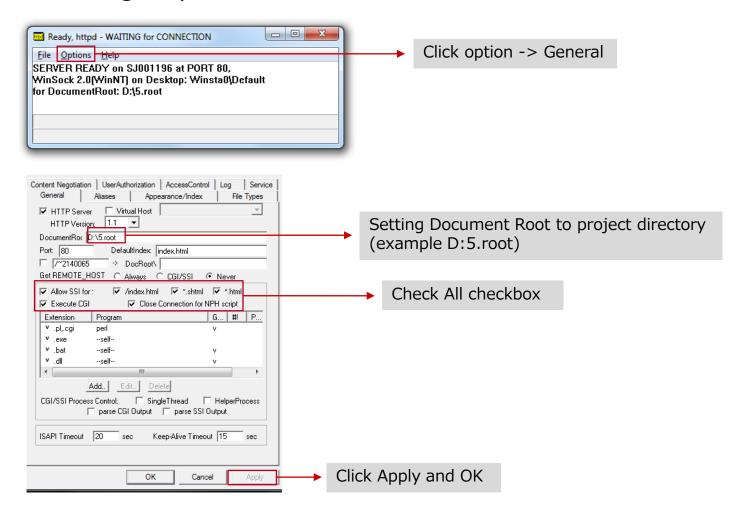# Coding Guideline

INDEX

1. Preparation

Httpd

Httpd is a software to run website into local server, this software very usefull when we working on root path directory, We can place multiple-projects by setting VirtualHost with this tool
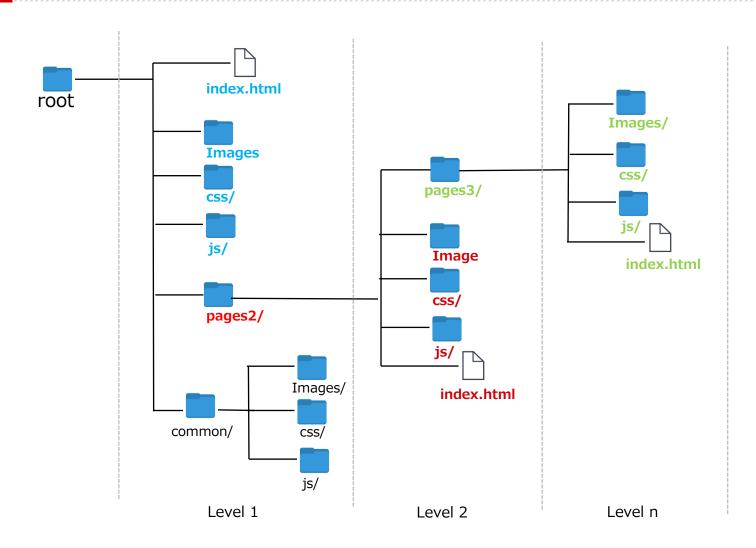
1. Preparation

1.Setting Httpd

Ready, httpd - WAITING for CONNECTION

File Options Help

SERVER READY on SJ001196 at PORT 80,
WinSock 2.0(WinNT) on Desktop: Winsta0\Default
for DocumentRoot: D:\5.root

Click option -> General

Content Negotiation | UserAuthorization | AccessControl | Log | Service
General | Aliases | Appearance/Index | File Types

☑ HTTP Server   ☐ Virtual Host
    HTTP Version: 1.1
DocumentRoot D:\5.root
Port: 80   DefaultIndex: index.html
☐ /~2140065   -> DocRoot\
Get REMOTE_HOST   ○ Always   ○ CGI/SSI   ● Never
☑ Allow SSI for :   ☑ /index.html   ☑ *.shtml   ☑ *.html
☑ Execute CGI   ☑ Close Connection for NPH script

Extension	Program	G...	#!	P...
.pl,.cgi	perl	v		
.exe	--self--			
.bat	--self--	v		
.dll	--self--	v		

Add.. Edit.. Delete

CGI/SSI Process Control: ☐ SingleThread   ☐ HelperProcess
                ☐ parse CGI Output   ☐ parse SSI Output

ISAPI Timeout 20 sec   Keep-Alive Timeout 15 sec

OK   Cancel   Apply

Setting Document Root to project directory (example D:5.root)

Check All checkbox

Click Apply and OK

1. Preparation

2.Running Httpd

Insert "localhost/[projectfoldername]" into browser and press enter
* Please always Use Httpd to running the HTML files

Hierarchy Folder

1. Preparation

Notes:

"Common/" directory resource like Images, CSS and JS Can used in many pages of sites

Storage the images,css and JS file that are used for only page to individual folder are set in the same hierarchy of html

*this structure maybe change based on project requirement

2. Coding General

TCID HTML/CSS Style Guide

2. Coding（Background and General style rules）

■ Background

This document defines TCID's formatting and style rules for HTML and CSS.
The case with undefined rule only.
Comply the existing rule or source code in the project.

