

PORQUE VOCÊ DEVE  
APRENDER **VIM** HOJE.

**PEDROFRANCESCHI**

pedrohfranceschi@gmail.com

pedrofranceschi.com

@pedroh96

# IDE

**x**

# Editores de texto

**“Write programs that do **one** thing and  
do it well.”**

**- Filosofia UNIX**

# Background...

# Background...

**TextMate** (3 anos)





# Background...



# Background...

**Emacs** (~6 meses)

**E  
M  
A  
C  
S**

**E**scape

**M**

**A**

**C**

**S**

**E**scape

**M**eta

**A**

**C**

**S**

**E**scape

**M**eta

**A**lt

**C**

**S**

**E**scape

**M**eta

**A**lt

**C**ontrol

**S**

**E**scape

**M**eta

**A**lt

**C**ontrol

**S**hift

# Emacs x Vim

## Editor war!!



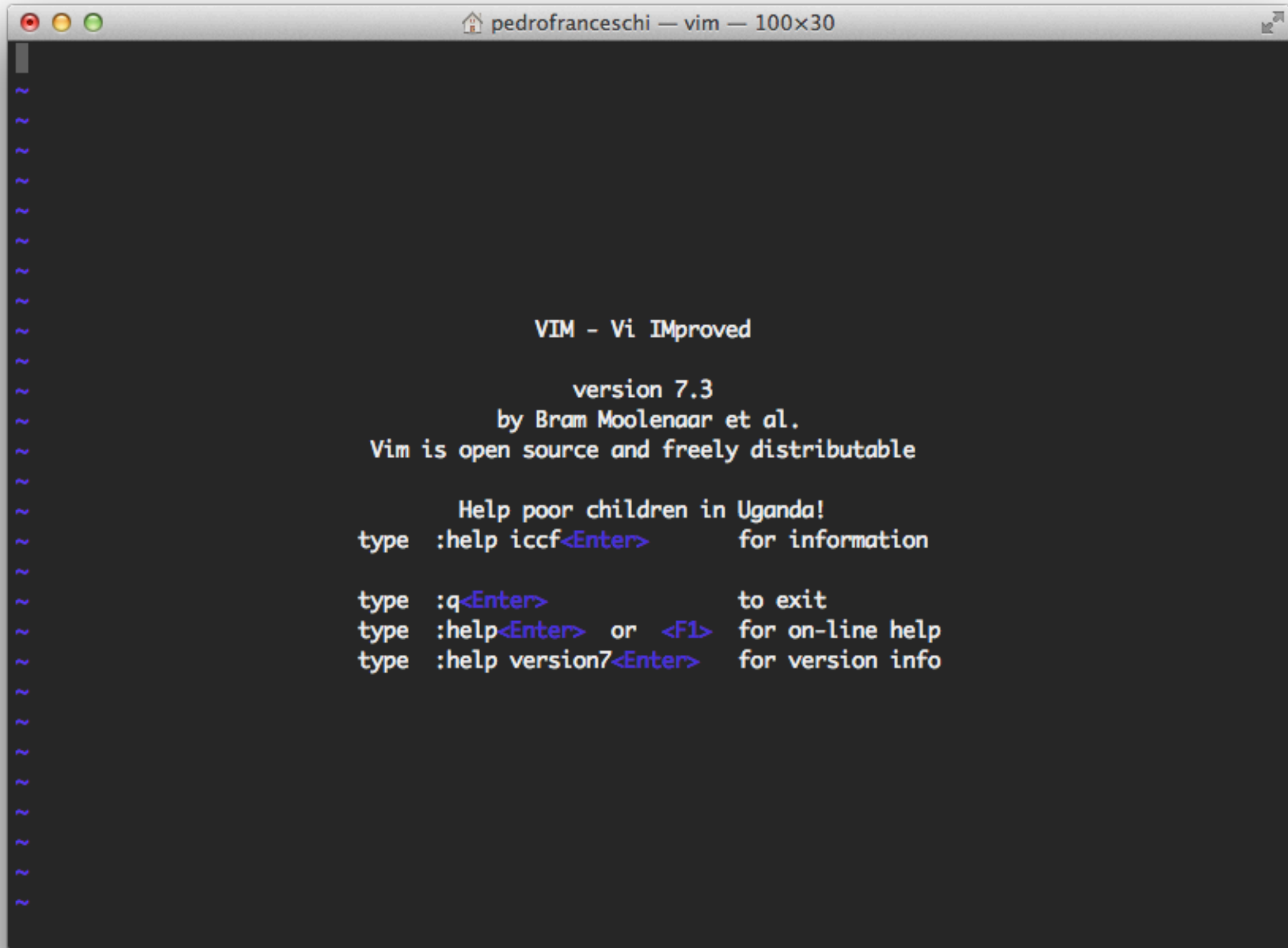
***“EMACS Makes Any Computer Slow.”***

**“Emacs é um ótimo sistema operacional.  
Só falta um editor de texto decente.”**



**Dando uma chance ao Vim...**





```
pedrofranceschi — vim — 100x30

VIM - Vi IMproved

version 7.3
by Bram Moolenaar et al.
Vim is open source and freely distributable

Help poor children in Uganda!
type :help iccf<Enter> for information

type :q<Enter> to exit
type :help<Enter> or <F1> for on-line help
type :help version7<Enter> for version info
```

