Regulation guidelines

1. **OS environment**

* Microsoft Windows XP / Vista / 7/8 IOS 5.0 or later
* iPhone, iPad, iPod touch (iOS 4.x is also considered depending on the case)
* Macintosh OS X
* Android 2.3 or later (depending on the case also 1.6 to 2.2)

1. **Supported browsers**

**Windows:** Internet Explorer 8 or later, Firefox latest version, Chrome latest version, Opera latest version Opera 11.x or later

**Macintosh:** Safari 5.x or later, Firefox latest version, Chrome latest version

**IOS:** Safari

**Android:** Standard browser

*\* It is not assumed as a standard, but consideration is given to display.*

IE6, Safari 4.x or later, Firefox 3.x or later,

Opera Mini, Opera Mobile

Which version of OS / Browser should be supported,

Be sure to make it clear before production. (Especially IE)

1. **Screen**

Desktop tablet: screen resolution 1024 x 768 px or more

*· Correspond to A4 size notebook*

*· Correspond to B5 size notebook*

Smartphone: screen resolution 320 × 480px or more

· The standard is based on iPhone 4S screen resolution 640 × 960px,

Or, based on iPhone 5's 640px × 1136px,

Corresponds to principle Retina display in principle.

Color depth: 16 bits or more

- Corresponds to visibility of liquid crystal monitor & supports reproducibility in monochrome printer

1. **Directory file design rule**

┣ css / · · · Css storage folder (Performance, import of CSS is not basically done)

┣bootstrap.css · · Read more http://getbootstrap.com/css/

┣ common.css · · CSS reset & default CSS (manage default style for OOCSS and responsive)

┣ style.css · · · Style sheet file common to all pages

┣ xxx. Css · · · Unique style sheet file of each page (eg home.css)

┣ js / · · · JavaScript file storage folder

┣bootstrap.js · · · Read more http://getbootstrap.com/javascript/

┣ script.js · · · Common JS (jQuery) file

┣ img / ... Image storage folder (stores favicon files directly under)

┣ common ... Common image storage folder

┣ home ... Image storage folder of the top page (index.html)

┣ xxx Image storage folder for each page

┣ bnr ... Image storage folder of banner

┣ htc / · · HTC file storage folder

┣ fonts / ... Web font file storage folder

┣ plugin / ... External storage folder

┣ media / ... Rich content storage folder

┣ docs / ... Document storage folder

┣ index.html ... Top page file

┣ company · · · Company profile html file storage folder

┣ index.html · · · Company profile Lower page file

┣ xxx · · · Each page html storage folder

┣ index.html · · · lower layer page file

┣ xxx.html ... Lower page file

┣ content / · · · Store external HTML file etc read by Ajax

1. **HTML**

Document type: HTML5

*\* The selection of the document type depends on the case. For tablet · smartphone optimization, use of HTML 5 is indispensable in future.*

The file extension is ".html" only. ".htm" is not acceptable.

(SSI setting is extended, enabling ".html")

1. **Image directory**

**"/ Img"**

An image file (GIF image or JPEG image) referred to from the HTML file is stored.

The extension of the stored file is ".gif" ".jpg" ".png" ".ico".

1. **CSS directory**

**"/ Css"**

Stores the CSS file referenced from the HTML file.

The extension of the stored file is ".css".

1. **JavaScript directory**

**"/ Js"**

Store JavaScript file referenced from HTML file.

The extension of the stored file is ".js".

1. **HTC（HTML Component）**

**"/ Htc"**

Store the HTC file for using the IE's behavior function.

The extension of the stored file is ".htc". (It also allows PHP files for general purpose)

\* Behaviors are HTC (HTML Component) files composed of HTML / Dynamic HTML / JavaScript etc.

Function to use on HTML. It is possible to reuse / encapsulate functions by using HTC files.

1. **Web fonts**

**'/ Fonts'**

Store a font file to be used for Web fonts and the like.

The extension of the stored file is ".eot" ".svg" ". Ttf" ". Woff" etc.

1. **Rich content**

**"/ Media"**

Movie files and audio files are stored.