■ General Style Rules

-Protocol

Omit the protocol portion (http:, https:) from URLs pointing to images and other media files, style sheets.
Don't use absolute path of external server because of influence when the external sever get down. Use root path basically.This comes in the case which needs absolute path.

<!-- Recommended -->

```
<script
src="//www.google.com/js/gw
eb/analytics/autotrack.js"></s
cript>
```

<!-- Not recommended -->

```
<script
src="http://www.google.com/j
s/gweb/analytics/autotrack.js">
</script>
```

■ General Formatting Rules

-Indentation

Indent by 2 spaces at a time.

Don't use tabs or mix tabs and spaces for indentation.

<!-- Recommended -->

<!-- Not recommended -->

```
<ul>
  <li>Fantastic
  <li>Great
</ul>
```
Use 2 spaces

```
<ul>
    <li>Fantastic
    <li>Great
</ul>
```
Use 1 tabs

2. Coding（Capitalization）

-Capitalization (Upper/Lowercase)

All code has to be lowercase: This applies to HTML element names, attributes, attribute values, CSS selectors, properties, and property values (with the exception of strings).

<!-- Recommended -->

<!-- Not recommended -->

<img src="google.png" **alt**="google">

<IMG src="google.png" **ALT**="google">

-Whitespace at the end of text

Remove white spaces at the end of text.
White spaces are unnecessary and can complicate diffs.

<!-- Recommended -->

<!-- Not recommended -->

<p>lorem ipsum</p>

<p>lorem ipsum </p>

White space are unnecessary

2. Coding（General Meta Rules）

■ General Meta Rules

-Encoding

If clients specify the encoding, use the specified encoding type.
If there is no specified encoding from clients, recommend UTF-8.

<!-- Recommended -->

```
<meta charset="utf-8">
```

<!-- Not recommended -->

```
<meta charset="ANSI">
```

-Comments

Explain code as needed by using comments. *Just as needed, you don't need to explain all codes.

2. Coding（Doctype ）

■ HTML Style Rules

-Document Type

If clients specify the document type, use that document type.

If there is no specified document type from clients, **recommend HTML5.**

<!-- Recommended -->

```
<!DOCTYPE html>
```

<!-- Not recommended -->

```
<!DOCTYPE html PUBLIC "-//W3C//DTD
XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xht
ml1-strict.dtd"><html
xmlns="http://www.w3.org/1999/xhtml"
xml:lang="en" lang="en">
```

2. Coding（HTML validity and semantic ）

-HTML Validity

Use valid HTML code as much as possible. *There might be some exceptional occasion.
Use tools such as the W3C HTML validator (http://validator.w3.org/) to test.

<!-- Recommended -->

```
<img src=""><br>
<article>lorem ipsum </article>
```

<!-- Not recommended -->

```
<img src=""><br />
<article>lorem ipsum
```

No close tag <article>

*unless specified by client , please use HTML 5 instead of HTML 4 or XHTML

Semantics

Use HTML according to its purpose.

For example, use heading elements for headings, p elements for paragraphs, a elements for anchors, etc. JS setting such as "onclick" should not be included as elements.

<!-- Recommended -->

```
<a href="recommendations/">All recommendations</a>
<h1>heading 1</h1>
```

<!-- Not recommended -->

```
<div onclick="goToRecommendations();">All recommendations</div>
<div><p class="heading">heading 1</p></div>
```

2. Coding （Multimedia Fallback）

-Multimedia Fallback

Provide alternative contents for multimedia.

For multimedia, such as images, videos, animated objects via canvas, make sure to offer alternative access. For images that means use of meaningful alternative text (alt) and for video and audio transcripts and captions, if available.

For images whose purpose is purely decorative which you cannot immediately use CSS for, use no alternative text by setting alt="".

This rule is very important. You MUST follow this rule unless other instruction.