**Vim = Vi(Improved)**

**Vim = Vi(Improved)**

# Vim = Vi(Improved)

- Vim philosophy



# Vim = Vi(Improved)

- Vim philosophy
- Download e uso

# Vim = Vi(Improved)

- Vim philosophy
- Download e uso
- .vimrc

# Vim = Vi(Improved)

- Vim philosophy
- Download e uso
- .vimrc
- Colorschemes (temas) e plugins

# Vim = Vi(Improved)

- Vim philosophy
- Download e uso
- .vimrc
- Colorschemes (temas) e plugins
- Demonstração

**Vim**

**philosophy**

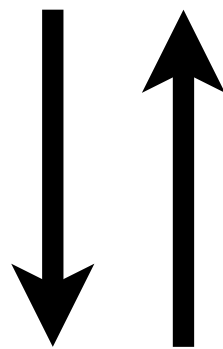
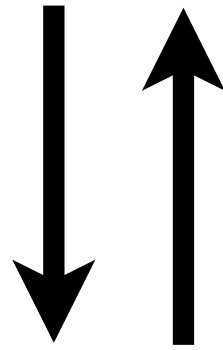
**Esqueça** tudo que você sabe sobre  
qualquer outro editor de texto.

**Vim = eficiência**



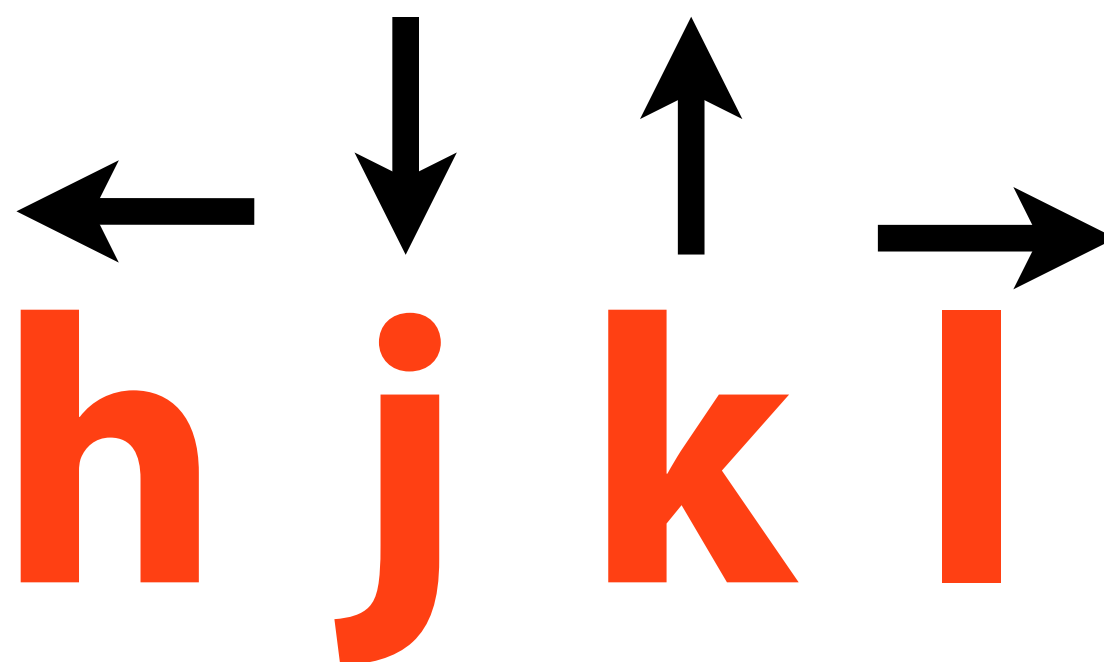


# Normal mode



# Insert mode

h j k l



A diagram illustrating four letters: 'h', 'j', 'k', and 'l'. Each letter is rendered in a bold, orange, sans-serif font. Above each letter is a black arrow indicating a direction: a left-pointing arrow above 'h', a downward-pointing arrow above 'j', an upward-pointing arrow above 'k', and a right-pointing arrow above 'l'.

**2 keystrokes para ir para  
qualquer lugar do arquivo.**

**2 keystrokes para ir para  
qualquer lugar do arquivo.**

**(sério)**

# operator

**change, delete, yank**  
**(copy), visual, etc**

+

# movement

**word, back word,**  
**end of word, find,**  
**'till, etc**

+

# argument

**(apenas quando o**  
**movement exige)**

# Exemplos

# Exemplos

**w** - avança uma palavra

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras



# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

**dw** - deleta uma palavra

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

**dw** - deleta uma palavra

**dW** - deleta até o próximo espaço

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

**dw** - deleta uma palavra

**dW** - deleta até o próximo espaço

**ci)** - deleta dentro dos parênteses e vai para o insert mode.

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

**dw** - deleta uma palavra

**dW** - deleta até o próximo espaço

**ci)** - deleta dentro dos parênteses e vai para o insert mode.

**da)** - deleta os parênteses e tudo nele

# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

**dw** - deleta uma palavra

**dW** - deleta até o próximo espaço

**ci)** - deleta dentro dos parênteses e vai para o insert mode.

**da)** - deleta os parênteses e tudo nele

**f@** - vai para a próxima ocorrência de "@"



# Exemplos

**w** - avança uma palavra

**3w** - avança 3 palavras

**W** - avança até o próximo espaço

**b** - volta uma palavra

**B** - volta até o último espaço

**dw** - deleta uma palavra

**dW** - deleta até o próximo espaço

**ci)** - deleta dentro dos parênteses e vai para o insert mode.

**da)** - deleta os parênteses e tudo nele

**f@** - vai para a próxima ocorrência de "@"

**df@** - deleta tudo até a próxima ocorrência de "@"

# Exemplos

# Exemplos

**yw** - copia a palavra atual para o clipboard

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi''** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca



# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

**\$** - vai para o final da linha

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

**\$** - vai para o final da linha

**Control + u** - sobe metade da página

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

**\$** - vai para o final da linha

**Control + u** - sobe metade da página

**Control + d** - desce metade da página

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

**\$** - vai para o final da linha

**Control + u** - sobe metade da página

**Control + d** - desce metade da página

**17G** - vai para a linha 17 do arquivo

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

**\$** - vai para o final da linha

**Control + u** - sobe metade da página

**Control + d** - desce metade da página

**17G** - vai para a linha 17 do arquivo

**gg** - vai para o início do arquivo

# Exemplos

**yw** - copia a palavra atual para o clipboard

**yi"** - copia tudo dentro das áspas para o clipboard

**p** - cola o que está no clipboard

**/var** - busca por "var" no arquivo

**n** - vai para a próxima ocorrência da busca

**N** - vai para a última ocorrência da busca

**0** - vai para o início da linha

**\$** - vai para o final da linha

**Control + u** - sobe metade da página

**Control + d** - desce metade da página

**17G** - vai para a linha 17 do arquivo

**gg** - vai para o início do arquivo

**G** - vai para o final do arquivo

**Download**

**e**

**instalação**



**O vim de `console` já vem instalado na  
grande maioria dos Linux/UNIX...**

**O vim de **console** já vem instalado na  
grande maioria dos Linux/UNIX...**

**... mas existem versões aprimoradas  
para **ambientes gráficos****

**MacVim:** versão otimizada para rodar  
no Mac OS X (com suporte a janelas,  
mouse, abas, etc)

<http://code.google.com/p/macvim/>

**gVim:** versão oficial do Vim com  
suporte a janelas para Windows e  
Linux (Ubuntu, etc)

<http://www.vim.org/download.php>

**.vimrc**

**O `.vimrc` é a mágica por trás da  
customização e da otimização do  
uso do vim.**

**.vimrc** (vimfiles)

# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.

# **.vimrc (vimfiles)**

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.



# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.
- Definição do **colorscheme** (tema)

# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.
- Definição do **colorscheme** (tema)
- Lista (Vundle) e configuração de **plugins**

# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.
- Definição do **colorscheme** (tema)
- Lista (Vundle) e configuração de **plugins**
- Fica localizado em **~/.vimrc**

# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.
- Definição do **colorscheme** (tema)
- Lista (Vundle) e configuração de **plugins**
- Fica localizado em **~/.vimrc**
- Versões do Vim com suporte a modo gráfico (MacVim, gVim, etc) usam também o **~/.gvimrc**

# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.
- Definição do **colorscheme** (tema)
- Lista (Vundle) e configuração de **plugins**
- Fica localizado em **~/.vimrc**
- Versões do Vim com suporte a modo gráfico (MacVim, gVim, etc) usam também o **~/.gvimrc**
- A comunidade **compartilha** seus vimfiles no Github.

# **.vimrc** (vimfiles)

- É o arquivo de **configuração** do Vim.
- Definição de preferências, atalhos, macros, etc.
- Definição do **colorscheme** (tema)
- Lista (Vundle) e configuração de **plugins**
- Fica localizado em **~/.vimrc**
- Versões do Vim com suporte a modo gráfico (MacVim, gVim, etc) usam também o **~/.gvimrc**
- A comunidade **compartilha** seus vimfiles no Github.
- **DICA:** comece o seu .vimrc do zero, não o copie, apenas inspire-se no dos outros usuários

