|  |  |
| --- | --- |
| I  R  Shift+r  Shift+a  esc или shift+C | Режим вставки / ввода текста. (перейти в режим i)  Замена для 1 символа  Замена последнего символа  Ввод текста вконце строки  Командный режим |
| :set autoindent (:set ai)  :set noautoindent (:set noai) | Включить выключить автовыравнивание строк: |
| :syntax on  :syntax off | Включить выключить подсветку синтаксиса: |
| set syntax=sh | Установить язык программирования если vim не распознал его автоматически: |
| :make [param]  :set autowrite on | Запустить make в текущем каталоге:  Перед запуском make желательно установить автоматическое сохранения редактируемых файлов перед компиляцией: |
| set number | показать нумерацию строк |
| set cursorline | подсвечивает курсор |
| set cursorcolumn | подсвечивает текущую строчку |
| set autoindent | копирует отступы с текущей строки и добавляет их при добавлении новой |
| set cindent | c indent = копирует отступы с текущей строки и добавляет их при добавлении новой для \*.c файлов |
| inoremap ( ()<left>  inoremap () () | добавляет ) после написания символа ( |
| inoremap { {}<Left><enter><up><end>  inoremap {} {}<Left> | добавляет } после написания символа { |
| inoremap " ""<left>  inoremap "" "" | добавляет " после написания символа " |
| highlight ExtraWhitespace ctermbg=red guibg=red  match ExtraWhitespace /\s\+$\|\s+\s{1}/ | подсвечивает красным пробелы в конце строки |
| highlight MoreThan80 ctermbg=blue guibg=blue  :2match MoreThan80 /\%81v.\+/ | подсвечивает синим строку, если та будет превышать 80 знаков |
| set tabstop=4 | определяет ширину 1ой Tab'уляции в 4 пробела |
| set shiftwidth=4 | определяет ширину 1ой Tab'уляции в 4 пробела, при сдвиге выделенного вертикального блока вправо |
| :cl  :cc [num]  :cn  :cp  [i  [d  %  Ctl-n или Ctrl-p  Ctrl-X Ctrl-L- | После завершения компиляции  список ошибок  показать ошибку с номером num или текущую  следующая ошибка  предыдущая ошибка  В режиме редактирования:  просмотреть определение (первое вхождение в тексте) переменной  просмотреть определение макроса (в языке Си)  найти парную скобку  автодополнение, подстановка слов из имеющейся программы (а в Си и из заголовочных файлов) по введенным первым буквам  автодополнение строки |
| K  J  I  N | Вверх  Вниз  Вправо  Влево |
| :term  :q! (shift+z shift+z) ctrl+c?  :w!  :qw  :wqa | Открыть терминал  Принудитель выйти  Принудительно сохранить  Сохранить и выйти  Закрыть все с сохранением |
| :help register  :help :h1  :help version7 |  |
| /  L b  X 4x  Dd. 3dd  u  $  ^  :e  :vs  ctrl-w  :sp  :q  :Stdheader + Enter  Fn + F1 | Поиск текста  Перемещение к концу слова / к началу слова  Удалить символ / Удалить 4 символа  Удалить строку / Удалить 3 строки  Вернуть последнее действие  перенесет курсор к концу строки,  перенесет курсор к началу строки.  открыть другой файл и имя файла, к примеру "test.c"  разделить зону на две  перемещение из одной области в другую (стрелками)  разделить по горизонтали  закрыть текущую область  вставить заголовок Школы 42 (со всеми си файлами)  вставить заголовок школы |

Vim

<https://ru.wikibooks.org/wiki/Vim>

https://sysdev.me/useful-vim/

eax.me/vim-commands

### 4.2 - Vim: [Vundle plugin manager](https://github.com/VundleVim/Vundle.vim) ###

Устанавливаем плагин согласно инструкции, в пункте, где надо прописывать в ~/.vimrc (если его нет, то создайте этот файл) - комментируем дефолтные плагины, которые показаны для иллюстрации.

\* Если ставили zsh, то следуя одному из пункту установки – пишем в ~/.vimrc в конце файла, под всеми настройками.

