

CSS Nite LP, Disk 26 "CSS Preprocessor Shootout"

# 新世代エディタで作る プリプロセッサ環境構築術 テキストブック

たにぐち まこと



## はじめに

本書は、2013年 1月 12日に開催された CSS Nite LP26 「CSS Preprocessor Shootout」でのセッション「新世代エディタで作る、プリプロセッサ環境構築術」での補助資料です。

本書を読むだけでも、役に立つ資料になるように努めましたと言葉が足りない部分などがあるかも知れません。ご了承ください。

H2O SPACE たにぐち まこと

<http://tomosta.jp>

## CSS Niteについて



2005年10月、アップルストア銀座にてマンスリーイベントとしてスタート。当初は文字通り「CSS」にフォーカスしていたが、現在は“Web制作者にとって有益であろうトピック”を広く扱っている。

名古屋、大阪、青森、福岡、沖縄、秋田、札幌、福井、仙台、福島、広島、新潟、京都、石川、高松、山形、富山、静岡、宮崎、熊本、北九州、岡山、徳島、松山、山梨、長野にて“地方版”を展開。地方版は現地の有志が運営し、地方のコミュニティ活性化の一助となっている。

これまでに、300回を超える関連イベントを通して、のべ40,000名を超える方にご参加いただいている（2012年11月現在）。

<http://cssnite.jp>

# Sublime Text 2



## Sublime Text 2とは

Sublime Text 2 (以下、ST2) は、Windows/OS X/Linux向けに開発されているエディタソフトです。

シンプルながら、非常に高い柔軟性と多くの「パッケージ」と呼ばれる拡張機能が魅力で、2011年にバージョン2が登場してから、爆発的にユーザーを伸ばしています。

## ST2のインストール

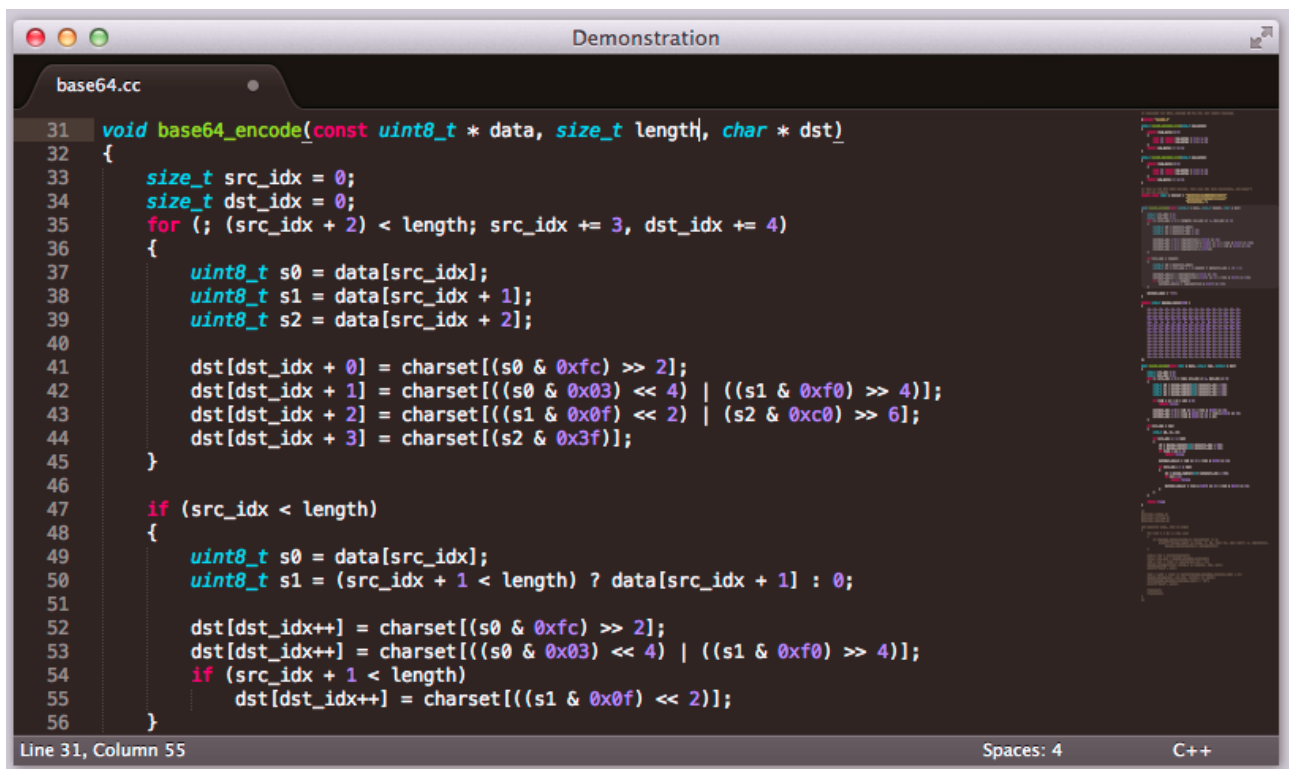
ST2は、難しい手続きを必要とすることなく、通常のダウンロードソフトと同様にセットアップすることができます。

また、Windows版にはUSBメモリなどにセットアップして、手軽に持ち歩ける「ポータブル版」も配布されています。いずれも、ダウンロードサイトから入手することができます。

Sublime Text 2

<http://www.sublimetext.com/2>

セットアップをして、起動すればすぐに使い始めることができますが、そのまま使うだけではSublime Text 2の魅力を感じることはできません。多くの「パッケージ」を入れて、カスタマイズしていく必要があります。



```
base64.cc
31 void base64_encode(const uint8_t * data, size_t length, char * dst)
32 {
33     size_t src_idx = 0;
34     size_t dst_idx = 0;
35     for (; (src_idx + 2) < length; src_idx += 3, dst_idx += 4)
36     {
37         uint8_t s0 = data[src_idx];
38         uint8_t s1 = data[src_idx + 1];
39         uint8_t s2 = data[src_idx + 2];
40
41         dst[dst_idx + 0] = charset[(s0 & 0xfc) >> 2];
42         dst[dst_idx + 1] = charset[((s0 & 0x03) << 4) | ((s1 & 0xf0) >> 4)];
43         dst[dst_idx + 2] = charset[((s1 & 0x0f) << 2) | (s2 & 0xc0) >> 6];
44         dst[dst_idx + 3] = charset[(s2 & 0x3f)];
45     }
46
47     if (src_idx < length)
48     {
49         uint8_t s0 = data[src_idx];
50         uint8_t s1 = (src_idx + 1 < length) ? data[src_idx + 1] : 0;
51
52         dst[dst_idx++] = charset[(s0 & 0xfc) >> 2];
53         dst[dst_idx++] = charset[((s0 & 0x03) << 4) | ((s1 & 0xf0) >> 4)];
54         if (src_idx + 1 < length)
55             dst[dst_idx++] = charset[((s1 & 0x0f) << 2)];
56     }
}
Line 31, Column 55 Spaces: 4 C++
```

なお、本書に出てくる画面ショットは、筆者がカスタマイズを施した状態のものを掲載しており、セットアップ直後の見た目とは異なっています。設定の変更・テーマの変更については後述します。

## Sublime Package Controlをインストールする

パッケージは、それぞれのダウンロードサイトからダウンロードして手動でインストールすることもできますが、現在では「Sublime Package Control」という機能を利用するのが一般的です。

Sublime Package Control

[http://wbond.net/sublime\\_packages/package\\_control](http://wbond.net/sublime_packages/package_control)

このサイトにアクセスし、画面右の「Install」ボタンをクリックすると次のような画面が表示されます。

## Installation

Installation is through the Sublime Text 2 console. This is accessed via the `ctrl+`` shortcut. Once open, paste the following command into the console:

```
import urllib2,os; pf='Package Control.sublime-package'; ipp=sublime.installed_packages_path(); os.makedirs(ipp) if not os.path.exists(ipp) else None; urllib2.install_opener(urllib2.build_opener(urllib2.ProxyHandler())); open(os.path.join(ipp,pf),'wb').write(urllib2.urlopen('http://sublime.wbond.net/'+pf.replace(' ','%20')).read()); print 'Please restart Sublime Text to finish installation'
```

この、グレーのボックス内に記述されているコードをコピーし、ST2のメニューから「View→Show Console」を選び、表示されたエリアにこのコードを貼り付けます。





```
blime.wbond.net/'+pf.replace(' ', '%20')).read()); print 'Please restart Sublime Text to fi
```

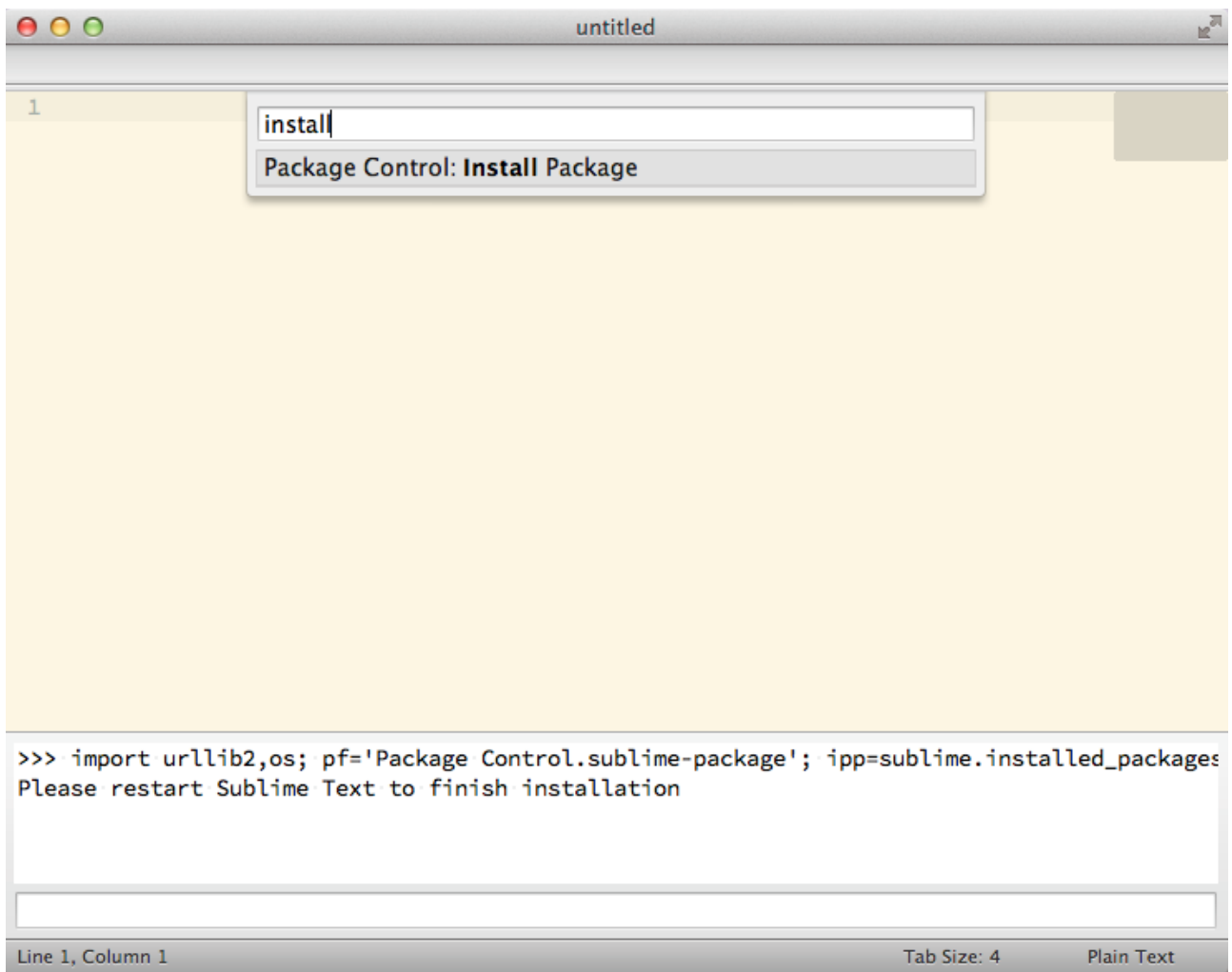
Line 1, Column 1      Tab Size: 4      Plain Text

Enterを押してしばらくすると「Please restart Sublime Text to finish installation」と表示されるので、ST2を再起動します。これで、Package Controlが利用できるようになりました。