# **Meus vimfiles**

**<https://github.com/pedrofranceschi/vimfiles>**

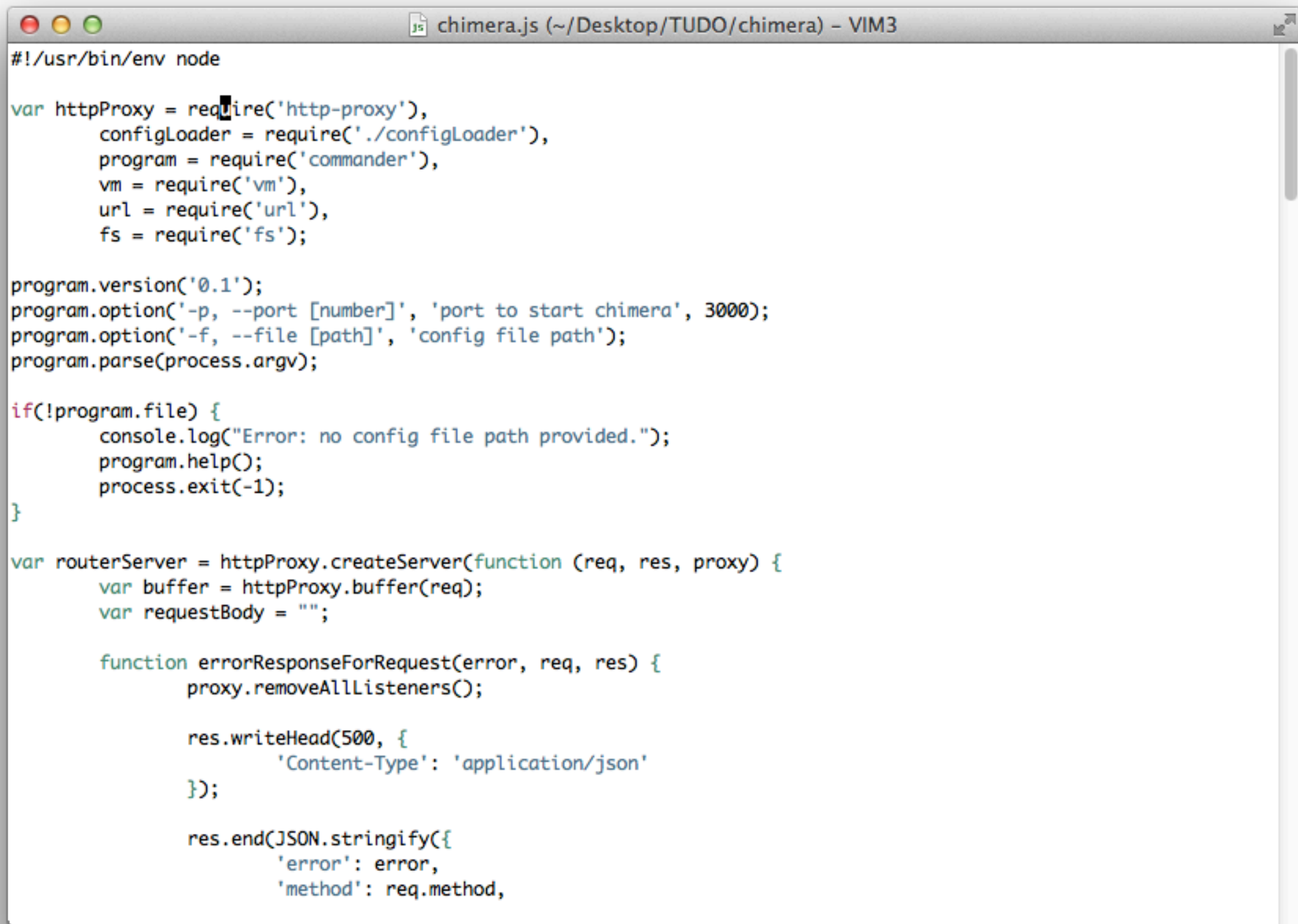
**Colorschemes**

**e**

**plugins**



**Um bom `colorscheme` torna o uso  
do Vim agradável aos olhos.**



```
#!/usr/bin/env node

var httpProxy = require('http-proxy'),
    configLoader = require('./configLoader'),
    program = require('commander'),
    vm = require('vm'),
    url = require('url'),
    fs = require('fs');

program.version('0.1');
program.option('-p, --port [number]', 'port to start chimera', 3000);
program.option('-f, --file [path]', 'config file path');
program.parse(process.argv);

if(!program.file) {
    console.log("Error: no config file path provided.");
    program.help();
    process.exit(-1);
}

var routerServer = httpProxy.createServer(function (req, res, proxy) {
    var buffer = httpProxy.buffer(req);
    var requestBody = "";

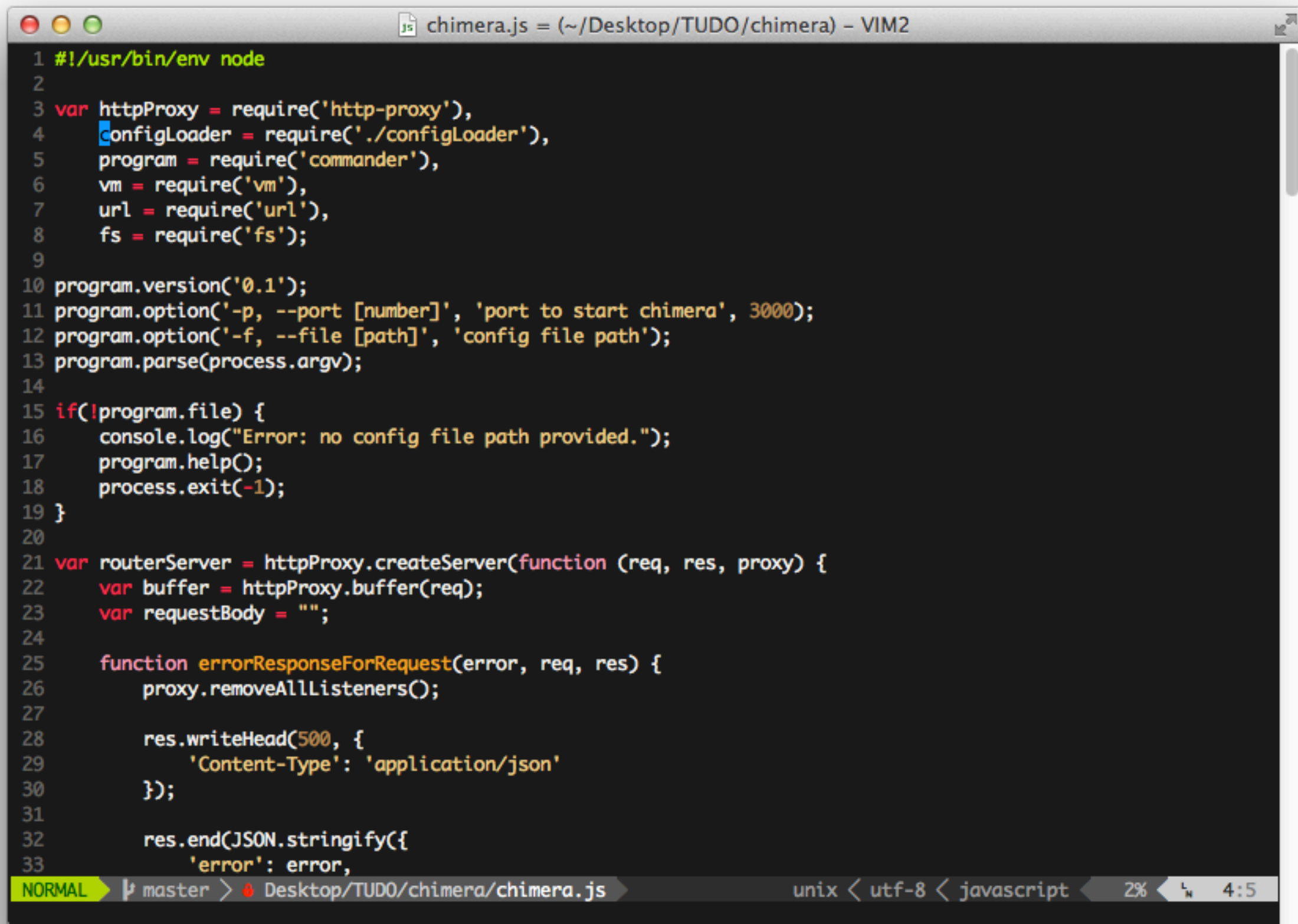
    function errorResponseForRequest(error, req, res) {
        proxy.removeAllListeners();

        res.writeHead(500, {
            'Content-Type': 'application/json'
        });

        res.end(JSON.stringify({
            'error': error,
            'method': req.method,