The extension of the stored file is various media files such as ".swf" ".mp3".

1. **document**

**"/ Doc"**

Various document files.

The extension of the stored file is ".doc" ". xls" ". pdf" etc.

Make the file name / directory name as short as possible and naming it so that you can imagine the contents of the file as you can understand at a glance.

1. **Usable characters / symbols**

For directory name / file name, you can use half size (1 byte) alphanumeric lowercase letters, numbers, and some symbols only.

Ex: *a b c d e f h i j k l m n o p q r s t u v w x y z 0 1 2 3 4 5 6 7 8 9 - \_ .*

1. **Word** **count**

In order to be able to handle files under various circumstances, it should be 31 characters at a half byte (1 byte).

Ex: *abcdefhijklmnopqrstuvwxyz01234567*

1. **Prohibited matter**

· Spaces (spaces) can not be used with file names.

· Do not use "\_" (underscore) or "-" (hyphen) twice in succession.

(Allow when continuing two attributes or words in file name / directory name.)

***Example:***

*○ sn\_contact\_price.gif*

*× sn\_\_contact - price.gif*

· Basic consists of alphanumeric characters and underscores. Do not use hyphens as the principle file name.

***Example:***

*○ photo\_01.gif*

*× photo-01.gif*

· Prohibit the use of the following file names.

*con. aux. nul. prn.*

*\* Because you can not operate files under Windows environment*

1. **Inscription**

The notation of the directory / file name is determined as follows.

· Japanese · · · Hepburn type romaji notation

· Foreign languages, Western languages ​​· · · Western notation

\* As a general rule, describe the meaning of a file or directory within 3 words and express it as one word as possible.

※ When the file name directory name becomes 2 words or more, do as follows.

***Example***: *Case at home · · · case\_home.html*

\* If the name is based on English, use that English word. If one word is long, use abbreviation or omit some.

※ If the name is based on Japanese, use the Roman alphabet (Hepburn style). Furthermore, if one word is long, use abbreviation,

Some are omitted.

\* If the file name / directory name is based on a combination of katakana or Japanese / English, do not unify it in English notation.

However, this is not required unless only numbers or abbreviations are permitted depending on constraints on the system (such as the maximum number of characters).

1. **Directory file naming convention**

|  |  |
| --- | --- |
| **Project** | **Description** |
| company | 会社、企業 |
| corporate | 法人の、企業の |
| about | ○○とは？ |
| register | 会員登録 |
| contact | お問い合わせ |
| terms | 利用規約 |
| privacy | プライバシーポリシー |
| philosophy | 理念 |
| legal | 特定商取引法に関する表示 |
| recruit | 採用情報 |
| fresh | 新卒採用 |
| career | 中途採用 |
| news | ニュース、最新情報 |
| info | お知らせ |
| history | 沿革、歴史 |
| greeting, message | 挨拶 |
| profile | プロフィール、経歴、略歴 |
| guide | はじめての方、ガイドライン、ご案内 |
| business | 事業内容 |
| feature | 特徴、特長、特色、特性 |
| specialty | 専門 特殊性 特徴 特色 |
| results | 実績、業績 |
| works | 作品 |
| cases | 事例、実例 |
| member | 会員、構成員、メンバー |
| access | 交通、アクセス |
| map | 地図、案内図 |
| faq | FAQ よくある質問 |
| reservation | 予約 |
| policy | 方針 |
| sitemap | サイトマップ |
| introduce | 紹介 |
| flow | 流れ、順序 |
| rule | ルール、規則、規定、規約 |

Reference: http://blog.btmup.com/web-general/contents-directory-file-name-list.html

1. **Image File Naming Convention**

|  |  |
| --- | --- |
| **Project** | **Description** |
| logo | Used in logo images.  *Example: logo.gif logo\_footer.gif* |
| side\_ | Used in side navigation images.  *Example: side\_xxx.gif* |
| icon\_ | Used in icon images.  *Example: icon\_xxx.gif* |
| btn\_ | Used in button images.  *Example: btn\_xxx.gif* |
| bg\_ | Used in background images. |
| line\_ | Used in ruled line images. |
| pht\_ | Used in photo images. |
| pic\_ | Used in illustration images. |
| h\_ | Used in the heading image. |
| bnr\_ | Used in banner images. |
| xxx\_ | Used in content images  *Example: team\_01.jpg* |

Note: As far as possible, write out one by one for each type and display it with CSS Sprite.