## Package Controlを利用する

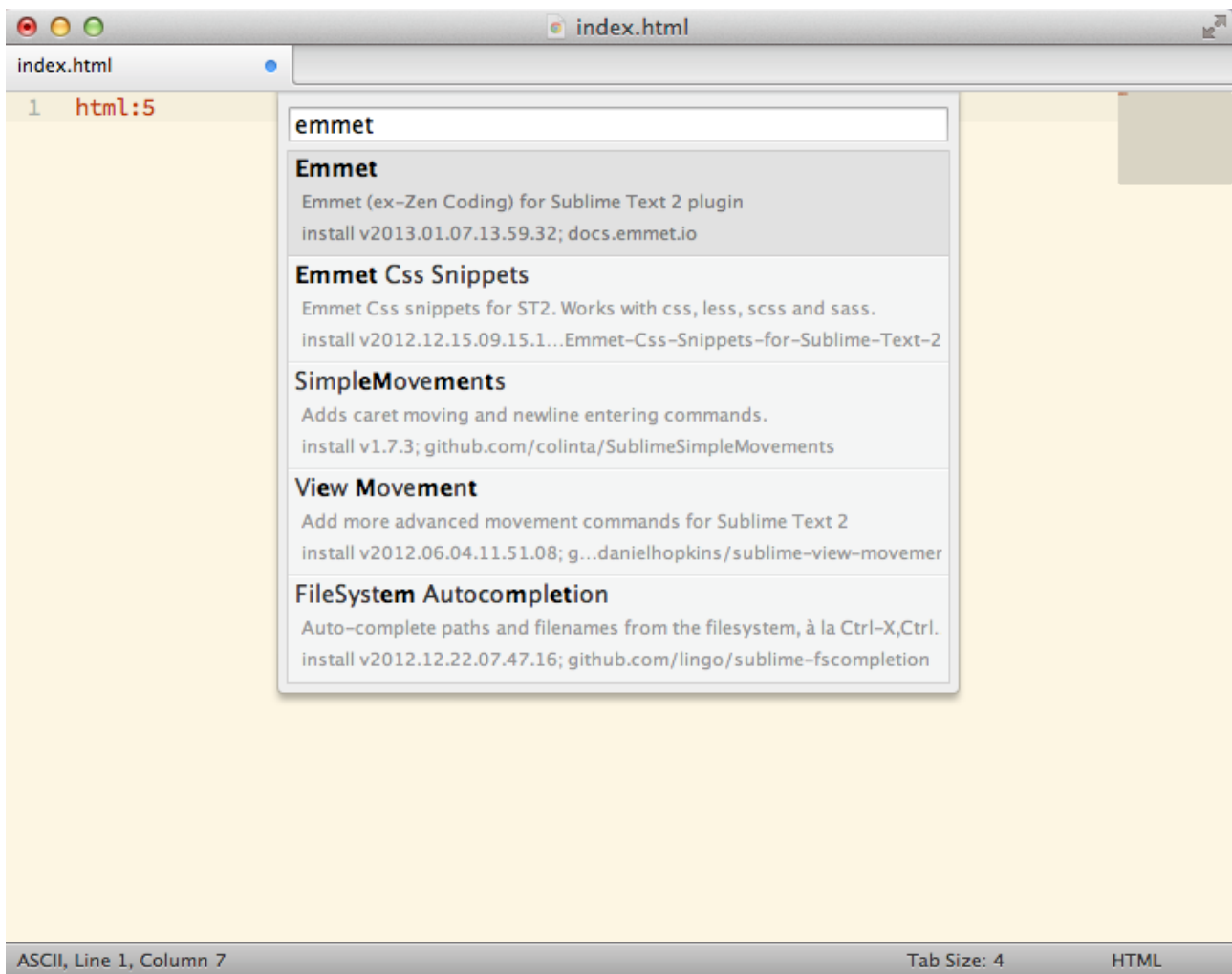
再起動した ST2のメニューから「Tools→Command Palette」を選びます（または、ショートカットの⌘ (Ctrl) +Shift+p)。

ここでは、ST2上で実行できるさまざまなコマンドを選ぶことができます。一部を打ち込めば、部分検索で絞り込むことができるので、ここでは「install」を入力しましょう。「Package Control: Install Package」と表示されれば、正常に動作しています。



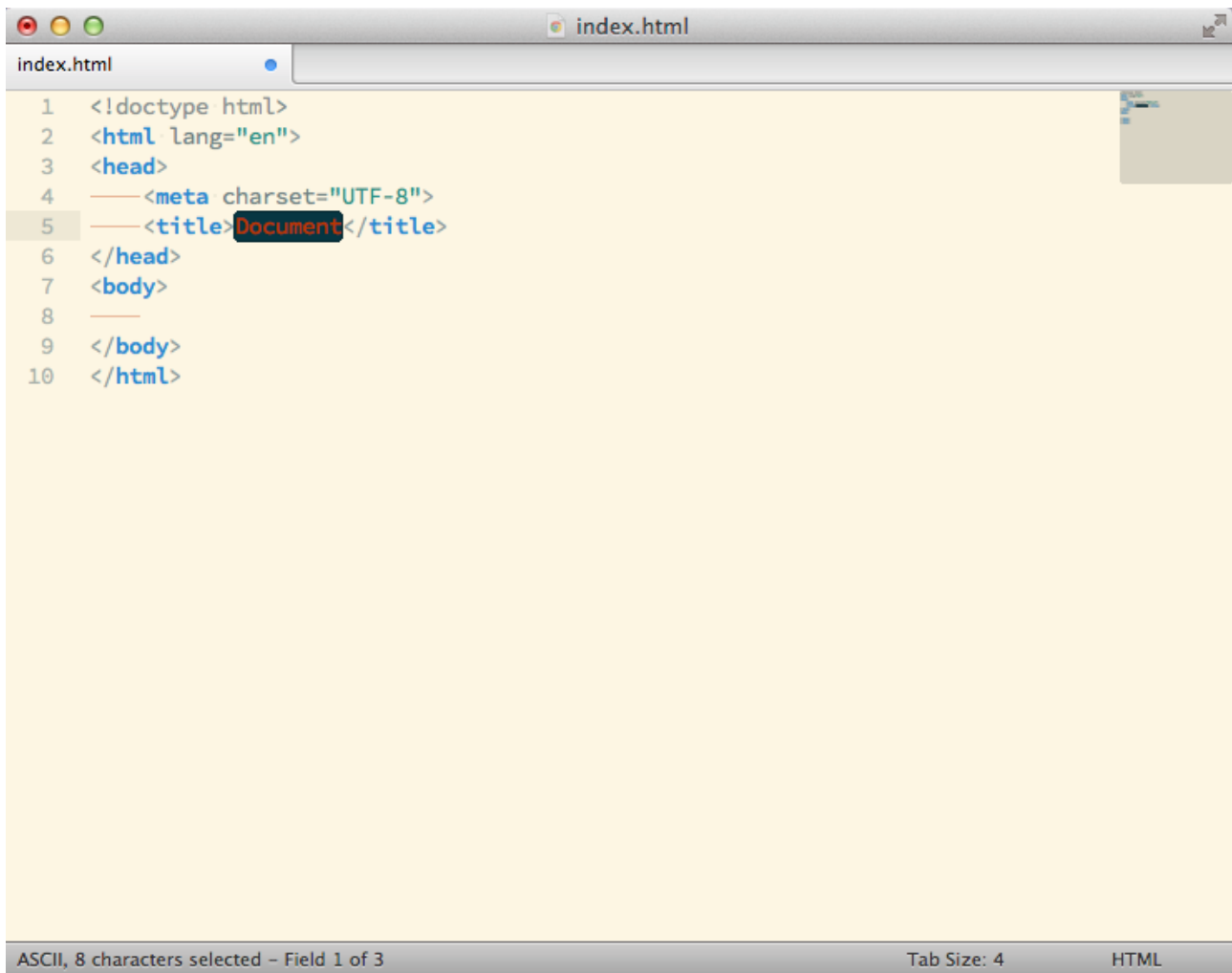
Enterを押すと、さらにリストが表示されます。これが、ST2にインストールできるパッケージの一覧です。リストから選んでインストールすることができます。先のコマンドと同様に、名前の一部から絞り込むこともできます。

例えばここでは、「emmet」と入力してみましょう。



「Emmet」というパッケージが見つかるので、Enterを押してインストールします。Emmetは短い記述で、HTMLを展開することができる手法で「Zen Coding」の次世代バージョンです。

インストールが完了したら、エディタに「html:5」と入力して「Control+E」と入力しましょう。次のように、HTML5のタグが展開されます。

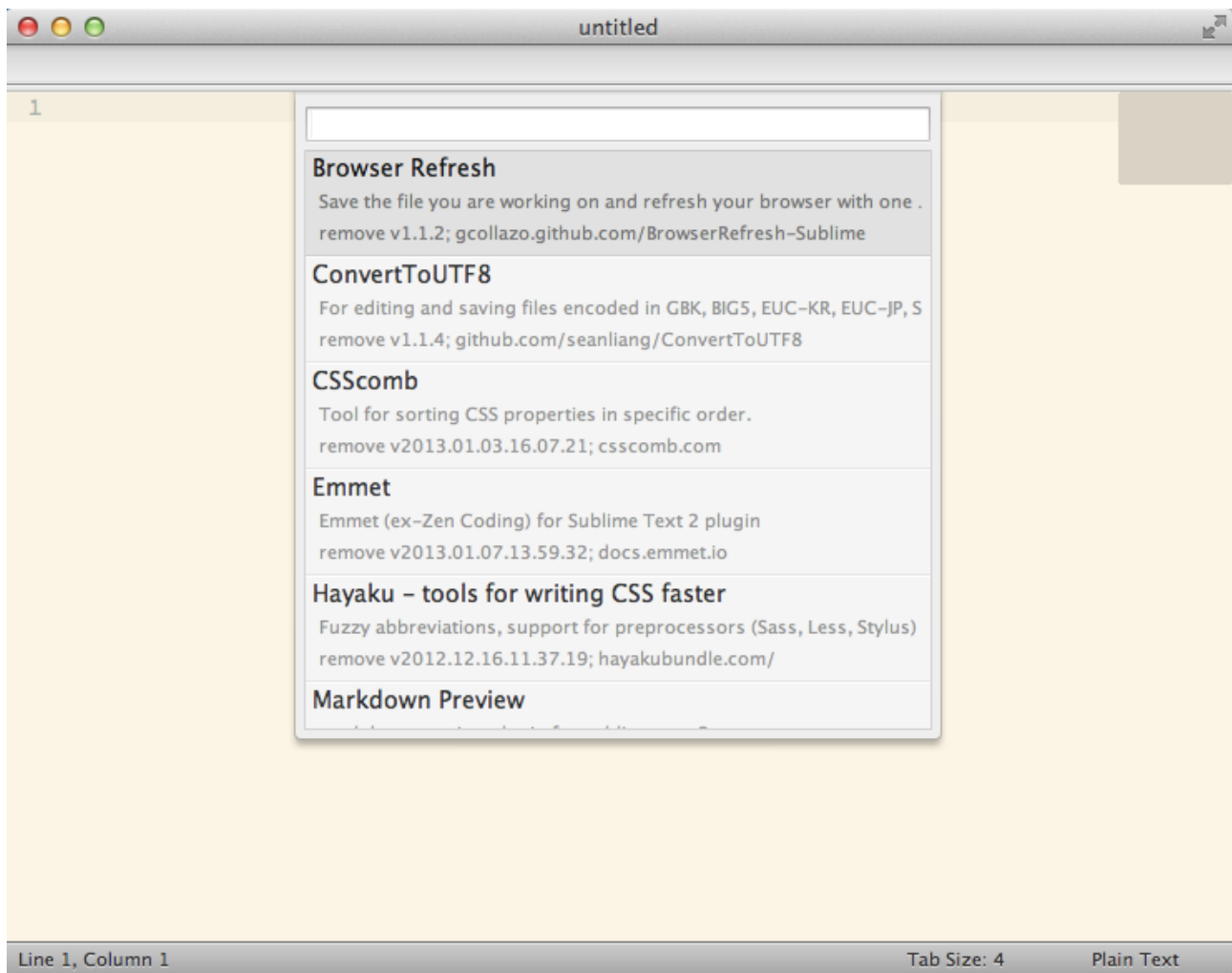
A screenshot of a code editor window titled "index.html". The editor shows a list of HTML5 tags with line numbers 1 through 10. The tags are: 1 <!doctype html>, 2 <html lang="en">, 3 <head>, 4 <meta charset="UTF-8">, 5 <title>Document</title>, 6 </head>, 7 <body>, 8 </body>, 9 </body>, 10 </html>. The word "Document" in the title tag is highlighted with a black background. The status bar at the bottom indicates "ASCII, 8 characters selected - Field 1 of 3", "Tab Size: 4", and "HTML".