<!-- Recommended -->

```
<img src="banner.jpg"
alt="banner adv">
```

<!-- Not recommended -->

```
<img src="banner.jpg">
```

2. Coding（External styling）

-Separation of Structure

Separate structure from its look from behavior.

Strictly keep structure (markup), look (styling), and behavior (scripting) apart.

<!-- Recommended -->

<link rel="stylesheet" href="style.css">
<h1>test</h1>

<!-- Not recommended -->

<h1 **style="text-decoration:bold"**>test</h1>

*unless specified by client, don't use inline and internal styling

Writing inline is basically prohibited.

-Entity References

Use entity reference for special characters.

Often used special characters are as below.

<!-- Recommended -->

© ®

*unless specified by client, please
Use HTML entity code

<!-- Not recommended -->

© ®

2. Coding（Type Attributes）

-Type Attributes

Omit type attributes for style sheets and scripts.
Do not use type attributes for style sheets (unless not using CSS) and scripts (unless not using JavaScript).
Specifying type attributes in these contexts is not necessary as HTML5 implies text/css and text/javascript as defaults.

<!-- Recommended -->

<!-- Not recommended -->

```
<link rel="stylesheet" href="style.css">

<script src="script.js"></script>
```

```
<link rel="stylesheet" href="style.css"
type="text/css">

<script src="script.js"
type="text/javascript"></script>
```

* Do not use type attributes for style sheets (unless not using CSS) and scripts (unless not using JavaScript).
Specifying type attributes in these contexts is not necessary as HTML5 implies text/css and text/javascript as defaults.

2. Coding （General Formatting）

■ HTML Formatting Rules

-General Formatting

Use a new line for every block, list, or table element, and indent every such child element.
 (If you run into issues around whitespace between list items it's acceptable to put all li elements in one line.)

<!-- Recommended -->

```
<blockquote>
  <p><em>Space</em>, the final
frontier.</p>
</blockquote>

<ul>
  <li>Moe</li>
</ul>

<table>
…
</table>
```

<!-- Not recommended -->

```
<blockquote>
  <p><em>Space</em>, the final
frontier.</p>
</blockquote>
<ul>
  <li>Moe</li>
</ul>
<table>
…
</table>
```

*please give line break after block, list or table tag

2. Coding（CSS validity & ID and Class naming）

■ CSS Style Rules

-CSS Validity

Use valid CSS as much as possible.

Exception: if you need to deal with CSS validator bugs or use requires syntax, use may not valid CSS code.

Use tools such as the W3C CSS validator (https://jigsaw.w3.org/css-validator/validator.html.en) to test.

Using valid CSS is a measurable baseline quality attribute that allows to spot CSS code that may not have any effect and can be removed, and that ensures proper CSS usage.

-ID and Class Naming

Use meaningful or generic ID and class names.

Instead of presentational or cryptic names, always use ID and class names that reflect the purpose of the element.

<!-- Recommended -->

```
/*meaning*/
#gallery .galleryContainer  .galleryContainerHeading:first-child{}

.galleryContainerList:nth-of-type(2) .galleryReadmore:nth-of-type(even) {}

#gallery:last-of-type {}
```

*Make the name to represent the Site Display, not the Site Function
**Avoid the use of number, use CSS Selector properties to change specific css

<!-- Not recommended -->

```
/*meaningless*/
#gal #WrpCnt  .bg10btn {}
```

2. Coding （Shorthand properties）

-Shorthand Properties

Use shorthand properties as much as possible.

CSS offers a variety of shorthand properties (like font) that should be used whenever possible, even in cases where only one value is explicitly set.

Using shorthand properties is useful for code efficiency and understandability.

<!-- Recommended -->

```
/* Recommended */
font: 100%/1.6 palatino, georgia, serif;
padding: 0 1em 2em;
```

<!-- Not recommended -->

```
/* Not recommended */
font-family: palatino, georgia, serif;
font-size: 100%;
line-height: 1.6;
padding-bottom: 2em;
padding-left: 1em;
padding-right: 1em;
padding-top: 0;
```

2. Coding（Color code format and values)

-0 Leading 0s and Units

Omit unit specification after "0" values. Omit leading "0"s in values.

Do not use 0 in front of values or lengths between -1 and 1.

-Hexadecimal Notation (Color Code)

Use 3 character hexadecimal notation where possible.

For color values that permit it, 3 character hexadecimal notation is shorter and more succinct.

