



Travis-CI

Getting Started

- Simply follow this guide <https://docs.travis-ci.com/user/tutorial/>

The Dreaded YAML file

- each new line is kinda like a semi colon in java
- it uses spaces to tell that code is assigned in a block think spaces here basically act like curly brackets in java
- I low key hate them

Heres our current yaml file

```
language: node_js node_js: - "stable" before_install: - stty cols 80 env:
CC_TEST_REPORTER_ID=56b4d0bd7e39c085bacccf8b63492ace01b88f1032c1853737145f
dist: trusty sudo: required addons: firefox: latest apt: sources: - google
packages: - google-chrome-stable fluxbox before_script: - curl -L
https://codeclimate.com/downloads/test-reporter/test-reporter-latest-linux
./cc-test-reporter - chmod +x ./cc-test-reporter - ./cc-test-reporter befo
- "export DISPLAY=:99.0" - "sh -e /etc/init.d/xvfb start" - sleep 3 - flux
>/dev/null 2>&1 & cache: directories: - "node_modules" script: - bundle ex
- npm run lint - npm run test-mocha - npm run browser-test after_script: -
test-reporter after-build --exit-code $TRAVIS_TEST_RESULT after_success: g
```

JavaScript ▾

what it basically does is it tells travis how we want our environment set up and what then we have it run some commands.

How da heck did I write the above

- I copy and pasted from travis-ci and then I looked at other example of yaml files that were doing what I needed aka GOOOGLEEEE!!!!
 - yeah I did not write this without references lol

```
script: - bundle exec rspec - npm run lint - npm run test-mocha - npm
run browser-test
```

JavaScript ▾

This bit is where we tell travis which tests we want it to run

```
- "export DISPLAY=:99.0" - "sh -e /etc/init.d/xvfb start" - sleep 3 -
fluxbox >/dev/null 2>&1 &
```

JavaScript ▾

This bit sets up the virtual display for our headless server

```
addons: firefox: latest apt: sources: - google-chrome packages: -
google-chrome-stable fluxbox
```

JavaScript ▾

This bit installs the browsers

```
env: global: -
CC_TEST_REPORTER_ID=56b4d0bd7e39c085bacccf8b63492ace01b88f1032c1853737145f
```

JavaScript ▾

This bit right here basically sets up some environment variables that code Climate uses to push a test report to their servers the Id is our code climate Repo ID

```
language: node_js node_js: - "stable"
```

JavaScript ▾

This bit just tells travis what language we are using like for instance you could replace node_js with ruby if you would like I believe it also sets things up for package management like using npm but travis may also install nvm by default which handles npm as well I'm not sure on this one. anyways heres a link to using differnt languages

- <https://docs.travis-ci.com/user/languages/>

Updated YAML File Also known as Attempting to Connect Test Coverage from Code Climate with Travis-CI

I apologize I should of been making this doc while I was making the changes but I thought it was going to be super fast

```
env: global: -
CC_TEST_REPORTER_ID=56b4d0bd7e39c085bacccf8b63492ace01b88f1032c1853737145f
matrix: include: - language: node_js node_js: - "stable" before_install: -
cols 80 dist: trusty sudo: required addons: code_climate: repo_token:
56b4d0bd7e39c085bacccf8b63492ace01b88f1032c1853737145fc658290c27 firefox:
apt: sources: - google-chrome packages: - google-chrome-stable fluxbox
before_script: - npm install codeclimate-test-reporter - "export DISPLAY=:
"sh -e /etc/init.d/xvfb start" - sleep 3 - fluxbox >/dev/null 2>&1 & after
- codeclimate-test-reporter < ./coverage/lcov.info cache: directories: -
"node_modules" script: - npm run lint - istanbul cover npm run test-mocha
istanbul cover npm run browser-test # - language: ruby # rvm: # - 2.2.0 #
install: # - bundle install # before_script: # - curl -L
https://codeclimate.com/downloads/test-reporter/test-reporter-latest-linux
./cc-test-reporter # - chmod +x ./cc-test-reporter # - ./cc-test-reporter
build # script: # - bundle exec rspec # after_script: # - ./cc-test-report
build --exit-code $TRAVIS_TEST_RESULT
```

JavaScript ▾

So the YAML file went through an update because I needed to get codeClimates test coverage to work and it was being a real pain so...

1. I attempted to run it without using ruby at which point it yelled at me that i didn't have bundle Gems and I went wtf is a bundle Gem

- So I googled and googled and tried various things only to come to the conclusion I need to run stuff in ruby as all of code climates examples were in ruby note: code climates examples do not work

2. BUILD A MATRIX!!!!

```
matrix: include:
```

JavaScript ▾

- basically you include these two lines and then you add a - to language and until it hits another language that will be one build so in the above it runs a node.js build and a ruby build.
- alas this was also the same GEM error, so i did some more googling and added the above

```
install: - bundle install
```

JavaScript ▾

- still no gooo

3. OMG THERES AN ADD ON!!!!

```
addons: code_climate: repo_token:  
56b4d0bd7e39c085bacccf8b63492ace01b88f1032c1853737145fc658290c27
```

JavaScript ▾

- now you may be thinking you dumbass why didn't you use that from the beginning well let me tell you its not in the tutorial on code climate and i only found it cause I had gone through so many google pages and even still I had doubts if it was legit
- alas this did not work either at this point I was quite bamboozled

4. Istanbul wtf

```
script: - npm run lint - istanbul cover npm run test-mocha -  
istanbul cover npm run browser-test
```

JavaScript ▾

- so you need some kinda thing to turn the test results into some kinda file called like a lcov okie sure enough this Istanbul doesn't sound sketchy at all but whatever at this point ill try anything.
- you would think at this point id found success but you would be wrong, so now it yells that theres no test coverage to be created and apparently if you've taken CSE120 its a forking issue for those of you who haven't what this basically is is that a process has a baby and that baby grows up to be a dick.

5. Currently Looking into how to fix this forking issue

QUESTIONS!!!!!!