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Document</title>
6 </head>
7 <body>
8 </body>
9 </body>
10 </html>
```

こうして、好きなパッケージを追加することで、自分好みのエディタに仕上げることができます。

## パッケージを管理する

もし、パッケージを間違えてインストールした場合や、正常に動作しない場合などはコマンドパレット (⌘+Shift+P) で、「remove」などと打ち込んで「Remove Package」を選べば、今インストールされているパッケージが一覧されるので、ここから削除することができます。



この他、次のようなコマンドを利用することができます。

- ・ Install Package  
パッケージのインストール
- ・ Remove Package  
パッケージの削除
- ・ Enable Package  
パッケージの有効化
- ・ Disable Package  
パッケージの無効化
- ・ Upgrade Package  
パッケージのアップグレード

## CSSプリプロセッサに便利なパッケージを入れる

それでは、この手順でCSSプリプロセッサや、その他のWeb制作に便利なパッケージをインストールしていきましょう。次のようなパッケージ類を入れておくと良いでしょう。

- ・ HTML5
- ・ CSS Snippets
- ・ LESS
- ・ SASS
- ・ SCSS
- ・ SCSS Snippets
- ・ JavaScript Snippets
- ・ jQuery
- ・ Emmet
- ・ Browser Refresh
- ・ CSScomb

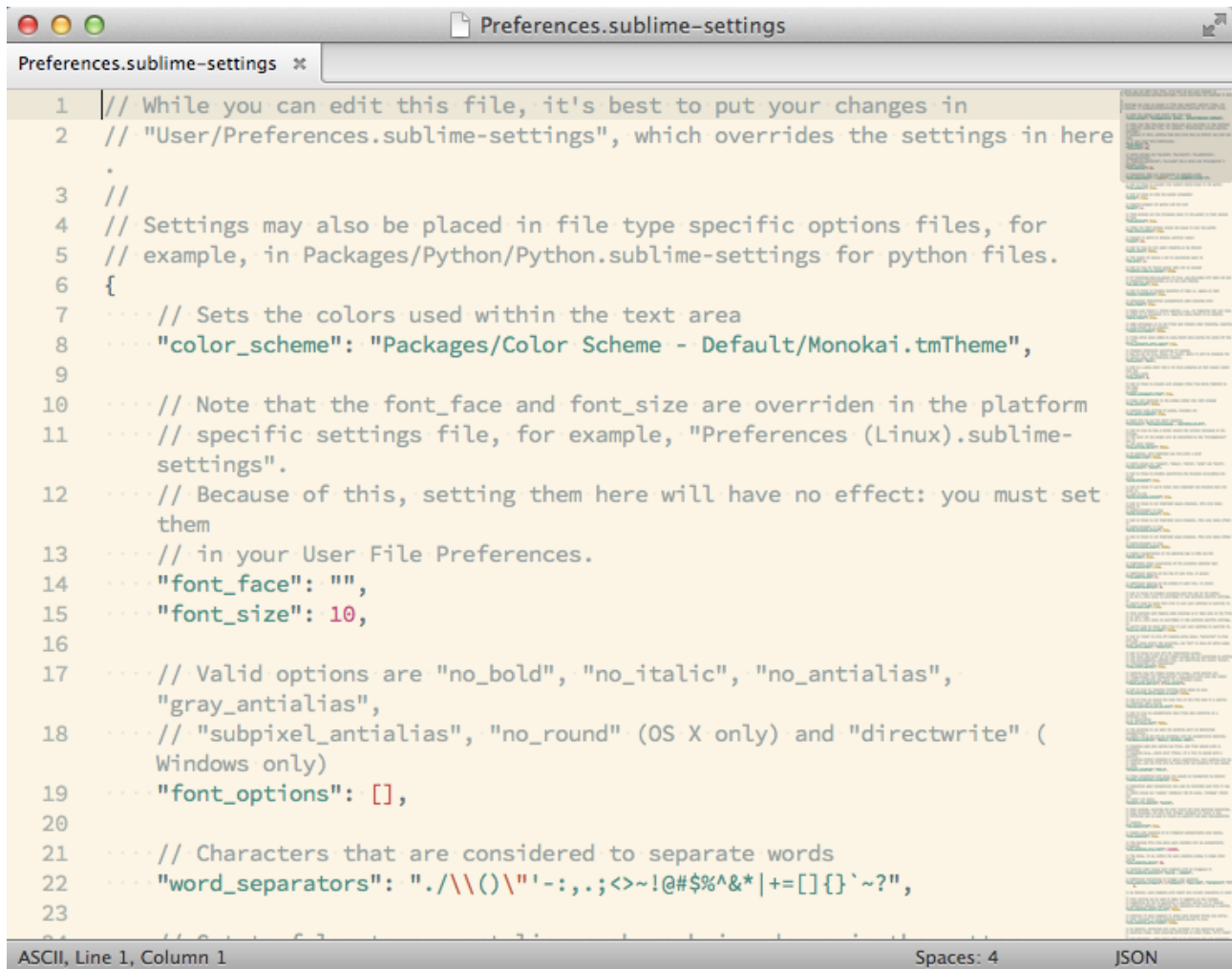
この他にも、無数のパッケージが配布されています。こもりまさあきさんが、公開している Web 制作者向けのパッケージリストなども参考にすると良いでしょう。

ST2: Sublime Text 2 Packages for Web Developers.

<https://gist.github.com/4048934>

## 設定を変更する

STの設定を行う場合、テキストファイルを書き換えることで行います。Windows版の場合はメニューバーの「Preferences」から（OS X版の場合は一番左のアプリケーション名メニュー内に「Preferences」があります）、「Settings - Default」を選びます。すると、ST2上に設定ファイルが展開されます。



```
1 // While you can edit this file, it's best to put your changes in
2 // "User/Preferences.sublime-settings", which overrides the settings in here
3 //
4 // Settings may also be placed in file type specific options files, for
5 // example, in Packages/Python/Python.sublime-settings for python files.
6 {
7     ....// Sets the colors used within the text area
8     ...."color_scheme": "Packages/Color Scheme - Default/Monokai.tmTheme",
9
10    ....// Note that the font_face and font_size are overridden in the platform
11    ....// specific settings file, for example, "Preferences (Linux).sublime-
12    ....// settings".
13    ....// Because of this, setting them here will have no effect: you must set
14    ....// them
15    ....// in your User File Preferences.
16    ...."font_face": "",
17    ...."font_size": 10,
18
19    ....// Valid options are "no_bold", "no_italic", "no_antialias",
20    ....// "gray_antialias",
21    ....// "subpixel_antialias", "no_round" (OS X only) and "directwrite" (
22    ....// Windows only)
23    ...."font_options": [],
24
25    ....// Characters that are considered to separate words
26    ...."word_separators": ". / \\ ( ) \\' \" - : , ; < > ~ ! @ # $ % ^ & * | + = [ ] { } ` ~ ? " ,
```

これを書き換えても構いませんが、ST2をバージョンアップしたときなどに上書きされてしまうため、別途「Preferences→Settings - User」とメニューを選んで開く空のファイルの方に書き込みます。このファイルは、ユーザー専用の設定ファイルで ST2本体とは別の場所に保存されます。

ここに、上書きしたい設定内容をコピーして内容を書き換えます。例えば、標準のフォントは「font\_face」という項目名になっていて、なにも指定されていませんが、ここに自分好みのフォントを指定する場合はこれをコピーし、次のように書き込みます。

```
{
  "font_face": "Source Code Pro",
}
```



保存すると、すぐに反映されますが、設定によっては再起動する必要がある場合もあります。ここでは、Adobeが無償配布しているプログラミング用フォント「Source Code Pro」を指定しました。

Source Code Pro

<https://github.com/adobe/source-code-pro>

この他、筆者の設定ファイルは次のようになっています。

```
{
  "close_windows_when_empty": true,
  "color_scheme": "Packages/User/Solarized (light).tmTheme",
  "draw_white_space": "all",
  "fallback_encoding": "Shift-JIS",
  "font_face": "Source Code Pro",
  "font_size": 14.0,
  "highlight_line": true,
  "highlight_modified_tabs": true,
  "ignored_packages":
  [
    "Vintage"
  ],
  "indent_guide_options":
  [
    "draw_active"
  ],
  "open_files_in_new_window": false,
  "shift_tab_unindent": true,
  "theme": "Soda Light.sublime-theme",
  "trim_trailing_white_space_on_save": true,
  "word_wrap": true
}
```

以下に最新の設定ファイルを公開しています。

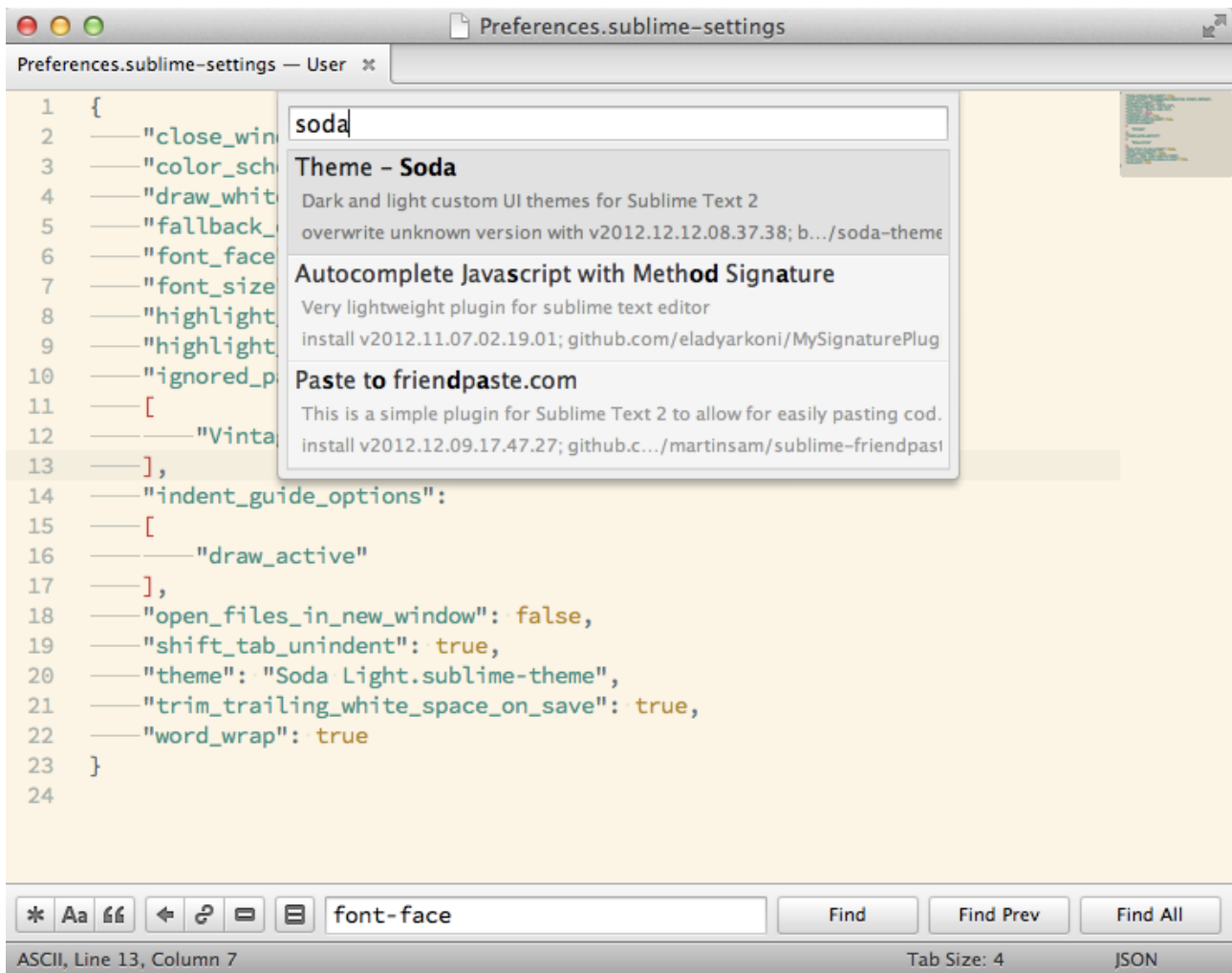
<https://gist.github.com/4408074>

### \* テーマを変更する

ST2は、「テーマ」という概念で見た目を管理しており、誰でも自由に見た目をカスタマイズすることができます。多くのクリエイタがカスタマイズ済のテーマを配布しているため、これをダウンロードして利用するだけでも気分を変えることができます。

筆者は、「Soda」というテーマを利用しています。テーマは、パッケージと同じく Package Controllerでインストールすることができます。

Package Controllerを起動し、「Package Control:Install Package」を起動したら「Soda」と打ち込んでインストールします。



そして、前節に従って設定ファイルに次のように書き込みます（先のサンプルにはすでに反映されています）。