```

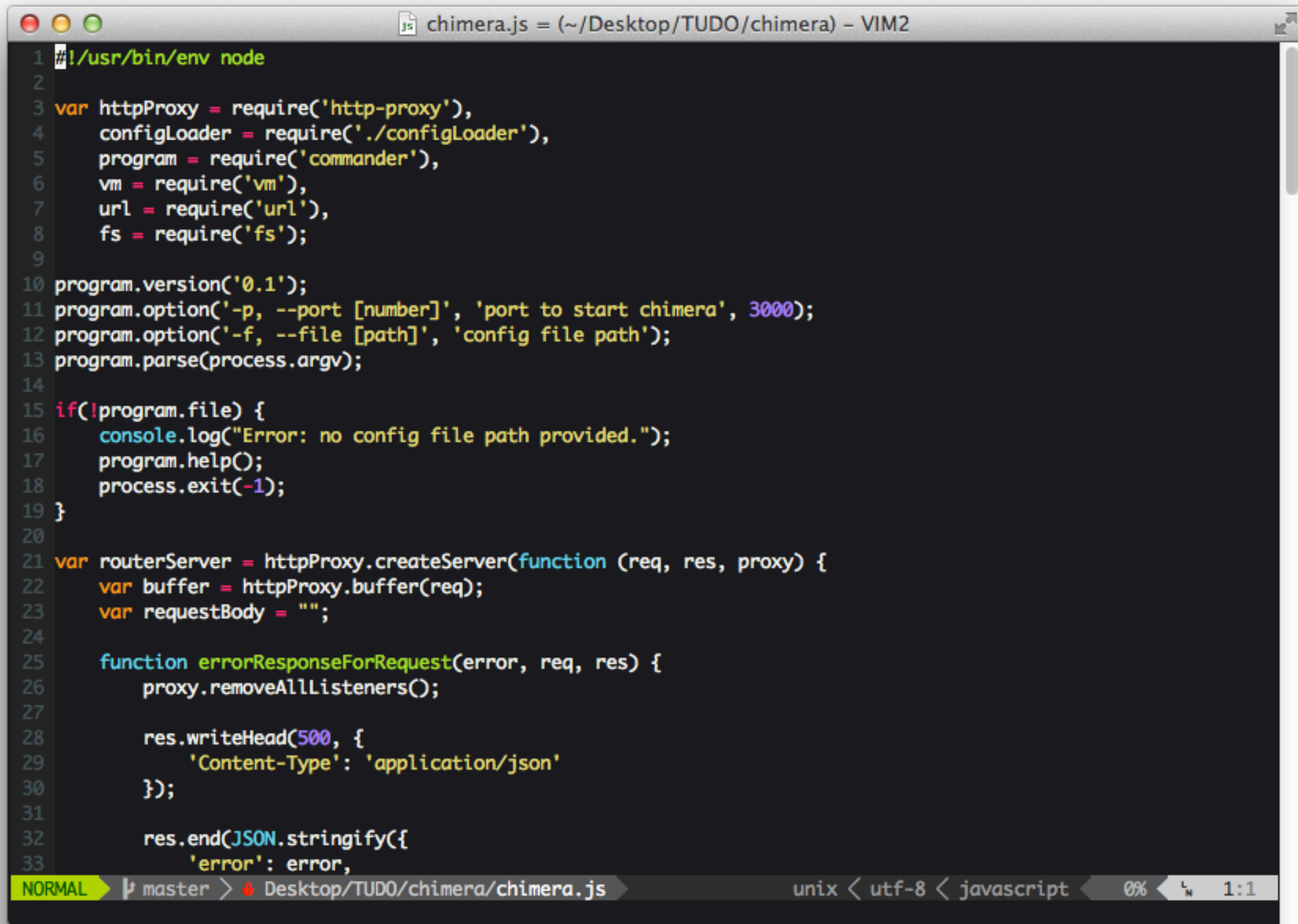
**Vim padrão (sem colorscheme)**



```
1 #!/usr/bin/env node
2
3 var httpProxy = require('http-proxy'),
4     configLoader = require('./configLoader'),
5     program = require('commander'),
6     vm = require('vm'),
7     url = require('url'),
8     fs = require('fs');
9
10 program.version('0.1');
11 program.option('-p, --port [number]', 'port to start chimera', 3000);
12 program.option('-f, --file [path]', 'config file path');
13 program.parse(process.argv);
14
15 if(!program.file) {
16     console.log("Error: no config file path provided.");
17     program.help();
18     process.exit(-1);
19 }
20
21 var routerServer = httpProxy.createServer(function (req, res, proxy) {
22     var buffer = httpProxy.buffer(req);
23     var requestBody = "";
24
25     function errorResponseForRequest(error, req, res) {
26         proxy.removeAllListeners();
27
28         res.writeHead(500, {
29             'Content-Type': 'application/json'
30         });
31
32         res.end(JSON.stringify({
33             'error': error,
```