<!-- Recommended -->

<!-- Not recommended -->

Margin: 0 auto auto auto;
Padding: 0 auto 10px 1px;

Font-size: .8em;

Background: #ebc;

⬆

Only if color can be shortened, No need to use for #123456 , #00000F , #AAABBB

Margin: 0**px** auto auto auto;
Padding: 0**px** auto 10px 1px;

Font-size: **0**.8em

Background: #e**eb****b**cc;

2. Coding（Prefixes）

-Prefixes

Prefix selectors with an application-specific prefix (optional).

In large projects as well as for code that gets embedded in other projects or on external sites use prefixes (as namespaces) for ID and class names. Use short, unique identifiers followed by a hyphen.

```
.adw-help {} /* AdWords */
#maia-note {} /* Maia */
```

2. Coding （ID and Class name Delimiter)

-ID and Class Name Delimiters

For ID and Class names, using lower camel case is recommended.

*About camel case

Camel case is the practice of writing compound words or phrases such that each word or abbreviation begins with a capital letter. (Referrer URL : http://en.wikipedia.org/wiki/CamelCase)

・**Types of camel case**

Name	Example	Note
Upper camel case （UCC） 、 or **Pascal case**	Get Input Reader	Write the head of compound word with capital letter
Lower camel case （LCC） 、 or **camel case**	get Input Reader	Write the head of compound word with lowercase letter.

<!-- Recommended -->

#videoId{};

.featuredContainerList{};

<!-- Not recommended -->

#video_Id{}; / #video-Id{};

.featuredcontainerList{};

2. Coding（CSS formatting rules)

-CSS Hacks

Avoid user agent detection as well as CSS "hacks"—try a different approach first.

It's tempting to address styling differences over user agent detection or special CSS filters, workarounds, and hacks. Both approaches should be considered last resort in order to achieve and maintain an efficient and manageable code base.

■CSS Formatting Rules

-Block Content Indentation

Indent all block content to make hierarchy understandable.

<!-- Recommended -->

```
#videoId{
  Height: 100px;
  Width: 100px;
};

.featuredContainerList{
  Height: 5px;
};
```

* Please remember to use 2 spaces for indent.

<!-- Not recommended -->

```
#videoId{
Height: 100px;
Width: 100px;
};

.featuredContainerList{Height: 5px;};
```

-Declaration Ends

Use a semicolon after every declaration.

<!-- Recommended -->

```
#videoId{
  Height: 100px;
  Width: 100px;
};


.featuredContainerList{
  Height: 5px;
};
```

<!-- Not recommended -->

```
#videoId{
  Height: 100px;
  Width: 100px
};


.featuredContainerList{
  Height: 5px
};
```

Property Name Ends

Use a space after a property name's colon. Lower importance, no need be thoroughly.

<!-- Recommended -->

```
Height: 5px;
font: 100%/1.6 palatino, georgia, serif;
padding: 0 1em 2em;
```

Spaces after colon

<!-- Not recommended -->

```
Height:5px;
font:100%/1.6 palatino, georgia, serif;
padding:0 1em 2em;
```

No spaces after colon

2. Coding（Selector and Declaration separation）

-Selector and Declaration Separation

Separate selectors and declarations by new lines.

<!-- Recommended -->

```
#videoId,
#galleryId,
#footerId{
  Height: 100px;
  Width: 100px;
};
```

<!-- Not recommended -->

```
#videoId, #galleryId, #footerId{
  Height: 100px;
  Width: 100px;
};
```

-CSS Rule Separation

Separate rules by new lines. Always put a blank line (two line breaks) between rules.

<!-- Recommended -->

```
html {
  background: #fff;
}

body {
  margin: auto;
  width: 50%;
}
```

<!-- Not recommended -->

```
html {
  background: #fff;
}
body {
  margin: auto;
  width: 50%;
}
```

← No Line Break

2. Coding （Important point）

■ CSS Meta Rules

-Section Comments

Group sections by a section comment (optional).

/* Header */

#adw-header {}

/* Footer */

#adw-footer {}

/* Gallery */

.adw-gallery {}

■ IMPORTANT POINT

Be consistent.

If you're editing code, take a few minutes to look at the code around you and determine its style. If they use spaces around all their arithmetic operators, you should too. If their comments have little boxes of hash marks around them, make your comments have little boxes of hash marks around them too.