```
"theme": "Soda Light.sublime-theme",
```

こうして、ST2を再起動すると反映されます。さらに筆者は、カースキーマとして「SOLARIZED」を組み合わせて利用しています。

SOLARIZED

<http://ethanschoonover.com/solarized>

これは、手作業でインストールする必要があります。

上記サイトの真ん中あたりに「Click Here To Download Latest Version」というリンクがあるので、ZIPファイルをダウンロードして解凍します。

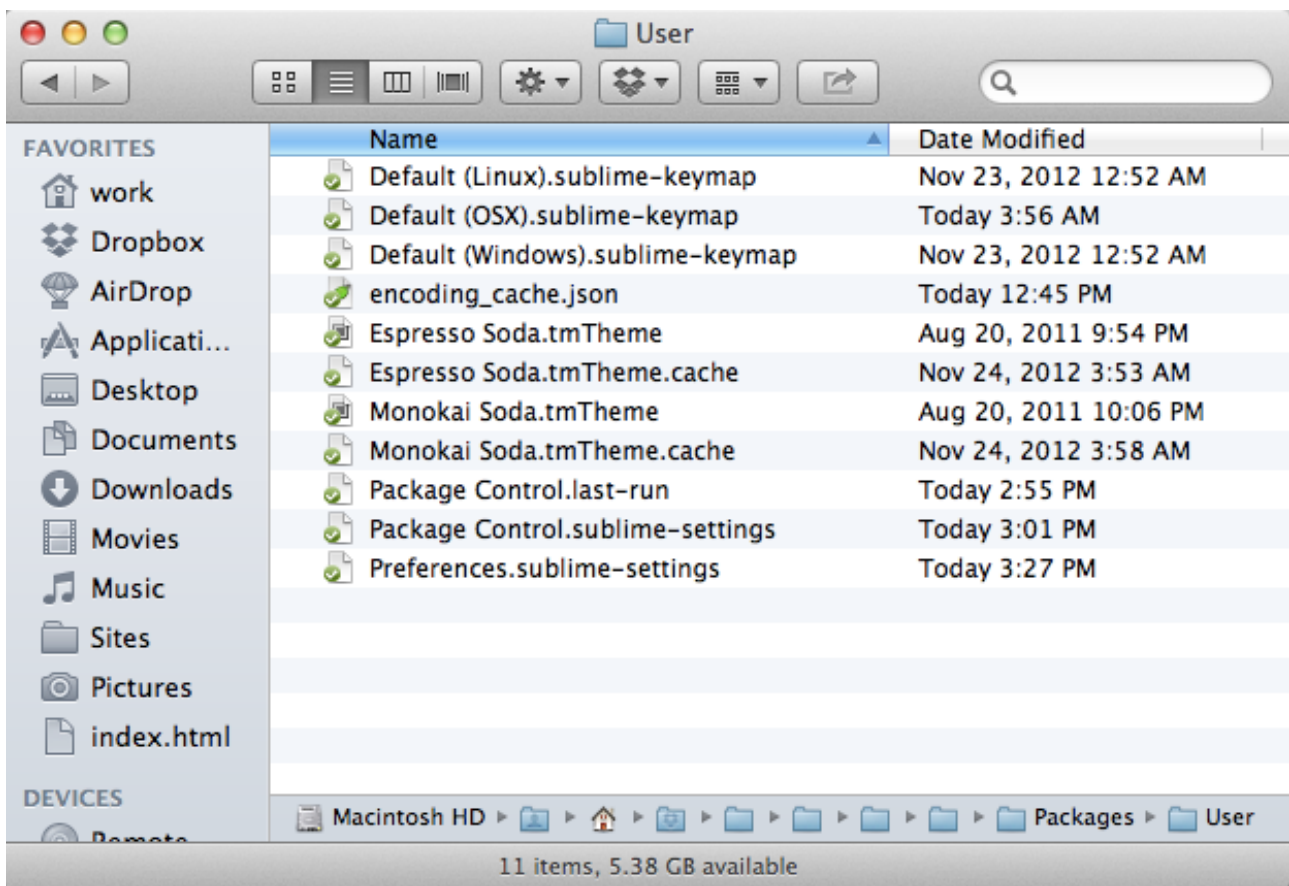
## Download



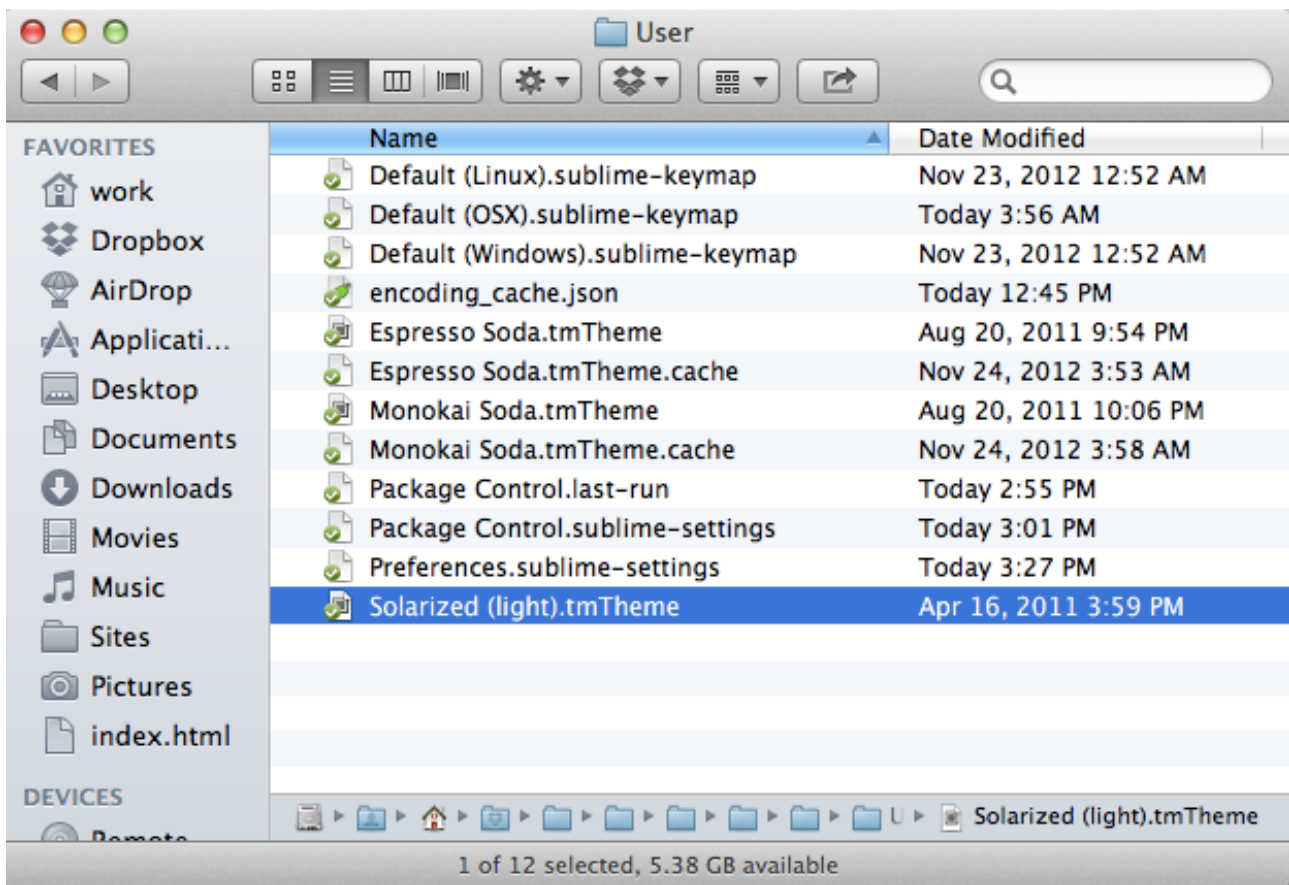
[Click Here To Download Latest Version](#)

Current release is **v1.0.0beta2**. See the [changelog](#) for details on what's new in this release.

ST2を起動し、「Preferences→Browse Packages...」メニューを選びます。パッケージを格納するフォルダが開きます。そしたら「User」フォルダを開きましょう。



ここに、スキーマファイルをコピーします。ここでは、解凍した配布ファイル内にある「textmate-colors-solarized」フォルダ内の「Solarized(light).tmTheme」ファイルをコピーします。



そして、「Preferences→Settings - User」メニューで設定ファイルに、次の記述を追加します。

