                    1000)
  --cpu-prof-name=... specified file name of the V8 CPU profile generated
                    with --cpu-prof
  --diagnostic-dir=... set dir for all output files (default: current

```

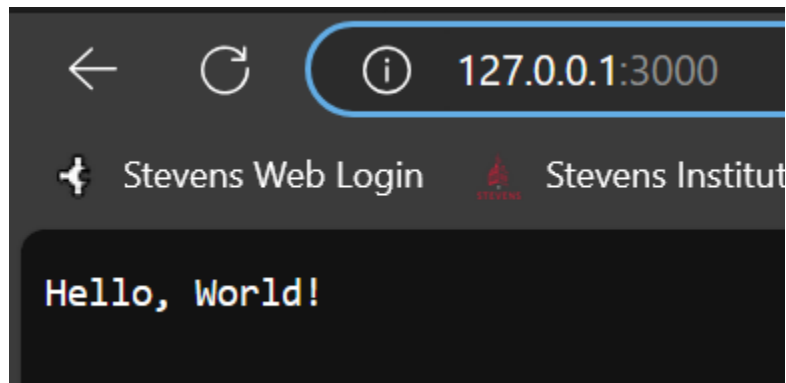
a.

```

jstefan1@DESKTOP-6SV6DJ0:~/iot/lesson6$ node hello-world.js
Server running at http://127.0.0.1:3000/

```

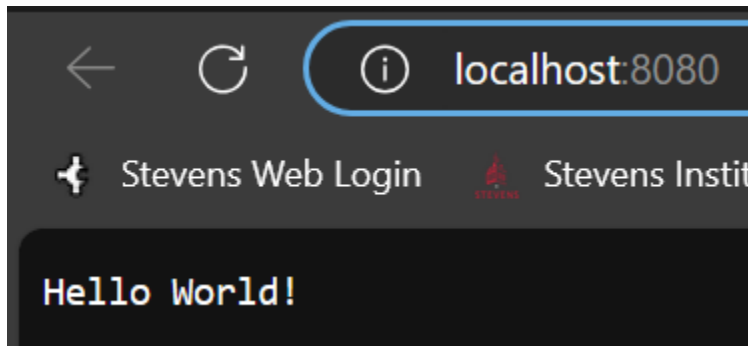
b.



c.

```
jstefan1@DESKTOP-6SV6DJ0:~/iot/lesson6$ node hello.js
Server running at http://127.0.0.1:8080/
response end call done
request end event fired
response end call done
request end event fired
```

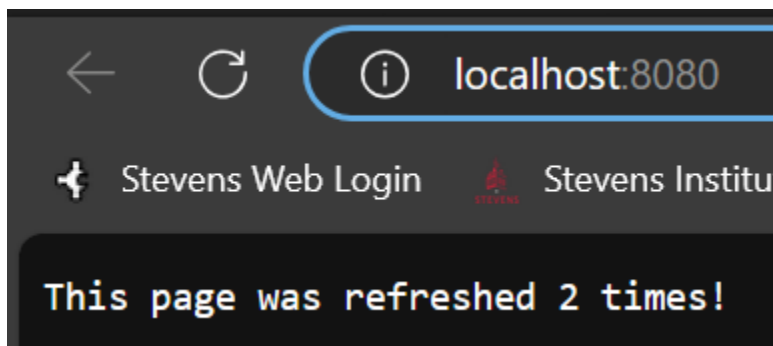
d.



e.

```
jstefan1@DESKTOP-6SV6DJ0:~/iot/lesson6$ node http.js
0
1
```

f.



g.

3. Installing Pystache:

```
jstefan1@DESKTOP-6SV6DJ0:~$ sudo pip3 install pystache
[sudo] password for jstefan1:
Collecting pystache
  Downloading pystache-0.6.5-py3-none-any.whl (81 kB)
    81.3/81.3 KB 2.8 MB/s eta 0:00:00
Installing collected packages: pystache
Successfully installed pystache-0.6.5
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system packa
ge manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
```

a.

4. Then I ran the following commands and files:

```
jstefan1@DESKTOP-6SV6DJ0:~$ cd ~/iot/lesson6
jstefan1@DESKTOP-6SV6DJ0:~/iot/lesson6$ cat say_hello.mustache
Hello, {{to}}!
```

a.

```
jstefan1@DESKTOP-6SV6DJ0:~/iot/lesson6$ cat say_hello.py
# https://github.com/defunkt/pystache
import pystache
print(pystache.render('Hi {{person}}!', {'person': 'Alexa'}))

# Create dedicated view classes to hold view logic
class SayHello(object):
    def to(self):
        return "World"
hello = SayHello()

# Use template in say_hello.mustache
renderer = pystache.Renderer()
print(renderer.render(hello))

# Pre-parse a template
parsed = pystache.parse('Hey {{#who}}{{.}}!{{/who}}')
print(parsed)
print(renderer.render(parsed, {'who': 'Google'}))
print(renderer.render(parsed, {'who': 'Siri'}))
```

b.

```
jstefan1@DESKTOP-6SV6DJ0:~/iot/lesson6$ python3 say_hello.py
Hi Alexa!
Hello, World!

['Hey ', _SectionNode(key='who', index_begin=12, index_end=18, parsed=[_EscapeNode(key='.'), '!'])]
Hey Google!
Hey Siri!
```

c.