The point of having style guidelines is to have a common vocabulary of coding so people can concentrate on what you're saying rather than on how you're saying it. We present global style rules here so people know the vocabulary, but local style is also important. If code you add to a file looks drastically different from the existing code around it, it throws readers out of their rhythm when they go to read it. Avoid this.

3. TCID standard Requirment

3. (TCID standard Requirment)

TCID standard Requirment

elemet	TCID standar	notes
DOCTYPE	<!DOCTYPE html>	use HTML 5
Content-Style-Type	<meta http-equiv="Content-Style-Type" content="text/css" />	
CSS version	CSS 3	Use CSS 2 if needed
Insert style sheet	Use external style sheet	<link href="" rel="stylesheet">
CSS chartshet	@charset "utf - 8";	
color	use hexadecimal color (#fffff)	
URL (link)	Use root path link	

4.HTML Markup Rules

4. HTML Markup Rules (Tag attributes)

The following HTML tag <u>attributes</u> are abolished or deprecated; instead, use CSS.

Tag	Attribute	Alternative CSS
<basefont>	size	font-size
	color	color
	face	font-family
<body>	alink	a:active
	background	background
	bgcolor	background
	link	a:link
	text	color
	vlink	a:visited
 	clear	clear
<font>	size	font-size
	color	color
	face	font-family
	compact	float & width & margin
	type	list-style-type
	compact	float & width & margin
	type	list-style-type
	type	list-style-type
<dl>	compact	float & width & margin
<h1>～<h6>	align	text-align
<div>	align	text-align
<p>	align	text-align

Tag	Attribute	Alternative CSS
<hr>	noshade	border
	size	border
	width	width
<input>	align	float vertical-align
<legend>	border	border
<iframe>	border	border
<img>	hspace	margin-left & margin-right
	vspace	margin-top & margin-bottom
<pre>	width	width
<object>	align	float
	border	border
	hspace	margin-left & margin-right
	vspace	margin-top & margin-bottom
<caption>	noshade	border
<table>	align	text-align
	bgcolor	background
<td>, <th>	bgcolor	background
	height	height
	width	width
<td>, <th>	nowrap	white-space
<tr>	bgcolor	background

4. HTML Markup Rules (HTML Tag attribute)

The following HTML tag <u>attributes</u> are abolished or deprecated;
instead, use other tag/attributes.

Tag	Attribute	Altenative Tag/Attribute
<map>	**name**	【XHTML 1.0】 <map id="umap" name="umap"> 【XHTML 1.1】 <map id="umap">
<a>	**name**	【XHTML 1.0】 <div id="anchor01" name="anchor01"> 【XHTML 1.1】 <a id="anchor01">
	target	No Alternative Tag/Attribute. DO NOT use the attribute.
<base>	**target**	No Alternative Tag/Attribute. DO NOT use the attribute.
<form>	**name**	【XHTML 1.0】 <form action="hoge" method="post" name="frm" id="frm"> 【XHTML 1.1】 <form action="hoge" method="post" id="frm">
	target	No Alternative Tag/Attribute. DO NOT use the attribute.
<link>	**target**	No Alternative Tag/Attribute. DO NOT use the attribute.
<html>	**version**	<!DOCTYPE>
<script>	**language**	<script type="text/javascript">

4. HTML Markup Rules (HTML Tag attribute)

The following HTML tag <u>attributes</u> are abolished or deprecated; instead, use CSS.

Tag	Alternative Tag or CSS
<b>	font-weight
<big>	font-size
<i>	font-style
<center>	text-align
<strike>	text-decoration
<s>	text-decoration
<u>	text-decoration
<noframes>	No Alternative Tag/Attribute. DO NOT use the attribute.
<small>	font-size
<tt>	No Alternative Tag/Attribute. DO NOT use the attribute.
<applet>	<object>
<isindex>	<input type="text">
<dir>	
<listing>	<pre>
<menu>	
<plaintext>	<pre>
<xmp>	<pre>
<frame>	No Alternative Tag/Attribute. DO NOT use the attribute.
<frameset>	No Alternative Tag/Attribute. DO NOT use the attribute.