```
"color_scheme": "Packages/User/Solarized (light).tmTheme",
```

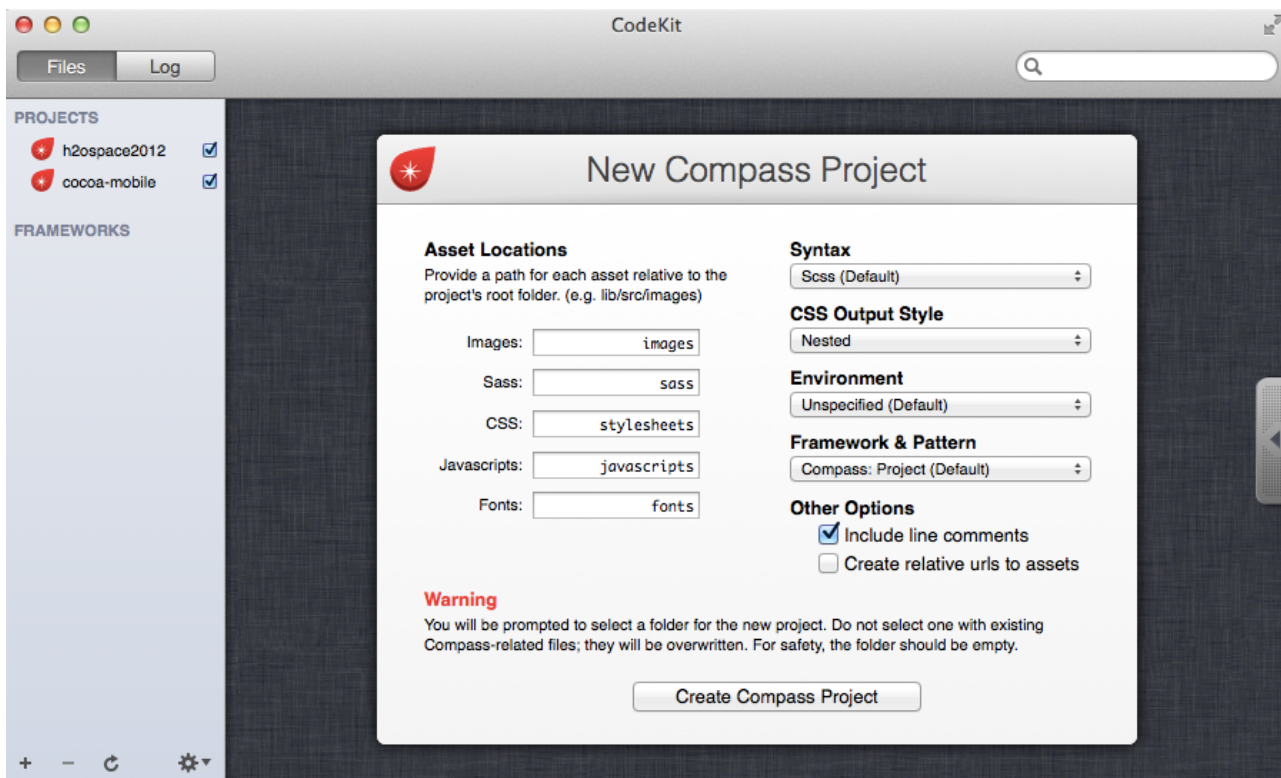
これで、OS Xらしい見た目が変わります。好みに合わせて、カスタマイズをしてみると良いでしょう。

## プロジェクトを作成する

ここまで準備ができたなら、いよいよ実際に使い始めることができます。通常のエディタと同様に、ファイルを開いて編集し、上書き保存をすれば使うことができますが、「プロジェクト」という機能を使えば、より管理しやすく利用できます。

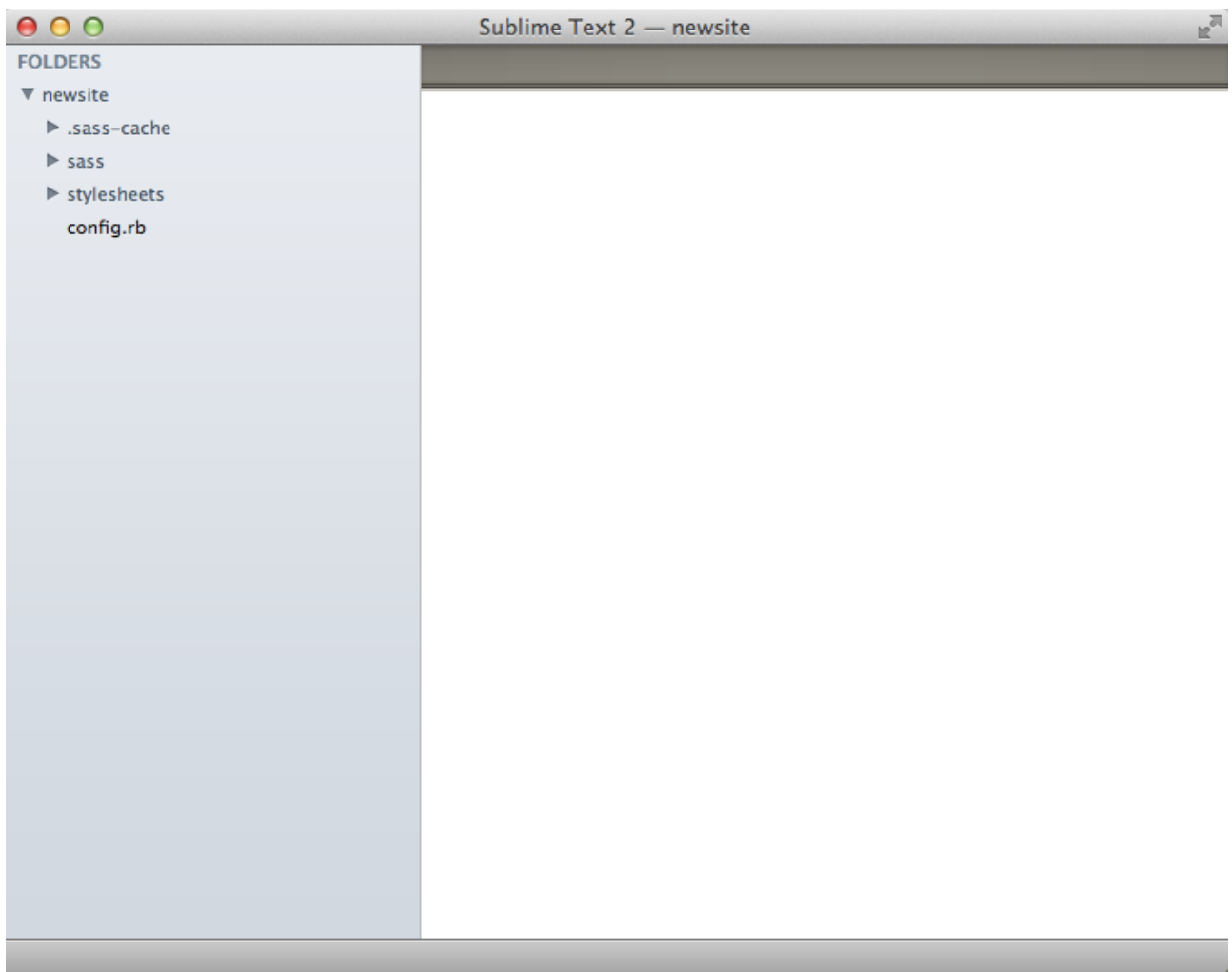
例えばここでは、CodeKitを使って新しい Compass Projectを作成します。なお本書では、CodeKit、Compassについての詳細は割愛します。

CodeKitを起動して、「File→Create New Compass Project...」を選びます。



「Create Compass Project」ボタンをクリックすると、保存先のフォルダを問われるので、新しいフォルダを作るなどしてCompass Projectを作成します。

できあがったフォルダは、そのまま ST2にドラッグドロップしましょう。すると、ST2の画面左側にそのフォルダの階層リストが表示されます。



フォルダ名を右クリック（または Control+クリック）すると、新しいフォルダ・ファイルを作成したり、名前の変更や削除なども可能です。

さらにこの状態で、メニューの「Project→Save Project As...」を選択しましょう。名前などを入力し、「プロジェクト」として今の状態を保存することができます。次回からは、「Project→Open Project」メニューまたは、「Project→Recent Project」から開くことができます。

## Coda 2



### Coda 2とは

Coda 2は、米Panic社が開発する OS X専用の Webサイト制作ソフトです。コードエディタ機能はもちろん、サイト管理機能やFTPクライアント・MySQL、SSHクライアントなど、非常に多くの機能を持ち合わせている軽量でパワフルなアプリです。

なお、本書は筆者が執筆した「Codaでつくる Webサイト制作実践 (H2O SPACE刊)」からの抜粋で構成されています。Coda 2をより深く知りたい場合には、こちらも合わせてご覧下さい。

Codaでつくる Webサイト制作実践

<http://p.booklog.jp/book/51419>

### セットアップから使い始めるまで

それでは、早速Codaを使い始めてみましょう。Codaは、OS X専用のソフトです。読者の方の中には、Windowsでなければ検討できないと考える方もいるかもしれませんが、近年、Web制作の現場では急激にMacが浸透して来ています。筆者など、もはやMacでなければ仕事にならない状況です。

一昔前は、かっこよさのために、多少無理してMacを使っているということも否めませんでした。が、現在ではMacの方が制作・開発マシンとして優れていると言い切ることができます。

Windowsが使い慣れているから、会社がWindowsしか利用できない環境だからなどという理由でMacを利用しないのは、非常にもったいなといえます。ぜひ、この機会に導入を検討しましょう。

### \* AppStore版と直販版

Codaには、OS X (Lion以降) に搭載されている AppStoreを通じて販売されるものと、直販のものがあります。それぞれの違いを紹介しましょう。

#### ・ iCloud連携機能が搭載[AppStore版のみ]

AppStore版の Codaには、直販版にはない機能として「iCloudによる同期機能」が搭載されています。これにより「サイト定義」と「クリップ」機能を同期することができます。(後述)



- ・ワンタッチバージョンアップ[AppStore版のみ]

AppStoreで購入したアプリケーションソフトは、AppStoreを通じてバージョンアップ版が配信され、ワンタッチでバージョンアップすることができます。

- ・試用版[直販版のみ]

AppStoreでの購入は、試用という制度がないため、購入するとすぐに決済されてしまいます。直販版の場合は、7日間の試用期間があります。なお、試用期間中は直販版を利用した後、いったんアンインストールをしてAppStore版を購入することもできます（設定は引き継がれません）。

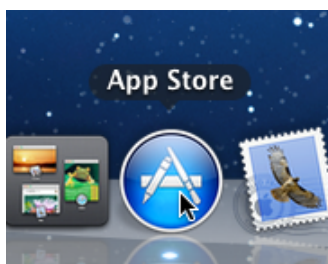
- ・バージョンアップが早い[直販版]

バージョンアップは、直販版もAppStore版も同時期に行われますが、AppStore版の場合はアップルによる審査を経ての公開となるため、直販版の方が素早く公開されます。

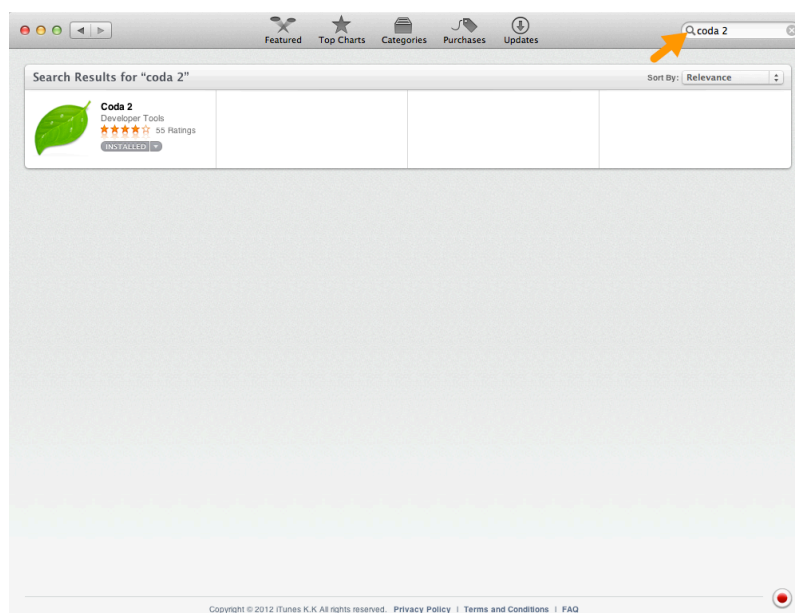
どちらも一長一短がありますが、筆者はiCloudの同期機能などの魅力があるため、AppStore版をオススメします。

## AppStoreを利用する

図のアイコンをクリックして起動し、検索窓に「Coda」と記入して検索しましょう。その場で決済して購入することができます。（Apple ID及び、クレジットカード登録が必要です。）



クレジットカードをお持ちでない方は、電気店やコンビニエンスストアで購入できる、iTunes Cardのコードで購入することもできます。詳しくは、AppStoreのヘルプなどをご覧ください。



## 直販版を購入する

以下のWebサイトにアクセスして、セットアップファイルを手に入れましょう。



## 起動する

Codaを起動すると、図のようなサイト管理画面が表示されます。



しかし、この画面はCodaにとって、必須の画面ではありませんので、無視して新しいファイルを作成しましょう（「⌘+N」）。エディタモードに切り替わります。

これだけで、Codaを利用する準備が整いました。後は、通常のテキストエディタと同様に使い始めて良いでしょう。少しHTMLやPHPなどを書き始めただけで、タグの補完をしてくれたり、属性

を色分けしてくれたりなど、エディタとしての便利さを実感できるはずです。慣れてきたら、その他の各機能も徐々に親しんでいきましょう。

## プラグイン

Codaは、そのまま利用しても十分便利なソフトウェアですが、他のエディタソフトに比べると、ちょっと機能が物足りない場合や、使いにくい部分があるかもしれません。

しかし、そんな時は「プラグイン」を利用することで解消することができます。Codaはプラグインの仕様を一般に公開しており、誰でも開発することができます。また、Codaのオフィシャルサイトでもプラグインをリストで紹介されていて、ここを見るだけでも多くのプラグインを利用することができます。

パニック - Coda 2 - プラグイン

<http://panic.com/jp/coda/plugins.php>

便利そうなプラグインがあったら、試してみましょう。

### \* プラグインをインストールする

プラグインは、ダウンロードすると図のような「.codaplugin」という拡張子のファイルが現れます。



このファイルを、ダブルクリックするだけでインストールすることができます。Codaが起動し（すでに起動している場合はウィンドウが切り替わり）、ほとんどのプラグインでは「プラグイン」メニュー内に専用のメニューが追加されています。

その他、キーボードショートカットに割り当たっていたり、右クリックメニューが追加されている場合などもありますが、各プラグインのドキュメントや配布サイトを確認しましょう。

### \* Zen Codingを利用しよう

ここでは、数あるプラグインの中でも必携の一つである、Zen Codingを利用できるようにしましょう。Codaでの Zen Codingプラグインには、「TEA for Coda」というプラグインがあります。次のサイトからダウンロードしましょう。

Zen.Coding-Coda

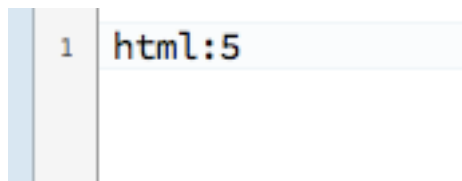
<http://code.google.com/p/zen-coding/downloads/detail?name=Zen.Coding-Coda.v0.7.1.zip>

ダウンロードしたら「.codaplugin」という拡張子のファイルをダブルクリックしてインストールします。すると、「プラグイン→TEA for Coda」というメニューが追加され、その中に Zen Codingの項目があります。

### \* 使ってみよう

Zen Codingとは、HTMLのショートカットコードのようなものを打ち込んで変換することで、自動的にHTMLに展開してくれるというものです。ここでは、新しいHTMLファイルを作成して、次のように打ち込んでみましょう。

html:5



そして、「Control+E」を押します。すると、次のようなHTMLコードに展開されます。

