

コーディング規約 / 02_ディレクトリ構成

- ディレクトリ構成
 - ディレクトリ構成詳細
 - - .htaccess
 - - robot.txt
 - - index.html
 - - 404.html
 - - assetsフォルダ
 - - stylesheetsフォルダ
 - - レイヤーフォルダ
 - - javascriptsフォルダ
 - - imagesフォルダ
 - - wpフォルダ
 - 構成ファイルの命名規則
 - - 使用禁止文字
 - - ページディレクトリの命名規則
 - - HTMLファイルの命名規則
 - - CSS、JavaScriptファイルの命名規則
 - - 画像ファイルの命名規則

ディレクトリ構成

ディレクトリ構成詳細

- .htaccess

WEBサーバーで使用されている「Apache」（アパッチ）の制御を行います。

- robot.txt

検索エンジンロボット（クローラー）の制御を行います。

- index.html

ページを作成する際は、基本としてディレクトリごとに「index.html」を1ファイル設置します。

「index.html」に関連して作成するページは、内容に即した名前を付け設置します。（問い合わせページのconfirm.htmlやthanks.phpなど）

- 404.html

クローラーの回遊性を高める（SEO対策）、UX向上の為に、404ページは必ず設置してください。

- assetsフォルダ

フォルダ名	説明
stylesheets	CSSファイルを格納します
javascripts	JavaScriptファイルを格納します
images	画像ファイルを格納します
movies	動画ファイルを格納します
fonts	フォントファイルを格納します
pdf	PDFファイルを格納します

- stylesheetsフォルダ

FLOCSSをベースにします。CSSファイルをモジュール化（部品化）し、「レイヤーフォルダ」に振り分けることで、スタイルの種類、詳細度、役割を認識しやすくし、メンテナンス性を向上させます。

- style.scss
 - 全てのSCSSファイルをインポートします。Gulpを使用する際は、全てのスタイルがコンパイルされます。
- モジュールファイル
 - ファイルの先頭に「_（アンダースコア）」を追加します。HTML内でこのファイルを直接読み込むことはありません。
- インポートファイル
 - 「モジュール」を「@import」で読み込みます。読み込ませる際は、詳細度の弱い順に読み込ませます。

```
//例:style.scss
@import "foundation/_reset";
@import "foundation/_base";
```

- レイヤーフォルダ

「モジュールファイル」には種類、詳細度、役割などによって「レイヤーフォルダ」に振り分けられます。
「レイヤーフォルダ」には、必ずそのフォルダに格納されている「モジュールファイル」をまとめるための「インポートファイル」を設置します。

レイヤー名	用途	モジュール例
_animates	アニメーションに関するスタイルを指定	_fade.scss、_slide.scss
_foundation	リセットCSS、要素セクタ・属性セクタのような最低限のスタイルを指定	_reset.scss、_base.scss
_layouts	大きな単位のコンテナブロックのスタイルを指定	_header.scss、_main.scss
_components	複数のページで再利用できるような小さな単位のモジュールのスタイルを指定	_button.scss、_title.scss
_project	プロジェクトで固有のスタイルを指定	_about.scss、

		_contact.scss
_utilities	汎用的に使える単一のスタイルを指定	_font.scss、_clearfix.scss

※_compornents、_page、_utilitiesは_objectsフォルダに内包される

- javascriptsフォルダ

各ページにのみ使用するJavaScriptは、使用するページ名を使用してください。

外部のプラグインのファイル名やフォルダ名を変更することは禁止します。また、外部プラグイン等でCSSファイルがセットになっているもの（※1）は、そのままjavascriptsフォルダに格納してください。

```
javascripts
├─ app.js (エントリーポイント)
├─ [ modules ]
│   └─ ページ名.js
└─ [ lib ]
    └─ プラグイン名.js (jqueryなど)
```

- imagesフォルダ

各ページにのみ使用する画像（※2）はページ名でフォルダを作成し、その中に格納します。

```
images
├─ 画像ファイル
├─ 画像ファイル
├─ [ _ページ名 ] ── 画像ファイル *2
│                  └─ 画像ファイル *2
└─ ⋮
```

- wpフォルダ

WordPressファイル一式を格納します。

構成ファイルの命名規則

以下を基本ルールとします。

- 半角英数字のみを使用する
- 必ずアルファベットからはじめ、数字から始めてはいけない
- 単語が2つ以上続く場合は「-」（ハイフン）で区切る

- 使用禁止文字

日本語	全角英数字	全角・半角スペース	機種依存文字
!	?	%	'
\$	&	=	^
+	*	÷	@

/	\	()
[]	{	}

- ページディレクトリの命名規則

- ページを簡潔に表す、ディレクトリ名にする
- ページタイトルに複数の単語がある際に、ディレクトリ名が長くなりすぎてしまう場合は、単語を1~2個に絞ってディレクトリ名にする
- 同じような形のメージが複数個ある場合は、連番を使用して命名することも可能とする

- HTMLファイルの命名規則

- 原則として「index.html」を使用する
- 同ディレクトリに複数のHTMLファイルを設定する場合は、そのHTMLを簡潔に表すファイル名にする

- CSS、JavaScriptファイルの命名規則

- ファイル内の記述を簡潔に表すものとする
- モジュール化しているファイルはファイル名の先頭に「_（アンダースコア）」を追加する

- 画像ファイルの命名規則

- 画像ファイル名は、種類を区別する部分（以下、種類）、個別の名前の部分（以下、個性）、性質を表す部分（以下、性質）に分類する
- レスポンシブサイトでブレークポイントで画像を切り替える場合、「性質」の後に使用するブレークポイントを区別する部分（以下、ブレークポイント）を記述する
- ファイル名は「種類」「個性」「性質」「ブレークポイント」の順に書き、その際は間に「_（アンダースコア）」を記述し分別する

「種類」

画像の目的が判断できる英単語を利用します。

タイトル	背景	ボタン	ロゴ	アイコン	バナー
ttl	bg	btn	logo	icn	bnr

「個性」

画像を使用するセクション、ブロックが判断できる英単語を利用します。

ヘッダー	フッター	お知らせ セクション	お問い合わせ セクション
hdr	ftr	news	contact

「性質」

頻繁に使われるもの、位置を特定しているものなど、画像の性質を表します。

--	--	--	--

連番	位置	マウスオーバー	多言語
01、02、03、04、05...	top、bottom、left、right	on、off	ja、en


「ブレイクポイント」

使用するディスプレイ幅を表します。

PC	タブレット	SP
_pc	_tb	_sp

//例
[種類]_[個性]_[性質]_[ブレイクポイント]
→ btn_hdr_on_pc

添付ファイル (0)



白藤 卓也によって 2020/10/26 19:15:00 に作成されました。



白藤 卓也によって 2020/12/08 10:35:02 に最終更新されました。