NORMAL | master > Desktop/TUDO/chimera/chimera.js | unix < utf-8 < javascript | 2% | 4:5

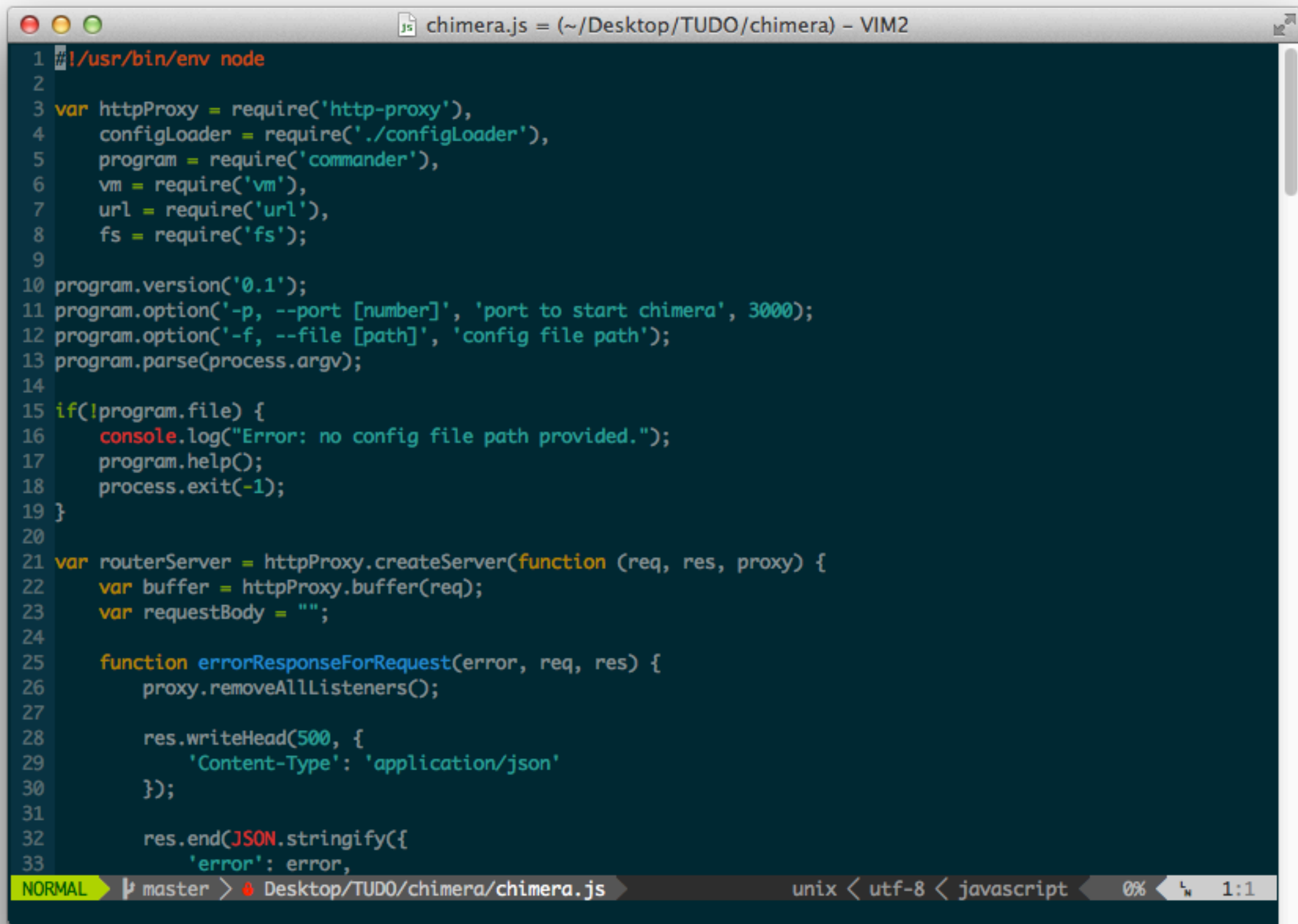
Colorscheme **badwolf**



```
1 #!/usr/bin/env node
2
3 var httpProxy = require('http-proxy'),
4     configLoader = require('./configLoader'),
5     program = require('commander'),
6     vm = require('vm'),
7     url = require('url'),
8     fs = require('fs');
9
10 program.version('0.1');
11 program.option('-p, --port [number]', 'port to start chimera', 3000);
12 program.option('-f, --file [path]', 'config file path');
13 program.parse(process.argv);
14
15 if(!program.file) {
16     console.log("Error: no config file path provided.");
17     program.help();
18     process.exit(-1);
19 }
20
21 var routerServer = httpProxy.createServer(function (req, res, proxy) {
22     var buffer = httpProxy.buffer(req);
23     var requestBody = "";
24
25     function errorResponseForRequest(error, req, res) {
26         proxy.removeAllListeners();
27
28         res.writeHead(500, {
29             'Content-Type': 'application/json'
30         });
31
32         res.end(JSON.stringify({
33             'error': error,
```