```
<!DOCTYPE HTML>
<html lang="en-US">
<head>
  <meta charset="UTF-8">
  <title></title>
</head>
<body>

</body>
</html>
```

また、新しいCSSを作成して次のように打ち込んでも、展開することができます。

bg+

次のようなプロパティになります。

```
background:#FFF url() 0 0 no-repeat;
```

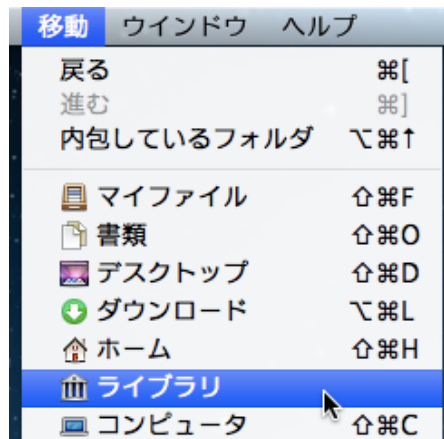
こうして、それぞれのショートコードを覚えていけば、非常に素早くコーディングをしていくことが可能になります。

### \* Zen Codingをカスタマイズしよう

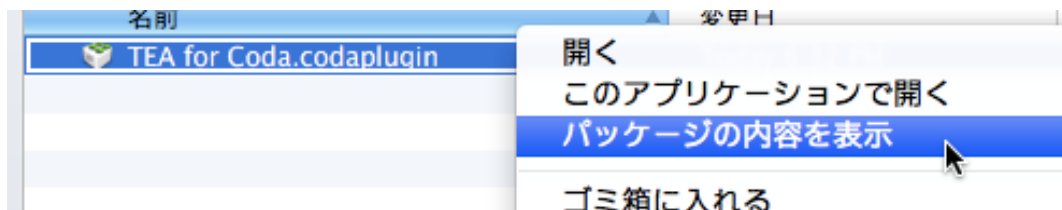
このように、非常に便利なZen Codingですが、標準では一部使いにくい部分があります。例えば、HTML5の基本タグを展開した場合、「lang」属性が「en-US」となっていて日本語を扱うWebサイトでは扱いにくくなっています。

その他、いつも使うライブラリをあらかじめ埋め込んでおくなど、カスタマイズをすることができます。次の手順で変更してみましょう。

Finderを表示し、「移動」メニューを開きます。「ライブラリ」メニューがあれば、これをクリックします。メニューに表示されていない場合は「Option」キーを押すと見えるようになります。



「Application Support→Coda 2→Plug-ins」のようにフォルダを辿ると、プラグインのファイルが見つかります。このアイコンを右クリック（または Control+クリック）し「パッケージの内容を表示」メニューをクリックします。



すると、実はこのプラグインはフォルダ構成になっており、さらに中にファイルなどが取まっています。「Support→Library→zencoding」とフォルダを辿り、「zen\_setting.py」というファイルをエディタソフトで開きます。Codaで開いても良いでしょう。

11行目付近から始まる、次の行を日本語向けに書き換えます。

```

7  # 変数名が予約されているため、このように使用
0      'variables': {
1          'lang': 'en',
2          'locale': 'en-US',
3          'charset': 'UTF-8',
4          'profile': 'xhtml',
5
6  # Inner element indentation
7      'indentation': '\\t'

```

書き換えると、次のようになります。

```

0  variables: {
1  'lang': 'ja',
2  'locale': 'ja-JP ',
3  'charset': 'UTF-8',
4  'profile': 'xhtml',
5
6  # Inner element indentation

```

また、出力されるHTMLファイルは、同じファイル内に記述されているので、ショートコードで検索をしてみると良いでしょう。例えば「html:5」で生成されるコードは、561行目付近から始まります。

```

560
561 'html:5': '<!DOCTYPE HTML>\n' +
562 '<html lang="{locale}">\n' +
563 '<head>\n' +
564 '  <meta charset="{charset}">\n' +
565 '  <title></title>\n' +
566 '</head>\n' +
567 '<body>\n\t${child}|\n</body>\n' +
568 '</html>'
569 },

```

これを書き換えると、生成されるHTMLが変わります。こうしてファイルを保存し、Codaを再起動しましょう。

### 【コラム】 Emmetを利用する

Zen Codingは 2012年現在、機能を向上し、名前を一新して「Emmet」というプロジェクト名で開発が進められています。

Emmetでは、Zen Codingの機能はそのままに、機能の追加や置き換えなどが行われています。例えば、Zen Codingでは次のようなHTMLを生成する場合、

```

<div id="main">
  <p></p>
</div>
<div id="sub">

```

次のようなコードを記述する必要がありました。

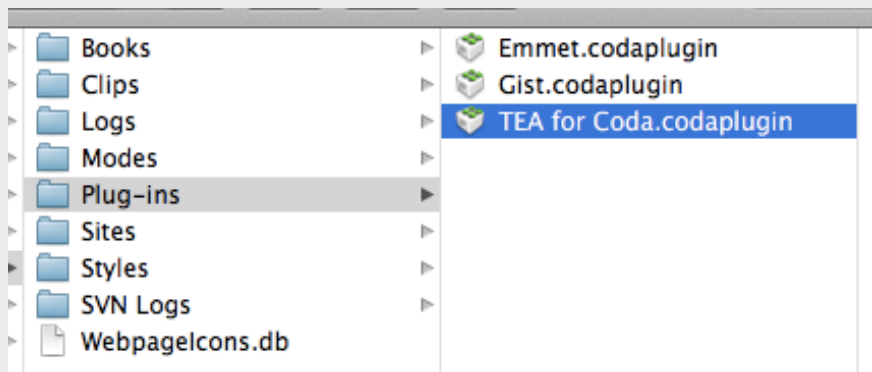
```
(div#main>p)+div#sub
```

しかし、Emmetの場合は新しく「^」という記号で、一つ上のレベルに戻ることができるようになったため、次のように記述することができます。

