gabriel@gabriel-Lenovo-920-13IKB: ~

File Edit View Search Terminal Help

```
gabriel@gabriel-Lenovo-920-13IKB:~$ node -v
v8.10.0
gabriel@gabriel-Lenovo-920-13IKB:~$ npm -v
3.5.2
gabriel@gabriel-Lenovo-920-13IKB:~$ node -h
Usage: node [options] [ -e script | script.js | - ] [arguments]
       node inspect script.js [arguments]

Options:
  -v, --version              print Node.js version
  -e, --eval script          evaluate script
  -p, --print                evaluate script and print result
  -c, --check                syntax check script without executing
  -i, --interactive          always enter the REPL even if stdin
                             does not appear to be a terminal
  -r, --require              module to preload (option can be repeated)
  -                          script read from stdin (default; interactive mode if a tty)
  --inspect[=[host:]port]    activate inspector on host:port
                             (default: 127.0.0.1:9229)
  --inspect-brk[=[host:]port]
                             activate inspector on host:port
                             and break at start of user script
  --inspect-port=[host:]port
                             set host:port for inspector
  --no-deprecation           silence deprecation warnings
  --trace-deprecation        show stack traces on deprecations
  --throw-deprecation        throw an exception on deprecations
  --pending-deprecation      emit pending deprecation warnings
  --no-warnings              silence all process warnings
  --napi-modules             load N-API modules (no-op - option
                             kept for compatibility)
  --abort-on-uncaught-exception
                             aborting instead of exiting causes a
                             core file to be generated for analysis
  --trace-warnings           show stack traces on process warnings
  --redirect-warnings=file
                             write warnings to file instead of
                             stderr
  --trace-sync-io            show stack trace when use of sync IO
                             is detected after the first tick
  --force-async-hooks-checks
                             enables checks for async_hooks
  --trace-events-enabled     track trace events
  --trace-event-categories   comma separated list of trace event
                             categories to record
  --track-heap-objects       track heap object allocations for heap snapshots
  --prof-process             process v8 profiler output generated
                             using --prof
  --zero-fill-buffers        automatically zero-fill all newly allocated
                             Buffer and SlowBuffer instances
  --v8-options               print v8 command line options
```

Hello World!

gabriel@gabriel-Lenovo-920-13IKB: ~/iot/lesson6

File Edit View Search Terminal Help

```
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ node hello-world.js
Server running at http://127.0.0.1:3000/
response end call done
response end event fired
^C
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ node http.js
0

1
2
3
4
^C
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ []
```

This page was refreshed 5 times!

gabriel@gabriel-Lenovo-920-13IKB: ~/iot/lesson6

File   Edit   View   Search   Terminal   Help

```
^C
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ node http.js
0

1
2
3
4
^C
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ sudo pip3 install pystache
The directory '/home/gabriel/.cache/pip/http' or its parent directory is not owned by the current user and the cache has been disabled. Please check the permissions and owner of that directory. If executing pip
with sudo, you may want sudo's -H flag.
The directory '/home/gabriel/.cache/pip' or its parent directory is not owned by the current user and caching wheels has been disabled. check the permissions and owner of that directory. If executing pip with su
do, you may want sudo's -H flag.
Collecting pystache
  Downloading https://files.pythonhosted.org/packages/3f/e7/8750ba6c6101d6aa5ceeb20c013adf2c6f3554a12c71d75654b468404bfa/pystache-0.6.0.tar.gz (78kB)
    100% |████████████████████████████████| 81kB 6.8MB/s
Installing collected packages: pystache
  Running setup.py install for pystache ... done
Successfully installed pystache-0.6.0
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ cd ~/iot/lesson6
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ cat say_hello.mustache
Hello, {{to}}!
gabriel@gabriel-Lenovo-920-13IKB:~/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'}))
gabriel@gabriel-Lenovo-920-13IKB:~/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!
gabriel@gabriel-Lenovo-920-13IKB:~/iot/lesson6$ []
```