`set shell=/bin/zsh`

### 4.3 - Vim: [NERDTree + nerdtree-git-plugin](https://github.com/Xuyuanp/nerdtree-git-plugin)

1. Прописываем в ~/.vimrc в разделе плагинов для Vundle

```

Plugin 'scrooloose/nerdtree'

Plugin 'Xuyuanp/nerdtree-git-plugin'

```

2. Под Vundle прописываем настройки для плагина

```

"/NERDTree settings/

let g:NERDTreeIndicatorMapCustom = {

\ "Modified" : "✹",

\ "Staged" : "✚",

\ "Untracked" : "✭",

\ "Renamed" : "➜",

\ "Unmerged" : "═",

\ "Deleted" : ":heavy\_multiplication\_x:",

\ "Dirty" : "✗",

\ "Clean" : "✔︎",

\ 'Ignored' : '☒',

\ "Unknown" : "?"

\ }

"Stick this in your vimrc to open NERDTree with Ctrl+n (you can set whatever key you want):

"Bind ctrl+n

map <C-n> :NERDTreeToggle<CR>

"Show hidden files

let NERDTreeShowHidden=1

```

3. Для установки и обновления плагином в vim пишем ```:PluginInstall```, чтобы открыть дерево используем ctrl+n, переключаться между окон в vim ctrl+w

### 4.4 - Vim: [Monokai тема](https://github.com/tomasr/molokai) + Syntax hightlight ###

1. Создаем папку под настройки vim в домашнем каталоге, если ее там нет ```mkdir -p ~/.vim/colors```

2. Скачиваем, устанавливаем и удаляем скачанную папку

```

git clone https://github.com/sickill/vim-monokai.git ~/vim-monokai && mv -v ~/vim-monokai/colors/monokai.vim ~/.vim/colors/ && rm -Rf ~/vim-monokai

```

3. [Airline: темы стрелок](https://github.com/vim-airline/vim-airline). Прописываем в ~/.vimrc в раздел bundle:

```

Plugin 'vim-airline/vim-airline'

Plugin 'Yavor-Ivanov/airline-monokai-subtle.vim'

```

4. Прописываем в ~/.vimrc в конце дока

```

"подсветка синтаксиса

syntax enable

"/Theme settings/

"выбор цветовой схемы

colorscheme monokai

let g:airline\_theme = 'monokai\_subtle'

"если стрелки не заработали

let g:airline\_powerline\_fonts = 1

```

5. Для установки и обновления плагинов через Vundle в vim пишем ```:PluginInstall``` и перезагружаем терминал

### 4.5 - Vim: 42header ###

❗️ Плагин нужен только тем, кто работает на своем пк / ноутбуке. На компьютерах Школы все и так прекрасно работает. ❗️

Обязательный заголовок школы 42, без него Normanette будет ругаться, а Moulinette откажется проверять.

1. Создаем папку под vim плагины в домашнем каталоге, если ее там нет ```mkdir -p ~/.vim/after/plugin```

2. Скачиваем, перемещаем и удаляем из дом.каталога

```

git clone https://github.com/pandark/42header.vim.git ~/42header.vim && mv -v ~/42header.vim/plugin/42header.vim ~/.vim/after/plugin/ && rm -rf ~/42header.vim

```

3. Настраиваем $USER и $MAIL: заходим в ~/.zshrc (или ~/.bashrc) и прописываем строки

```

export USER=\*username\*

export MAIL=\*usermail\*

```

4. Перезагружаем терминал и тестим командой ```:FortyTwoHeader``` в vim

5. Чтобы поменять с :FortyTwoHeader на :Stdheader - измените строку 187 в ~/.vim/after/plugin/42header.vim на ```command! Stdheader call s:fortytwoheader ()```.

\* Чтобы забиндить Хедер на f5, пропишите ```nmap <f5> :Stdheader<CR>``` в файле ~/.vimrc (если его нет, то создайте)

### 4.6 - Vim: Code lifehack's ###

Пропишите данные команды в конце файла vim ~/.vimrc: