//User

{

"color\_scheme": "Packages/Color Scheme - Default/Monokai.tmTheme",

"font\_size": 16,

"ignored\_packages":

[

"AutoFileName",

"jQuery",

"RemoveSelection",

"Sass",

"Vintage"

],

"preview\_on\_click": false,

"spell\_check": "true",

"word\_wrap": "true"

}

//Default

// Place your settings in the file "Packages/User/Preferences.sublime-settings",

// which overrides the settings in here.

//

// Settings may also be placed in syntax-specific setting files, for

// example, in Packages/User/Python.sublime-settings for python files.

{

// Sets the colors used within the text area

"color\_scheme": "Packages/Color Scheme - Default/Monokai.tmTheme",

// Note that the font\_face and font\_size are overridden in the platform

// specific settings file, for example, "Preferences (Linux).sublime-settings".

// Because of this, setting them here will have no effect: you must set them

// in your User File Preferences.

"font\_face": "",

"font\_size": 10,

// Valid options are "no\_bold", "no\_italic", "no\_antialias", "gray\_antialias",

// "subpixel\_antialias", "no\_round" (OS X only), "gdi" (Windows only) and

// "directwrite" (Windows only)

"font\_options": [],

// Characters that are considered to separate words

"word\_separators": "./\\()\"'-:,.;<>~!@#$%^&\*|+=[]{}`~?",

// Set to false to prevent line numbers being drawn in the gutter

"line\_numbers": true,

// Set to false to hide the gutter altogether

"gutter": true,

// Spacing between the gutter and the text

"margin": 4,

// Fold buttons are the triangles shown in the gutter to fold regions of text

"fold\_buttons": true,

// Hides the fold buttons unless the mouse is over the gutter

"fade\_fold\_buttons": true,

// Columns in which to display vertical rulers

"rulers": [],

// Set to true to turn spell checking on by default

"spell\_check": false,

// The number of spaces a tab is considered equal to

"tab\_size": 4,

// Set to true to insert spaces when tab is pressed

"translate\_tabs\_to\_spaces": false,

// If translate\_tabs\_to\_spaces is true, use\_tab\_stops will make tab and

// backspace insert/delete up to the next tabstop

"use\_tab\_stops": true,

// Set to false to disable detection of tabs vs. spaces on load

"detect\_indentation": true,

// Calculates indentation automatically when pressing enter

"auto\_indent": true,

// Makes auto indent a little smarter, e.g., by indenting the next line

// after an if statement in C. Requires auto\_indent to be enabled.

"smart\_indent": true,

// Adds whitespace up to the first open bracket when indenting. Requires

// auto\_indent to be enabled.

"indent\_to\_bracket": false,

// Trims white space added by auto\_indent when moving the caret off the

// line.

"trim\_automatic\_white\_space": true,

// Disables horizontal scrolling if enabled.

// May be set to true, false, or "auto", where it will be disabled for

// source code, and otherwise enabled.

"word\_wrap": "auto",

// Set to a value other than 0 to force wrapping at that column rather than the

// window width

"wrap\_width": 0,

// Set to false to prevent word wrapped lines from being indented to the same

// level

"indent\_subsequent\_lines": true,

// Draws text centered in the window rather than left aligned

"draw\_centered": false,

// Controls auto pairing of quotes, brackets etc

"auto\_match\_enabled": true,

// Word list to use for spell checking

"dictionary": "Packages/Language - English/en\_US.dic",

// Sets which scopes are checked for spelling errors

"spelling\_selector": "markup.raw, source string.quoted - punctuation - meta.preprocessor.c.include, source comment - source comment.block.preprocessor, -(source, constant, keyword, storage, support, variable, markup.underline.link, meta.tag)",

// Set to true to draw a border around the visible rectangle on the minimap.

// The color of the border will be determined by the "minimapBorder" key in

// the color scheme

"draw\_minimap\_border": false,

// Always visualise the viewport on the minimap, as opposed to only

// showing it on mouse over

"always\_show\_minimap\_viewport": false,

// If enabled, will highlight any line with a caret

"highlight\_line": false,

// Valid values are "smooth", "phase", "blink" and "solid".

"caret\_style": "smooth",

// These settings control the size of the caret

"caret\_extra\_top": 0,

"caret\_extra\_bottom": 0,

"caret\_extra\_width": 0,

// Set to false to disable underlining the brackets surrounding the caret

"match\_brackets": true,

// Set to false if you'd rather only highlight the brackets when the caret is

// next to one

"match\_brackets\_content": true,

// Set to false to not highlight square brackets. This only takes effect if

// match\_brackets is true

"match\_brackets\_square": true,

// Set to false to not highlight curly brackets. This only takes effect if

// match\_brackets is true

"match\_brackets\_braces": true,

// Set to false to not highlight angle brackets. This only takes effect if

// match\_brackets is true

"match\_brackets\_angle": false,

// Enable visualization of the matching tag in HTML and XML

"match\_tags": true,

// Highlights other occurrences of the currently selected text

"match\_selection": true,

// Additional spacing at the top of each line, in pixels

"line\_padding\_top": 0,

// Additional spacing at the bottom of each line, in pixels

"line\_padding\_bottom": 0,

// Set to false to disable scrolling past the end of the buffer.

// On OS X, this value is overridden in the platform specific settings, so

// you'll need to place this line in your user settings to override it.

"scroll\_past\_end": true,

// This controls what happens when pressing up or down when on the first

// or last line.

// On OS X, this value is overridden in the platform specific settings, so

// you'll need to place this line in your user settings to override it.

"move\_to\_limit\_on\_up\_down": false,

// Set to "none" to turn off drawing white space, "selection" to draw only the

// white space within the selection, and "all" to draw all white space

"draw\_white\_space": "selection",

// Set to false to turn off the indentation guides.

// The color and width of the indent guides may be customized by editing

// the corresponding .tmTheme file, and specifying the colors "guide",

// "activeGuide" and "stackGuide"

"draw\_indent\_guides": true,

// Controls how the indent guides are drawn, valid options are

// "draw\_normal" and "draw\_active". draw\_active will draw the indent

// guides containing the caret in a different color.

"indent\_guide\_options": ["draw\_normal"],

// Set to true to removing trailing white space on save

"trim\_trailing\_white\_space\_on\_save": false,

// Set to true to ensure the last line of the file ends in a newline

// character when saving

"ensure\_newline\_at\_eof\_on\_save": false,

// Set to true to automatically save files when switching to a different file

// or application

"save\_on\_focus\_lost": false,

// Save via writing to an alternate file, and then renaming it over the

// original file.

"atomic\_save": false,

// The encoding to use when the encoding can't be determined automatically.

// ASCII, UTF-8 and UTF-16 encodings will be automatically detected.

"fallback\_encoding": "Western (Windows 1252)",

// Encoding used when saving new files, and files opened with an undefined

// encoding (e.g., plain ascii files). If a file is opened with a specific

// encoding (either detected or given explicitly), this setting will be

// ignored, and the file will be saved with the encoding it was opened

// with.

"default\_encoding": "UTF-8",

// Files containing null bytes are opened as hexadecimal by default

"enable\_hexadecimal\_encoding": true,

// Determines what character(s) are used to terminate each line in new files.

// Valid values are 'system' (whatever the OS uses), 'windows' (CRLF) and

// 'unix' (LF only).

"default\_line\_ending": "system",

// When enabled, hovering over a word will show a popup listing all

// possible locations for the definition symbol. Requires index\_files.

"show\_definitions": true,

// When enabled, pressing tab will insert the best matching completion.

// When disabled, tab will only trigger snippets or insert a tab.

// Shift+tab can be used to insert an explicit tab when tab\_completion is

// enabled.

"tab\_completion": true,

// Enable auto complete to be triggered automatically when typing.

"auto\_complete": true,

// The maximum file size where auto complete will be automatically triggered.

"auto\_complete\_size\_limit": 4194304,

// The delay, in ms, before the auto complete window is shown after typing

"auto\_complete\_delay": 50,

// Controls what scopes auto complete will be triggered in

"auto\_complete\_selector": "meta.tag - punctuation.definition.tag.begin, source - comment - string.quoted.double.block - string.quoted.single.block - string.unquoted.heredoc",

// Additional situations to trigger auto complete

"auto\_complete\_triggers": [ {"selector": "text.html", "characters": "<"} ],

// By default, auto complete will commit the current completion on enter.

// This setting can be used to make it complete on tab instead.

// Completing on tab is generally a superior option, as it removes

// ambiguity between committing the completion and inserting a newline.

"auto\_complete\_commit\_on\_tab": false,

// Controls if auto complete is shown when snippet fields are active.

// Only relevant if auto\_complete\_commit\_on\_tab is true.

"auto\_complete\_with\_fields": false,

// Controls what happens when pressing the up key while the first item in

// the auto complete window is selected: if false, the window is hidden,

// otherwise the last item in the window is selected. Likewise for the

// down key when the last item is selected.

"auto\_complete\_cycle": false,

// Automatically close HTML and XML tags when </ is entered.

"auto\_close\_tags": true,

// By default, shift+tab will only unindent if the selection spans

// multiple lines. When pressing shift+tab at other times, it'll insert a

// tab character - this allows tabs to be inserted when tab\_completion is

// enabled. Set this to true to make shift+tab always unindent, instead of

// inserting tabs.

"shift\_tab\_unindent": false,

// If true, the copy and cut commands will operate on the current line

// when the selection is empty, rather than doing nothing.

"copy\_with\_empty\_selection": true,

// If true, the selected text will be copied into the find panel when it's

// shown.

// On OS X, this value is overridden in the platform specific settings, so

// you'll need to place this line in your user settings to override it.

"find\_selected\_text": true,

// When auto\_find\_in\_selection is enabled, the "Find in Selection" flag

// will be enabled automatically when multiple lines of text are selected

"auto\_find\_in\_selection": false,

// When drag\_text is enabled, clicking on selected text will begin a

// drag-drop operation. This is not currently implemented under Linux.