```
div#main>p^+div#sub
```

Codaもすでに、Emmetに対応したプラグインがリリースされており、これをインストールすることで利用できます。次の手順でインストールしましょう。

- はじめに、「TEA for Coda」とショートカットキーなどが重複しているため、先に削除しておきます。ユーザーディレクトリの「ライブラリ→Application Support→Coda 2→Plug-ins」と移動して、「TEA for Coda.codaplugin」ファイルを削除します。



- 次のサイトから、Coda用のプラグインである「Emmet for Coda」というファイルをダウンロードして、ZIPファイルを解凍します。

<https://github.com/emmetio/emmet-objc/downloads>

- 拡張子が「.codaplugin」のファイルをダブルクリックしてインストールします

これで「プラグイン」メニューに「Emmet」が追加され、利用できるようになります。上記のコードなどを記述して「Control+E」のショートカットで展開してみてください。

## LESS用プラグインを追加する

Coda 2は、標準で Sassには対応していますが LESSは対応していません。利用する場合には、上記と同じ方法で次のプラグインをインストールします。

LESS CSS Plugin for Coda

<http://incident57.com/coda/index2.php>

## サイト定義を使おう

FTP接続をしてファイルを管理するなら、「サイト定義」をした方がなにかと便利になります。FTPの接続先を保存しておくことができ、さらに簡単な操作でファイルをアップロードしたり、ダウンロードして最新版に保つことも可能になります。

サイトは、初回起動時に表示されているほか、ツールバーの左端の「サイト」ボタンをクリックすることで移動することができます。





ここで、画面右下の「+」ボタンをクリックするか、「Shift+⌘+A」のショートカットで新規サイトウィンドウが表示されます。

サイト名: 名称未設定サイト

プロトコル: SFTP  パッシブモード

サーバアドレス: 入力例: domain.com 22

ユーザ名:

パスワード:   その都度尋ねる

トップページ: 入力例: http://www.domain.com/

ローカル URL: 入力例: http://localhost:3000/

リモートルート: /www/domain/ (オプションです)

ローカルルート: ~/Sites/domain/ (オプションです)

次の各項目を設定していきましょう。

#### \* サイト名

サイト一覧を見たときに、見分けがつくような名前にします。

#### \* プロトコル・サーバアドレス・ユーザ名・パスワード

それぞれ、FTPの情報を設定していきます。パスワードは保存させずに「その都度尋ねる」にチェックを入れても良いでしょう。

### \* トップページ

ここでは、FTPでアップロードした時のWebサイトのURLを入力します。例えば、

`http://h2o-space.com`

などと入力すれば、後でプレビューなどに利用できるようになります。入力しなくても構いません。

### \* ローカルURL

MAMP（後述）などを用いて、ローカルサーバーを利用している場合にここにそのアドレスを入力することで、プレビューなどで利用することができます。

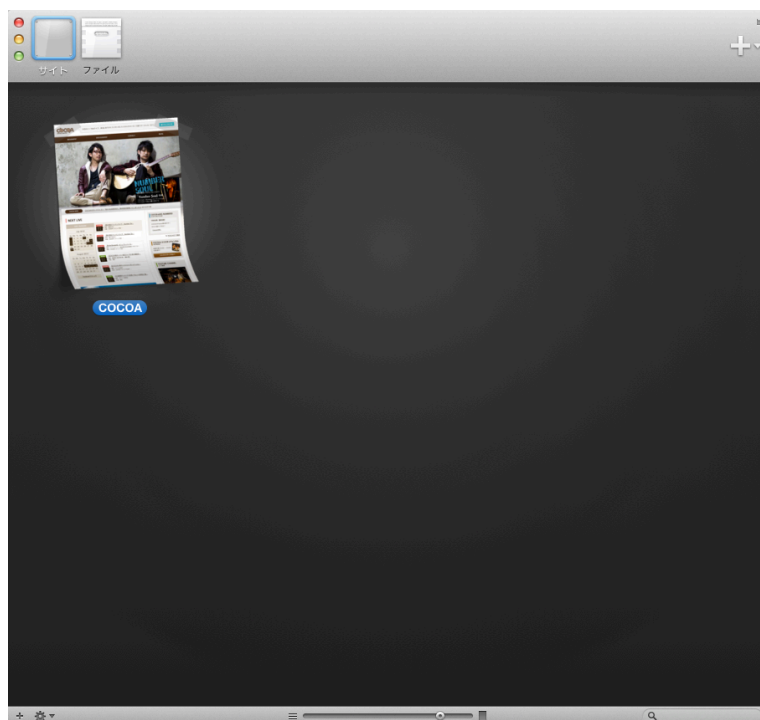
### \* リモートルート

FTPで接続をしたときに最初に表示されるディレクトリを入力します。「ドキュメントルート」や「ウェブルート」などとも呼ばれます。例えば、レンタルサーバー「Heteml」の場合は「web/」がリモートルートになります。

### \* ローカルルート

開発しているWebサイトのファイル群を保存しているディレクトリを指定します。ファイルパネルの初期位置として指定することができます。

以上の設定を行うと、自動的にトップページのキャプチャが撮影されて、アイコンとして利用されます。



そのWebサイトに BASIC認証を設定している場合などは認証ウィンドウが出てくるので、IDとパスワードを記入してログインしましょう。

画面下のスライダーを利用して、大きさを調整することができます。また、複数作成した場合に順番がばらばらになってしまったら、空いているエリアを右クリック（Control+クリック）して「整頓」メニューで整頓し直すこともできます。



また、「グループ」を作ることができます。まとめたアイコンをドラッグドロップで他のまとめたアイコンに重ねると、以下の図のようにグループができあがります。



### 【ワンポイント】FTPに接続できない場合

定義したサイトを開くときに、次のようなエラーメッセージが表示されることがあります。



この場合、次のような点をチェックして行くと良いでしょう。

- 各種情報の書き間違い

最も多いのは、サーバーアドレスやID、パスワードの書き間違いです。改めて定義書類などを確認しながら、正しく設定しましょう。

- ・プロトコルの違い

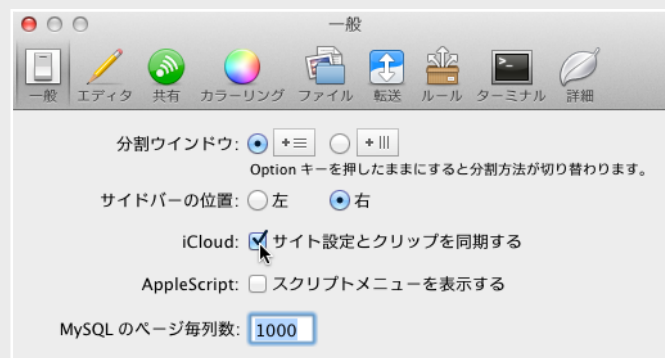
FTPには大きく分けて、「FTP」「FTPS」「SFTP」などの種類があります。改めて確認しましょう。

- ・パッシブモードのチェックを変更してみる

FTPには接続の方法に「パッシブモード」があります。標準でこのチェックボックスにはチェックが付いた状態になっていますが、これを外すことで接続できる場合があります。試してみると良いでしょう。

## 【ワンポイント】 iCloudでのサイト定義の同期

AppStore版でCodaを購入をすると、iCloudでの同期機能がサポートされます。「環境設定→一般」から「iCloud」のチェックをつけると、サイト定義（とクリップ（後述））が iCloudにアップロードされ、別の環境から同じアカウントでログインしている Codaに同期されます。



デスクトップパソコンとノートパソコンなど、複数の環境で利用している場合に便利でしょう。

## 【ワンポイント】 複数のサイトを同時に編集する

Codaは、1つのウィンドウでは1つのサイトしか編集することができません。もし、複数のサイトを同時に編集する場合は「ファイル→新規ウィンドウ」メニュー（ショートカットはShift+Control+N）で新しいウィンドウを開き、別のサイトを開くと良いでしょう。

また、サイトのアイコンを右クリックすると「新規ウィンドウで開く」というメニューがあり、これでも同じように新しいウィンドウで開くことができます。



### 【ワンポイント】 Transmitからの「お気に入り」の取り込み

Codaを開発しているPanic社は、FTPのクライアントソフトである「Transmit」という製品も販売しており、利用している方も多いでしょう。

Codaには、このTransmitの「お気に入り」をサイトの情報として取り込む機能があります。「ファイル→サイト→Transmit お気に入りを読み込む」メニューで、取り込みたい接続情報を選びます。

また逆に、Transmitにも「お気に入り→読み込み→Coda」メニューからサイト情報を取り込むメニューが準備されており、こうして両製品を使い分けて1つのサイトをメンテナンスすることができます。

### 【コラム】 Codaと Transmitの使い分け

Codaを利用していると、Transmitを利用する機会はあまりないかもしれません。特に、Coda 2からはFTPのエンジンが強化され、前バージョンでは容量の大きなファイルの取得に失敗したりしていた現象が少なくなったように感じます。

TransmitにあってCodaにない機能としては、次のようなものがあります。

ファイルの同期

サーバー同士でのファイルのやりとり

サーバー上のフォルダを通常のフォルダに用に扱える「Transmitディスク」

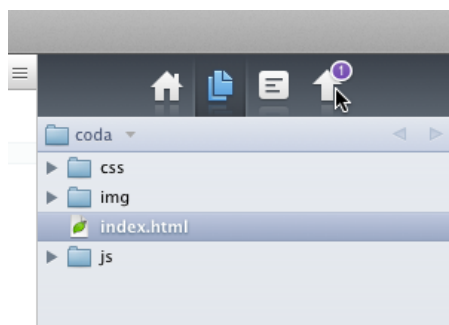
## タブを利用した複数のサーバーへの同時接続

このようにネットワークフォルダを高度に扱いたい場合に、Transmitは威力を発揮します。魅力を感じたら、Codaと一緒に利用すると良いでしょう。