1. **Character entity reference**

“>” and “" “that exist in regular text in HTML are described by character entity reference.

\* When using the above symbol with alt = "", depending on the browser, it may be garbled, so do not use it and describe as normal.

1. **Symbol**

The following symbols are displayed without any problems in all models, so you can use them as they are.

1. **Character symbols that require attention to use**

Half angle marks (\) are garbled in backslashes if they are Macintosh. The following entity reference should be used.

*& Yen;*

1. **Characters and symbols that need attention to use**

Although the following letters and symbols have no problem with regard to use on the Internet, there are cases where it is not displayed with the old model.

1. **Model dependent letters and symbols**

When texts including the following letters and symbols are created, different characters are displayed depending on the environment, not only do not display as intended,

Do not use it as it may cause garbled characters.

※ If necessary for layout, correspond with images.

1. **Character code / line feed code**

The character code ( "character encoding format" or "encoding") of the HTML file is always specified by the META element (<meta> tag).

Since the line feed code of HTML file, CSS file, and JavaScript file is not broken at the time of file editing directly on the server

1. **Unicode normalization form**

Because Unicode covers all languages, there are multiple identical characters.

As a measure for considering the arrangement and functions of these characters the same, a means called normalization is used.

Using Unicode normalization makes it easier to compare texts and is useful for searching and aligning.

It can be selected only when the character code of the document is UTF - 8.

Of the initial configuration **C** that you leave (applying the standard synthesis after the standard decomposition).

1. **Unicode signature (BOM)**

BOM is an abbreviation of Byte Order Mark, which specifies endianness in the 16-bit Unicode encoding method.

In UTF-8, it is possible to determine that there is a BOM if it is Unicode, so even if a document written in UTF-8 is at the beginning

There are times when you can turn on BOM.

Since the basic to use not include the BOM, remove the check of the **Include Unicode** **Signature.**

1. **Image basic specification**

Please refer to the CI manual etc. for the color to be used, confirm the basic color and the specified color.

Only GIF file, JPEG file, PNG file are allowed for image file.

**■ PNG**

PNG-8, PNG-24, and PNG-32.

PNG - 24 and PNG - 32 correspond to full color. Lossless compression formula. The extension is ".png". It is impossible to create animation with a single file.

PNG-8 exports up to 256 colors of 8-bit color. (※ IE6 even transmission can be specified) When an icon or logo began to write under the same conditions,

Compression ratio is often higher than GIF.

PNG-24 is exported in 24-bit color (full color). Transparency can not be specified.

PNG-32 is exported in 24-bit color (full color). Furthermore, alpha channel information (8 bits) is added.

You can specify transparency, but the capacity of the image is large. Transparency from IE 7 when not using JS etc.

※ Photoshop's "PNG - 24" notation is handled as "real PNG - 32" if you put a check in the transparent part.

**■** **GIF**

Image format that can be used for general purpose. The extension is ".gif". Used for transmission and animation.

Since it can not use PNG when responding to old-style Garakae etc, use GIF.

When exporting in Photoshop, set the color setting to "specify".

When exporting in Fireworks, set the color setting to "All colors assigned".

**■** **JPEG**

In principle, photographic images are treated as JPEG images. The extension is ".jpg".

When writing out, while reducing deterioration to the minimum, reduce the file size.

**■** **SVG**

SVG stands for Scalable Vector Graphics.. The extension is ".svg".

Advantages of using SVG over other image formats (like JPEG and GIF).

1. **JavaScript basic specifications of the**

Most of them can be implemented by jQuery.

The easiest thing is to use basic jQuery.

Version: JavaScript 1.6 jQuery 1.12.4

In principle, JavaScript is managed with an external file. It is desirable to read immediately before </ body>.