NORMAL | master > Desktop/TUDO/chimera/chimera.js | unix < utf-8 < javascript | 0% | 1:1

Colorscheme **molokai**



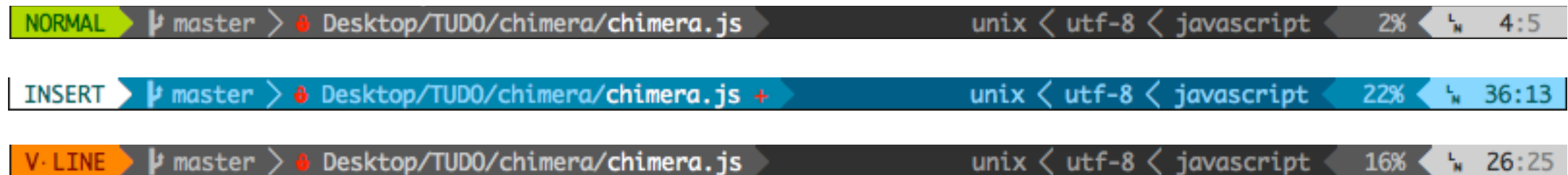
```
1 #!/usr/bin/env node
2
3 var httpProxy = require('http-proxy'),
4     configLoader = require('./configLoader'),
5     program = require('commander'),
6     vm = require('vm'),
7     url = require('url'),
8     fs = require('fs');
9
10 program.version('0.1');
11 program.option('-p, --port [number]', 'port to start chimera', 3000);
12 program.option('-f, --file [path]', 'config file path');
13 program.parse(process.argv);
14
15 if(!program.file) {
16     console.log("Error: no config file path provided.");
17     program.help();
18     process.exit(-1);
19 }
20
21 var routerServer = httpProxy.createServer(function (req, res, proxy) {
22     var buffer = httpProxy.buffer(req);
23     var requestBody = "";
24
25     function errorResponseForRequest(error, req, res) {
26         proxy.removeAllListeners();
27
28         res.writeHead(500, {
29             'Content-Type': 'application/json'
30         });
31
32         res.end(JSON.stringify({
33             'error': error,
```

NORMAL | master > Desktop/TUDO/chimera/chimera.js | unix < utf-8 < javascript | 0% | 1:1