### 公開機能を使いこなそう

サイト定義を利用する最大のメリットは「同期機能」が利用できるようになることです。サイト定義で「リモートルート」と「ローカルルート」を正しく設定すると、Codaは両フォルダを監視し、ファイルの数や更新日時に差異がないかを確認するようになります。

そして、ローカルルート上のファイルを変更するとサイドバーの「公開」ボタンに、図のように数字バッジが表示されるようになります。

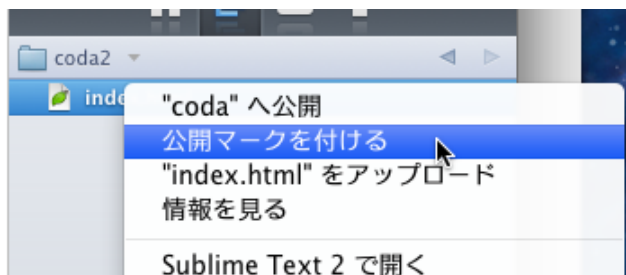


これをクリックすると、図のように変更されたファイルが一覧され、下部のボタンから一括でアップロードをすることができます。



こうして、編集から公開までの流れをスムーズに制御することができます。

なお、もし他のエディタソフトなどでファイルを変更した場合で、公開対象のファイルにならなかった場合、公開したいファイルを右クリック（Control+クリック）して「公開マークをつける」をクリックすることで、リストに加えることができます。



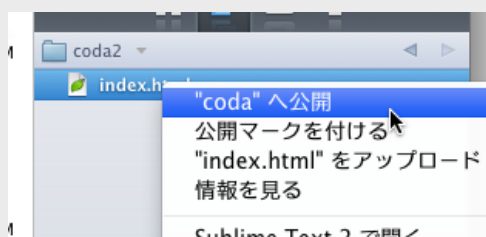
### 【ワンポイント】監視はローカルルートのみ

Codaが監視するのはローカルルートの変更のみです。そのため、他の人がサーバー上のファイルを変更している場合などに、ローカルルートの方が「古い」ということは通知されませんので注意しましょう。

そのまま「公開」をすると、リモートルート上のファイルが上書きされてしまいます。複数人で管理しているWebサイトの場合は、作業を始める前に必ず最新ファイルをダウンロードしてから進めるようにしましょう。

### 【ワンポイント】「公開」と「アップロード」の違い

一括公開の他に、1つのファイルを選択しての公開も行うことができます。公開したいファイルを右クリックすると「【サイト名】へ公開」メニューが表示されます。これをクリックすると、リモートルートにアップロードされます。



しかしここで、同じメニューに「【ファイル名】をアップロード」という機能もあります。この二つの機能の違いはなんでしょうか。それは、「フォルダ構成を意識するか」にあります。

「アップロード」は、リモート機能で現在表示されているフォルダにアップロードされます。つまり、そのファイルが「/index.html」というファイルなのか「/dir1/dir2/index.html」とフォルダの中のファイルなのかは無視され、リモート側で今表示されているフォルダにアップロードされます。



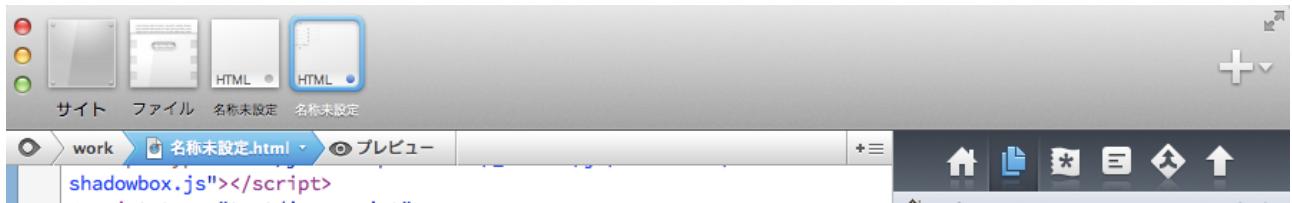
逆に、「公開」ではフォルダ構成が保持されます。つまり「dir1/dir2/index.html」というファイル公開した場合はリモート上に「dir1/dir2」というフォルダが作成され、ファイルがアップロードされます。

基本的にサイト定義をした後で「アップロード」の機能を利用することはあまりありません。

## プレビュー機能

Codaには、ブラウザ機能が内蔵されていて、作成しているファイルをその場で簡単にプレビューすることができます。

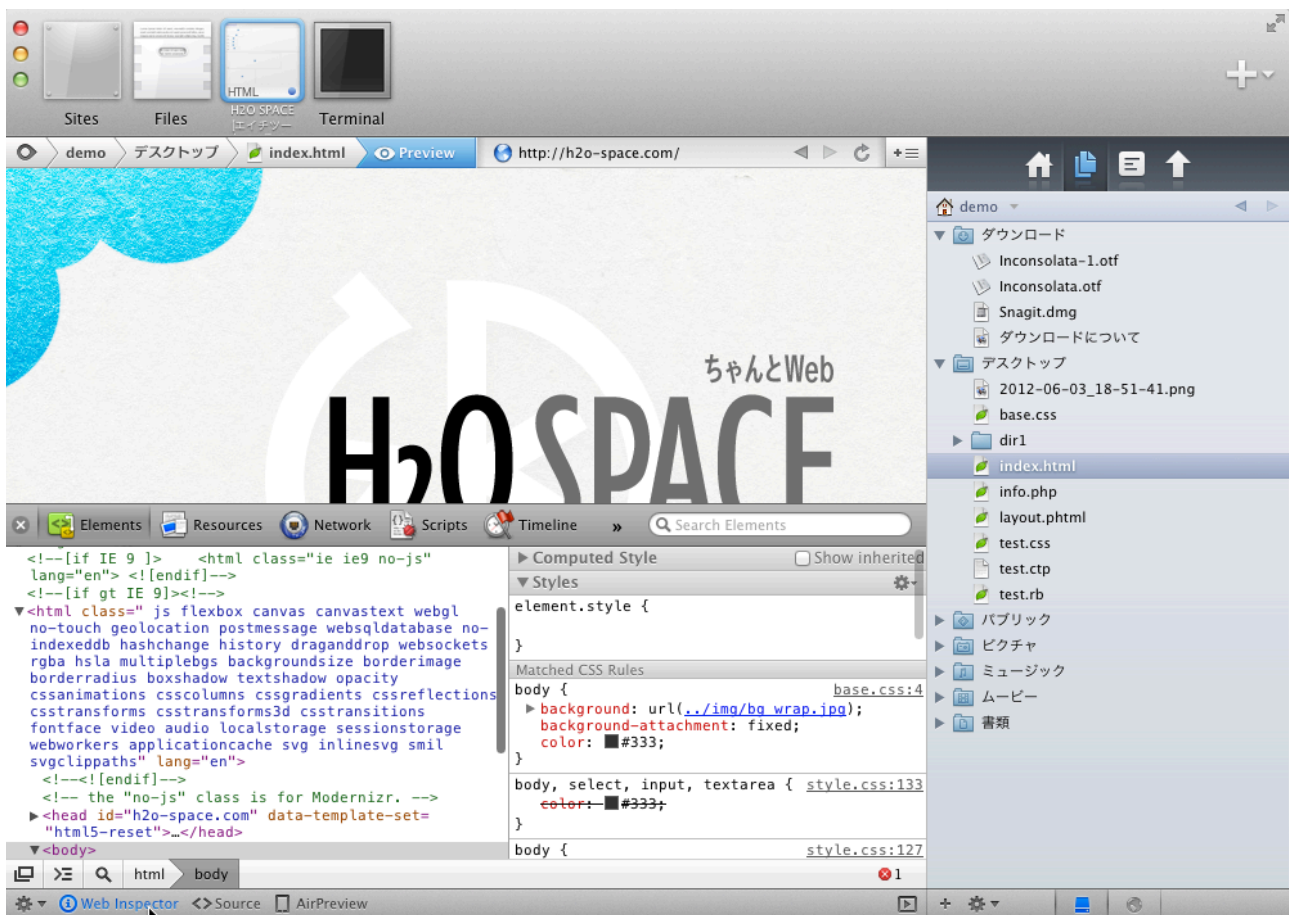
エディタでファイルを編集していると、常に上部のパスバーに「Preview」ボタンが表示されていて、これをクリックすることでブラウザに切り替えることができます。キーボードショートカットの「⌘+4」でも切り替えることができます。



プレビューが終わったら、隣のファイル名をクリックするか、「⌘+3」でエディタに戻ります。ここで表示されるWebブラウザは、Safariとほぼ同等の機能を持っており、WebKit系のWebブラウザの確認ならここで済ませることができます。JavaScriptも正常に動作します。

### \* Webインスペクタ・ソースを表示する

画面下部の各ボタンをクリックすると、要素の解析や、JavaScriptのデバッグなどに利用する「Webインスペクタ」や、生成されたHTMLを閲覧する「ソース」など表示することもできます。PHPなどの開発で、生成されたHTMLを確認したいときなどに利用するとよいでしょう。



### \* ユーザーエージェントを変更する

Webブラウザごとの挙動を確認する場合は、画面右下の歯車ボタンをクリックし、「ユーザーエージェント」メニューから各ブラウザを選択すれば、変更することができます。iPhoneなどのエージェント情報もありますが、バージョンの新しいブラウザはリストになく、現状このリストを変更する方法はないようなので、Codaのバージョンアップを待つ必要があるようです。

### \* 外部ブラウザでプレビューする

FirefoxなどのWebKit系以外のWebブラウザでプレビューしたい場合や、Google Chromeなどで詳細に分析をしたい場合は、外部ブラウザと連携します。画面左下の歯車ボタンの「このアプリケーションで開く」メニューから、ブラウザを選んで起動しましょう。

一番先頭に表示されているWebブラウザは、キーボードの「⌘+Option+B」でも起動することができます。このブラウザは、OS Xの設定で「デフォルトWebブラウザ」に設定されているWebブラウザが採用されます。

変更したい場合は、Safariの環境設定から変更しましょう。

### \* 任意のURLをプレビューする

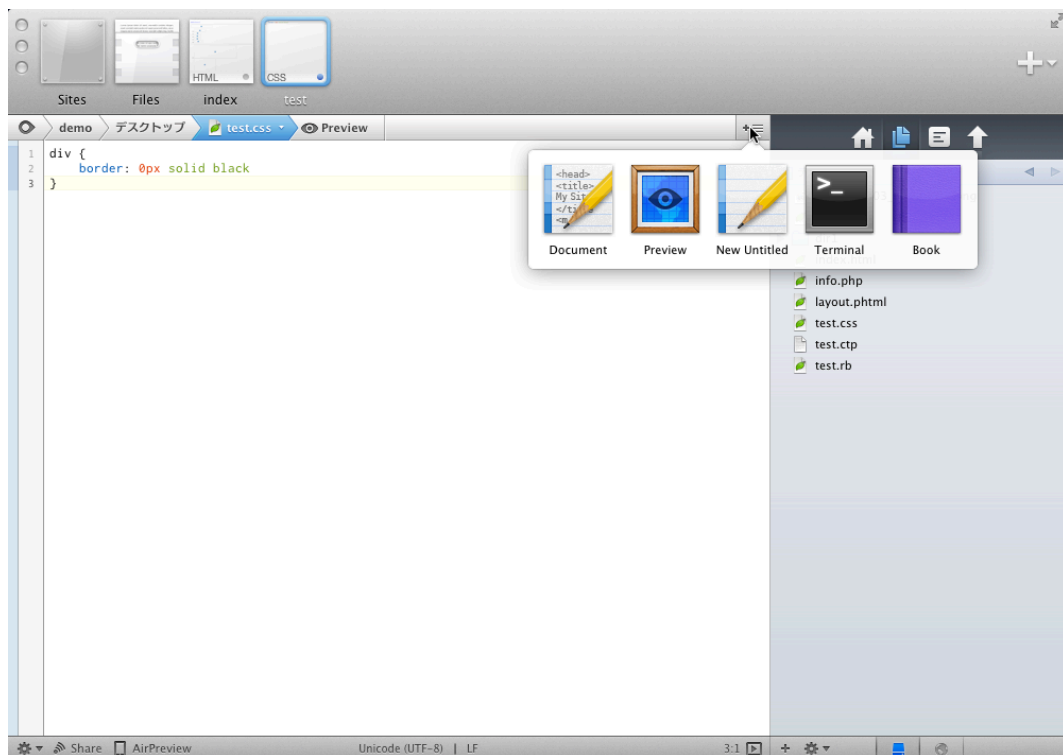
Codaのプレビューウィンドウは、Webブラウザとして完全に機能するため、編集中のファイル以外のURLもプレビューすることができます。

図のアドレスバーをクリックし、URLを変更してEnterキーを押せば、そのURLに移動することができます。PHPのフレームワークなどを利用している場合や、WordPressなどのテーマファイルを編集している場合などで、編集しているファイルと実際のパスが異なる場合に、利用するとよいでしょう。



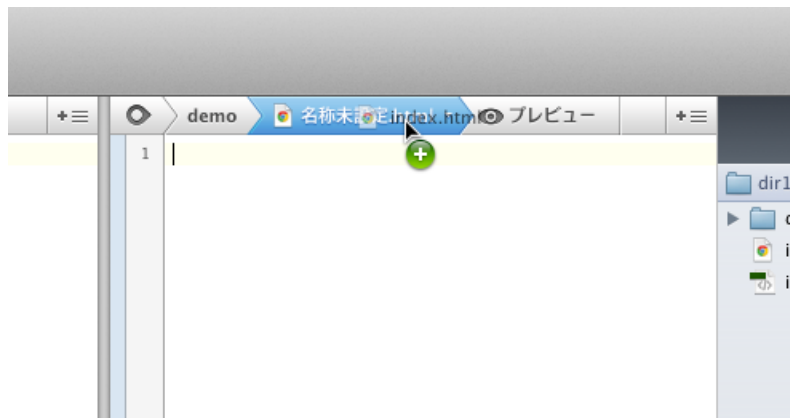
## ウィンドウの分割

Codaは、シングルウィンドウのアプリケーションですがウィンドウを分割することで、複数のファイルを閲覧することなどができます。画面右上の分割ボタンをクリックと、選択肢が表示されるためここから「書類」を選ぶと、同じファイルを閲覧することができます。「⌘+Control+L」を押しても分割できます。



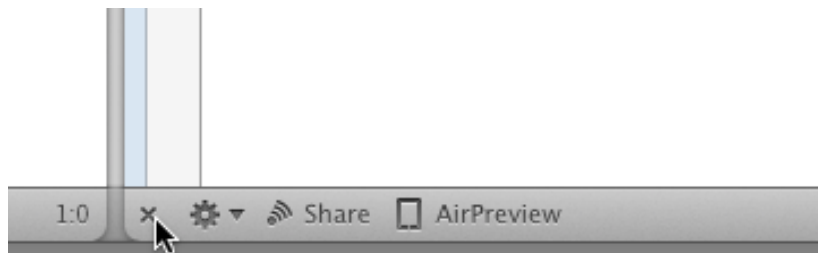
分割したウィンドウは、それぞれ独立してスクロールをしたり、開くファイルを変えたり、プレビューやブック（後述）を表示することなどもできます。

なお、分割したウィンドウにファイルを開くとき、ダブルクリックなどで開くと新しいウィンドウが開いてしまったりするため、一番確実なのは図の部分に開きたいファイルをドラッグドロップすると良いでしょう。



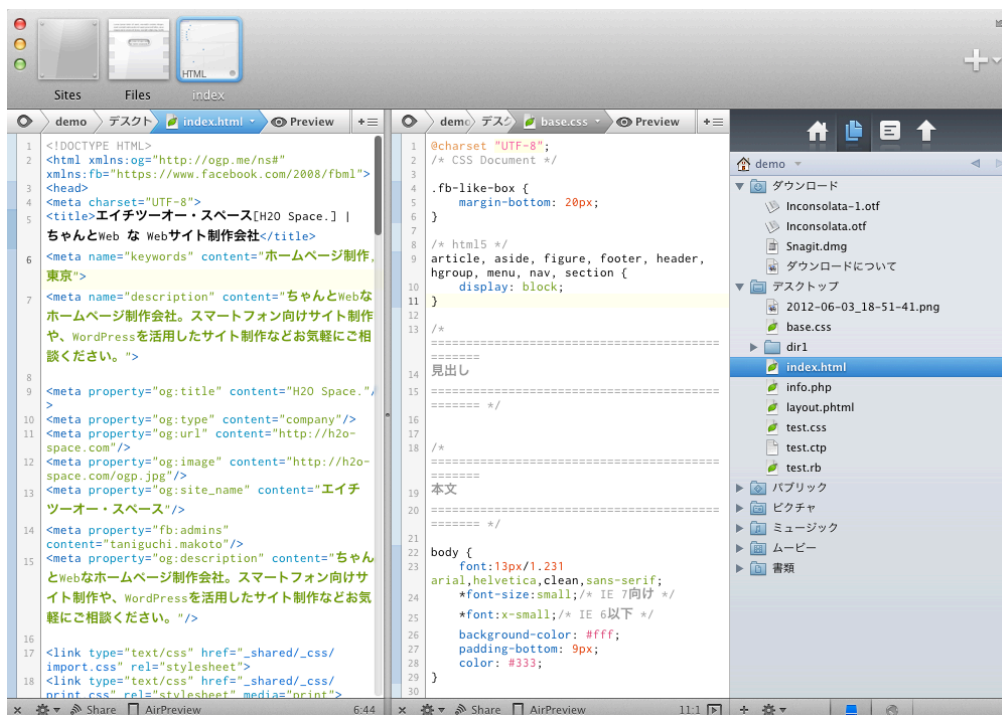
複数分割することもできるので、画面が広い場合は同時にいくつかのファイルを編集することができます。

不要なウィンドウは、図のボタンをクリックするか「⌘+W」で閉じることができます。



### \* 縦分割

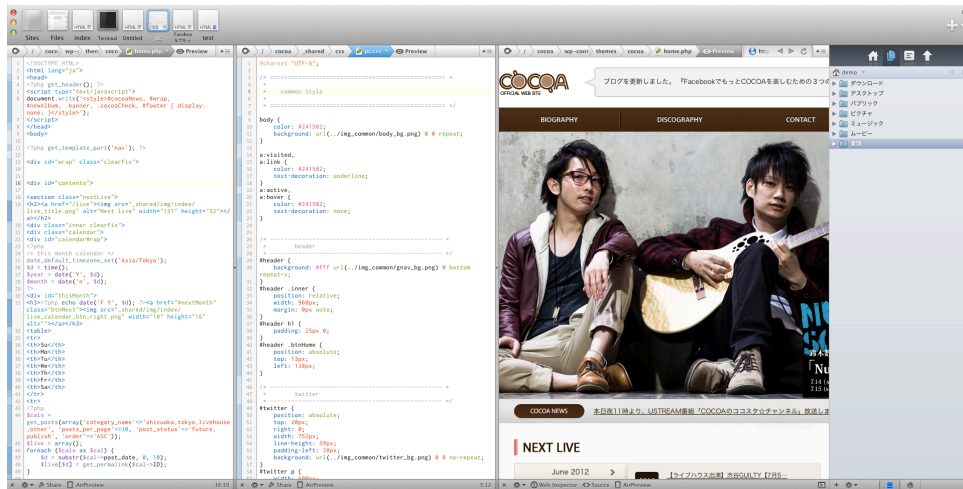
また、分割ウィンドウがない状態で「Option」キーを押しながら、分割ボタンをクリックすると縦方向に分割することもできます。「環境設定」から、縦分割を標準に設定することもできます。



ただし、縦分割と横分割を混在させることはできません。

## プレビューとの組み合わせでライブプレビュー

ウィンドウ分割が力を発揮するのは、プレビューウィンドウと組み合わせたときでしょう。図のように、縦分割をしてプレビューウィンドウを表示するのがおすすめです。



同時に編集しているHTMLとCSSも開いておきます。Codaのプレビューウィンドウは、編集しているファイルと、プレビューウィンドウに表示されているパスが一致している場合、ファイルを保存すると自動的に再読み込みされるため、通常のWebブラウザに表示するよりも、スムーズにプレビューをすることができます。

著者： たにぐち まこと  
発行日： 2013年 1月 12日 初版発行