1. **External JavaScript file list**

|  |  |  |
| --- | --- | --- |
| **File name** | **Contents** | **Storage folder** |
| common.js | Back to the top of the page Smooth scrolling etc. All pages common JQ | / js / |

1. **Naming rules**

Be sure to declare a variable name followed by var.

In principle, use related English or abbreviated words and use lowercase letters.

When naming, when the related word becomes two or more words, the first letters after the second word are described in uppercase letters (Camel Case).

***Description example***

*Var countTime;*

1. **JavaScript comments**

Describe the script for which action is described by comment.

Example:

*// Var countTime;*

1. **reading of jQuery**

Use CDN. Many sites use Google. Basically use compressed lightweight min file.

\* The method of referring to the latest version of jQuery 1 series is deprecated. \* If there is something wrong with the latest version, it will be affected directly.

***Description example***

*<script src = "//ajax.googleapis.com/ajax/libs/jquery/#version#/jquery.min.js"> </ script>*

1. **The contents of script.js**

Since script.js describes a general-purpose standard script, refer to it. Delete unnecessary functions and add new.

1. **Effect of display speed**

Web site performance affects conversions and SEO, so consider carefully about display speed.

Designers and markup engineers devise the front-end parts of HTML, CSS, images, JavaScript, etc.,

It is possible to greatly improve the display speed.

1. **Mark-up that complies with the W3C**

• Understand the document structure of HTML and make accurate markup

• Simple and logical source code

1. **Reduce the number of HTTP requests**

Display speed can be improved by reducing requests for images and external files

1. **External file**

When CSS and JavaScript are described inline, when an HTML document is requested

Since it will be downloaded every time, the basic is treated as an external file.

Reading CSS does not use @import.

Also when writing in the head element, read in the order of external CSS → external JS.

(If \* inline description is required, describe it after reading the external file)

1. **Exporting images**

• Export with actual size (this is not the case when responsive design or Retina display is compatible)

• Images are minimized to the extent that they do not deteriorate.

Compression tool

*- http://www.jpegmini.com/*

*- http://www.punypng.com/*

- http: // www.gigafree.net/tool/graphiccomp/caesium.html

• Use JPEG, GIF, PNG as appropriate. (PNG 8 is often higher compression ratio than GIF)

"Smush.it"

Http://www.smushit.com/ysmush.it/

Although 8-bit-PNG image files also contain metadata called chunks,

Not everything is necessary for display. By using the tool "Smush.it"

It is possible to generate an image in which unnecessary portions are omitted.

• For background images etc., instead of using a single large image, use small size images repeatedly.

• Prefetch (preload) the image as necessary. \* Prevent flickering of images by rollover etc. with JavaScript.

1. **Description of CSS**

• Decrease descendant selectors as much as possible for selector specification

※ adoption of **OOCSS (Object-Oriented CSS)**

Object-oriented way of thinking to specify more than one class to another purpose, coded in **OOCSS (Object-Oriented CSS).**

In addition to becoming a countermeasure against the reading speed of the page, if it consists of reusable modules

The number of man-hours required for coding can be reduced.

• Use the image minimized to the extent that it does not deteriorate

• If effects such as rounded corners and gradations are available, use CSS 3 instead of images

• Avoid double designation as much as possible

• Avoid universal selector ( '\*') because it slows rendering

• If you are coding for PC · smartphone site, color specification is omitted and description (# FFFFFF → # FFF)

\* Since it may not be recognized in the Gala site, it is described properly in 6 digits

• For fixed layout, use table-layout: fixed for table tag

1. **Compression and weight of the CSS**

■ CSS Optimize (CSS optimization tool)

*Http://css.webcreativepark.net/csstidy/*

(CSS description example)

*Margin: 0 px → margin: 0*

*Color: # FFFFFF → color: # FFF*

*Font: 1.0em → font: 1em*

*Padding: 5px 10px 5px 10px -> margin: 5px 10px*

1. **CSS Sprite**

Write out the image as one same image and display only the necessary parts with CSS.

I often use it for navigation etc.

Since it is only necessary to read one image, HTTP request can be reduced.

Use it positively if possible.