Colorscheme **solarized**

# DICA: use o Powerline.

<https://github.com/Lokaltog/powerline>



**Plugins** permitem você fazer  
literalmente o que quiser com o Vim.

# Instalação de plugins

**Pathogen** (coloque cada plugin em uma pasta e eles serão instalados)



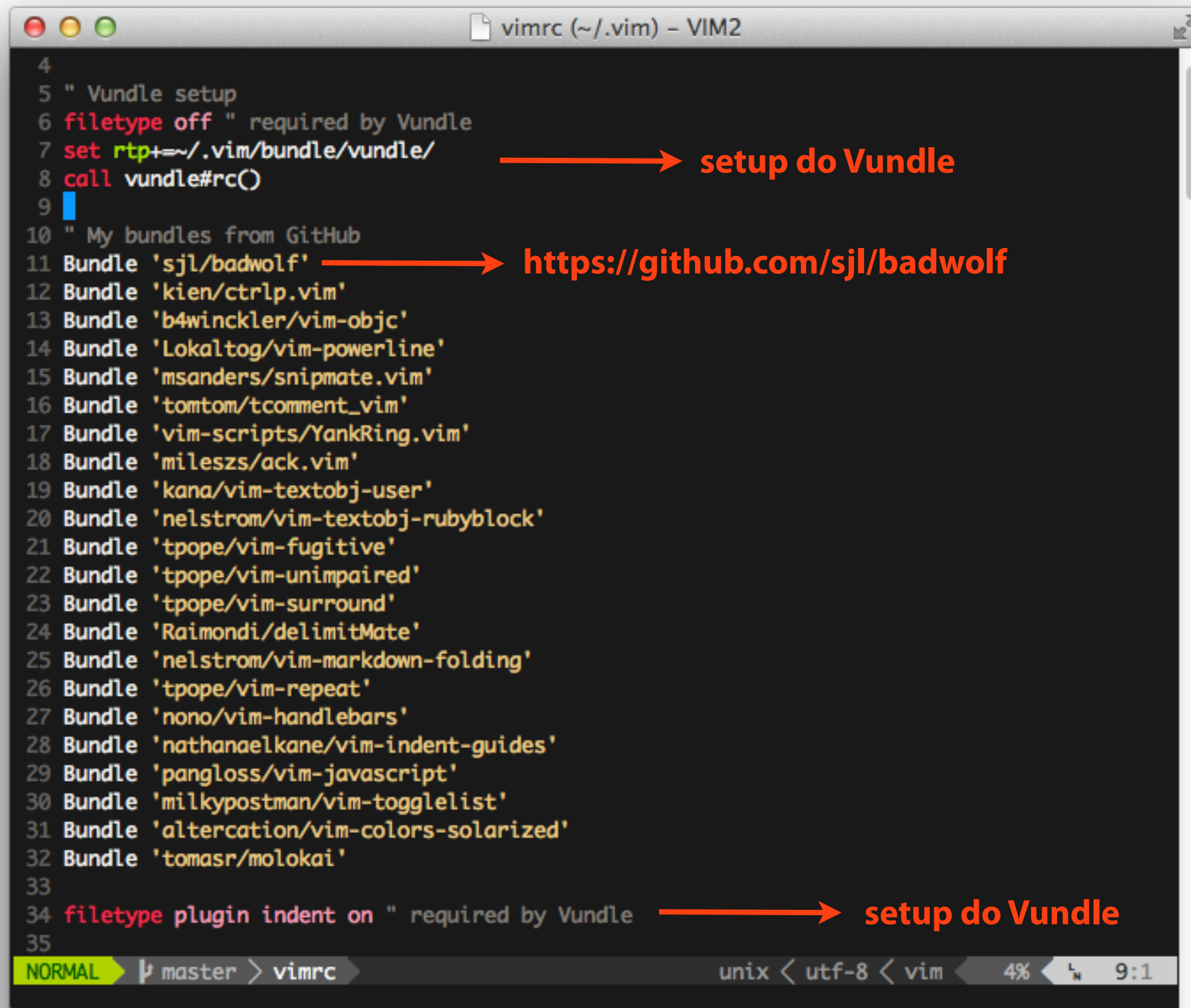
# Instalação de plugins

**Pathogen** (coloque cada plugin em uma pasta e eles serão instalados)

... OU ...

**Vundle** (defina no seu **.vimrc** a lista de plugins e eles serão instalados do Github)

# Vundle no .vimrc



The image shows a Vim editor window titled "vimrc (~/.vim) - VIM2". The editor displays a .vimrc file with the following content:

```
4
5 " Vundle setup
6 filetype off " required by Vundle
7 set rtp+=~/.vim/bundle/vundle/
8 call vundle#rc()
9
10 " My bundles from GitHub
11 Bundle 'sjl/badwolf'
12 Bundle 'kien/ctrlp.vim'
13 Bundle 'b4winckler/vim-objc'
14 Bundle 'Lokaltog/vim-powerline'
15 Bundle 'msanders/snipmate.vim'
16 Bundle 'tombtom/tcomment_vim'
17 Bundle 'vim-scripts/YankRing.vim'
18 Bundle 'mileszs/ack.vim'
19 Bundle 'kana/vim-textobj-user'
20 Bundle 'nelstrom/vim-textobj-rubyblock'
21 Bundle 'tpope/vim-fugitive'
22 Bundle 'tpope/vim-unimpaired'
23 Bundle 'tpope/vim-surround'
24 Bundle 'Raimondi/delimitMate'
25 Bundle 'nelstrom/vim-markdown-folding'
26 Bundle 'tpope/vim-repeat'
27 Bundle 'nono/vim-handlebars'
28 Bundle 'nathanaelkane/vim-indent-guides'
29 Bundle 'pangloss/vim-javascript'
30 Bundle 'milkypostman/vim-togglelist'
31 Bundle 'altercation/vim-colors-solarized'
32 Bundle 'tomasr/molokai'
33
34 filetype plugin indent on " required by Vundle
35
```

Annotations in the image:

- An orange arrow points from the line `set rtp+=~/.vim/bundle/vundle/` to the text **setup do Vundle**.
- An orange arrow points from the line `Bundle 'sjl/badwolf'` to the text **<https://github.com/sjl/badwolf>**.
- An orange arrow points from the line `filetype plugin indent on` to the text **setup do Vundle**.

The status bar at the bottom of the Vim window shows: **NORMAL** | master > vimrc | unix < utf-8 < vim | 4% | 9:1

# Bons plugins

# Bons plugins

**CtrlP** - busca de arquivos incrivelmente rápida

# Bons plugins

**CtrlP** - busca de arquivos incrivelmente rápida

**Powerline** - linha que mostra o status do Vim

# Bons plugins

**CtrlP** - busca de arquivos incrivelmente rápida

**Powerline** - linha que mostra o status do Vim

**YankRing** - navegue pelo histórico de yanks (cópias)

# Bons plugins

**CtrlP** - busca de arquivos incrivelmente rápida

**Powerline** - linha que mostra o status do Vim

**YankRing** - navegue pelo histórico de yanks (cópias)

**ack.vim** - busca no projeto (pasta) inteiro

# Bons plugins

**CtrlP** - busca de arquivos incrivelmente rápida

**Powerline** - linha que mostra o status do Vim

**YankRing** - navegue pelo histórico de yanks (cópias)

**ack.vim** - busca no projeto (pasta) inteiro

**Surround** - manipulação de parênteses, áspas, etc



# Bons plugins

**CtrlP** - busca de arquivos incrivelmente rápida

**Powerline** - linha que mostra o status do Vim

**YankRing** - navegue pelo histórico de yanks (cópias)

**ack.vim** - busca no projeto (pasta) inteiro

**Surround** - manipulação de parênteses, áspas, etc

**Fugitive** - o melhor plugin de git de todos os tempos

# Demonstração

# Conclusões...

**“Vim é um **gosto adquirido**. Você o odeia até imergir nele. Pouca gente consegue, a maioria simplesmente o odeia.”**

**- @AlexandreTorres**

**Vim é **doloroso** de aprender, mas quem se acostuma não volta atrás.**



**Pedro Franceschi**  
@pedroh96

Tentar usar o vim: pqp. Pena da minha memória muscular...

 View translation

 Reply  Delete  Favorited  More

**1**  
FAVORITE



4:04 AM - 29 Dec 12

**DICA: \$ vimtutor**

**DICA: [vimgolf.com](http://vimgolf.com)**

**Não tenha medo de tentar  
editores e ferramentas novas.**



# OBRIGADO! :)

**PEDROFRANCESCHI**

pedrohfranceschi@gmail.com

pedrofranceschi.com

@pedroh96

PORQUE VOCÊ DEVE  
APRENDER **VIM** HOJE.

**PEDROFRANCESCHI**

pedrohfranceschi@gmail.com

pedrofranceschi.com

@pedroh96