"drag\_text": true,

//

// User Interface Settings

//

// The theme controls the look of Sublime Text's UI (buttons, tabs, scroll bars, etc)

"theme": "Default.sublime-theme",

// Set to 0 to disable smooth scrolling. Set to a value between 0 and 1 to

// scroll slower, or set to larger than 1 to scroll faster

"scroll\_speed": 1.0,

// Controls side bar animation when expanding or collapsing folders

"tree\_animation\_enabled": true,

// Controls animation throughout the application

"animation\_enabled": true,

// Makes tabs with modified files more visible

"highlight\_modified\_tabs": false,

"show\_tab\_close\_buttons": true,

// Show folders in the side bar in bold

"bold\_folder\_labels": false,

// OS X only: Set to true to disable Lion style full screen support.

// Sublime Text must be restarted for this to take effect.

"use\_simple\_full\_screen": false,

// OS X only. Valid values are true, false, and "auto". Auto will enable

// the setting when running on a screen 2560 pixels or wider (i.e., a

// Retina display). When this setting is enabled, OpenGL is used to

// accelerate drawing. Sublime Text must be restarted for changes to take

// effect.

"gpu\_window\_buffer": "auto",

// Valid values are "system", "enabled" and "disabled"

"overlay\_scroll\_bars": "system",

// Allows tabs to scroll left and right, instead of simply shrinking

"enable\_tab\_scrolling": true,

// Display file encoding in the status bar

"show\_encoding": false,

// Display line endings in the status bar

"show\_line\_endings": false,

//

// Application Behavior Settings

//

// Exiting the application with hot\_exit enabled will cause it to close

// immediately without prompting. Unsaved modifications and open files will

// be preserved and restored when next starting.

//

// Closing a window with an associated project will also close the window

// without prompting, preserving unsaved changes in the workspace file

// alongside the project.

"hot\_exit": true,

// remember\_full\_screen will allow Sublime Text to start in full screen

// mode if it was exited in full screen mode. When set to false, Sublime

// Text will never start in full screen mode.

"remember\_full\_screen": false,

// Always prompt before reloading a file, even if the file hasn't been

// modified. The default behavior is to automatically reload a file if it

// hasn't been edited. If a file has unsaved changes, a prompt will always

// be shown.

"always\_prompt\_for\_file\_reload": false,

// OS X only: When files are opened from finder, or by dragging onto the

// dock icon, this controls if a new window is created or not.

"open\_files\_in\_new\_window": true,

// OS X only: This controls if an empty window is created at startup or not.

"create\_window\_at\_startup": true,

// Set to true to close windows as soon as the last file is closed, unless

// there's a folder open within the window.

// On OS X, this value is overridden in the platform specific settings, so

// you'll need to place this line in your user settings to override it.

"close\_windows\_when\_empty": false,

// Show the full path to files in the title bar.

// On OS X, this value is overridden in the platform specific settings, so

// you'll need to place this line in your user settings to override it.

"show\_full\_path": true,

// Shows the Build Results panel when building. If set to false, the Build

// Results can be shown via the Tools/Build Results menu.

"show\_panel\_on\_build": true,

// Shows build errors just under the line on which they occur.

"show\_errors\_inline": true,

// Preview file contents when clicking on a file in the side bar. Double

// clicking or editing the preview will open the file and assign it a tab.

"preview\_on\_click": true,

// folder\_exclude\_patterns and file\_exclude\_patterns control which files

// are listed in folders on the side bar. These can also be set on a per-

// project basis.

"folder\_exclude\_patterns": [".svn", ".git", ".hg", "CVS"],

"file\_exclude\_patterns": ["\*.pyc", "\*.pyo", "\*.exe", "\*.dll", "\*.obj","\*.o", "\*.a", "\*.lib", "\*.so", "\*.dylib", "\*.ncb", "\*.sdf", "\*.suo", "\*.pdb", "\*.idb", ".DS\_Store", "\*.class", "\*.psd", "\*.db", "\*.sublime-workspace"],

// These files will still show up in the side bar, but won't be included in

// Goto Anything or Find in Files

"binary\_file\_patterns": ["\*.jpg", "\*.jpeg", "\*.png", "\*.gif", "\*.ttf", "\*.tga", "\*.dds", "\*.ico", "\*.eot", "\*.pdf", "\*.swf", "\*.jar", "\*.zip"],

// File indexing parses all files in the side bar, and builds an index of

// their symbols. This is required for Goto Definition to work.

"index\_files": true,

// Set the number threads to use for indexing. A value of 0 will make

// Sublime Text guess based on the number of cores. Use the index\_files

// setting to disable all workers.

"index\_workers": 0,

// index\_exclude\_patterns indicate which files won't be indexed.

"index\_exclude\_patterns": ["\*.log"],

// List any packages to ignore here. When removing entries from this list,

// a restart may be required if the package contains plugins.

"ignored\_packages": ["Vintage"]

}