Do Not use uppercase in extensions

File format	extensions
HTML file	html
Style sheet file	css
JavaScript file	js
Text file	txt
GIF image file	gif
JPEG image file	jpg
PDF file	pdf
Flash file	swf
Windows Media video file	wmv
Quick Time movie file	mov
self-extract compressed file	exe
Compressed file for windows	zip

File Name Rule

- Extension have to be "html"
- The first page of each category should be named "index.html"
- Do Not use uppercase in file name
- File name should describe the page contents
- Use English words
- Use underscore"_" if you want to connect two or more words

4. HTML Markup Rules | Head

<title>

o   Title is required in all HTML documents and it defines the title of the document.

o   You can't have more than one <title> element in an HTML document

<title> Digital Marketing | service | Transcosmos Indonesia </title>

<meta name="description">

o   Meta description will appears on search engine results page under page title

o   Aim 156 characters including spaces

o   Try for no more than 160 (English) and 124 (Japanese) characters

<meta name="description" content="content="transcosmos Indonesia is a professional Call Center Outsource service and digital marketing service provider" />

<meta name="keyword">

o   Meta keyword tags where major factor search engines consider when rangking sites before.

o   Please do 3~5 words are fine.

<meta name="keywords" content="transcosmos, call center, digital marketing" />

<link href="/favicon.ico" rel="icon" type="images/x-icon">

o   Icon is represent the document.

o   Favicon is a small graphic that is associated wiwth a page or website.

o   Favicon allow allows web developer to custumize the tab bar that is displayed in many browser as well as in the bookmarks when site is saved.

o   The icon should be 16 X 16 pixels and ICO format

<link href="/favicon.ico" rel="shortcut icon" type="image/x-icon" />
<link href="/favicon.ico" rel="icon" type="image/x-icon" />

<link>

o   Link tag is defines the relationship between a document and external resource.

o   In HTML the <link> tag has no end tag

o   In the XHTML the <link> tag must be properly closed.

```
<link href="/common/css/base.css" rel="stylesheet" type="text/css" media="all" />
<link href="/common/css/parts.css" rel="stylesheet" type="text/css" media="all" />
<link href="/common/css/print.css" rel="stylesheet" type="text/css" media="print" />
<!-- local css -->
<link href="/css/index.css" rel="stylesheet" type="text/css" media="all" />
<!-- /local css -->
```

<script>

o   Import javascript tag should after import CSS tag, because showing structure correctly is the first priority.

```
<script src="/common/js/jquery.js" type="text/javascript"></script>
<script src="/common/js/jquery.common.js" type="text/javascript"></script>
<!- local js >
<script src="/js/top.js" type="text/javascript"></script>
<!-- /local js -->
```
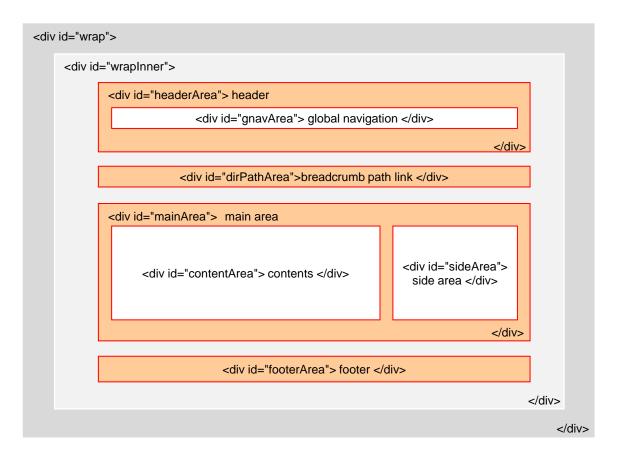
<!--..-->

o Comment tag is used for insert comments in the source code.

o Comment are not showing in the browser

o Use comment to explain the code, which can help you when you edit the source code at a later date

```
<start class="class_name">
The quick brown fox jumps over the lazy dog
<!-- /.class_name --></end>

<start class="class_name" id="id_name" >
The quick brown fox jumps over the lazy dog
<!-- /.class_name#id_name --></end>
```

4 HTML Markup Rules |ID naming

The naming rule of Div that construct main section

```
<div id="wrap">

    <div id="wrapInner">

        <div id="headerArea"> header

            <div id="gnavArea"> global navigation </div>

                                                    </div>

        <div id="dirPathArea">breadcrumb path link </div>

        <div id="mainArea">   main area

            <div id="contentArea"> contents </div>    <div id="sideArea">
                                                       side area </div>

                                                    </div>

        <div id="footerArea"> footer </div>

                                                    </div>

                                                </div>
```

Id name	Describe
wrap	Div wraps whole body.
wrapInner	Div wraps page area that is set aligned center.
headerArea	Div wraps whole header section.
gnavArea	Div wraps whole global navigation section.
dirPathArea	Div wraps whole breadcrumbs path section.
mainArea	Div wraps contents area and side area section.
contentsArea	Div wraps whole contents section.
sideArea	Div wraps whole side section.
footerArea	Div wraps whole footer section.

43

5.CSS Markup Rules

CSS

```
.clear:before,
.clear:after{
        display:table;
        content:"";
}

.clear:after{
        clear:both;
}

.clear{
        zoom:1;
}
```

HTML

```
<div class="clear">
    <div class="columnBlock01">
        aaaa
    </div>
    <div class="columnBlock02">
        bbbbb
    </div>
</div>
```
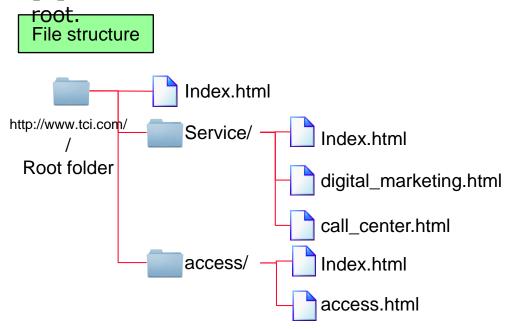
There are three ways to define the destination of a hyperlink.
[1]absolute link: An absolute link is created when the href value is a fully qualified URL.
[2]relative : A relative link is created when the destination href value is relative to the location of the current webpage. The ../ directs the browser to move (up) out of the current directory.
[3]root: A root-relative link is created when the href value is relative to the website root.

File structure

Sample

http://www.tci.com/
/
Root folder

Index.html

Service/
— Index.html
— digital_marketing.html
— call_center.html

access/
— Index.html
— access.html

[1]absolute link : External Link
href="http://www.tci.com/" or href="http://www.tci.com/index.html"
[2]relative : in This Project
href="../" or href="../index.html"
[3]root link : TCI Standard
href="/" or href="/index.html"

example

```
/* ====================================================================
1st level section

============================================================== *
/* 2nd level section
-----------------------------------*/
div.imageLeft div.image {
  margin: 0 15px 15px 0;
  float: left;
}

div.imageRight div.image {
  width: ;
  margin: 0 0 15px 15px;
  float: right;
}
```

A CSS comment starts with $/*$ and ends with $*/$. Comments can also span multiple lines

5. CSS markup rules |shorthand

Use shorthand

property	value
margin	-top,-right,-bottom,-left
padding	-top,-right,-bottom,-left
background	-color,-image,-repeat,-position,-attachment
border	-width,-style,-color
border-top border-right border-bottom border-left	-width,-style,-color
border-width border-stlye border-color	-top,-right,-bottom,-left
list-style	-type,-image,-position
font	-font-style,-font-weight,-font-size,-line-height,-font-family

5. CSS markup rules |File naming Rule

Common CSS files that are shared many pages

- **base.css** ⋯include reset CSS that resets browser's consisting CSS setting as default. Also include CSSs that construct page layout like page background, header, footer, global navigation and so on.
- **parts.css** ⋯include CSSs that are used in contents area such as strong text, margin set, basic list style and so on.
- **print.css** ⋯include CSSs that construct layout for hard copy.

CSS that shared by only specific category

category.css *Do NOT change	category.css	Sample

CSS that used by only specific page

[file name].css	access.css	Sample

5. CSS markup rules |Reset.css

Reset.css

o Reset.css use to reduce browser inconsistencies in things like default line heights, margins and font sizes of handlings, and so on.

o Change the value that showed in red texts depend on design.

```
html{background:#ffffff;}
body,div,dl,dt,dd,ul,ol,li,h1,h2,h3,h4,h5,h6,pre,code,form,fieldset,input,textarea,p,blockquote,th,td{margin:0;padding:0;}
table{border-collapse:separate;border-spacing:0;}
fieldset,img{border:0;}
img{vertical-align:top;}
address,caption,em,strong,th{font-style:normal;font-weight:normal;}
ol,ul{list-style:none;}
caption,th,td{text-align:left;}
h1,h2,h3,h4,h5,h6{font-size:100%;font-weight:normal;}
sup{vertical-align:text-top;}
sub{vertical-align:text-bottom;}
input,textarea,select{font-family:inherit;font-size:inherit;font-weight:inherit;}
input,textarea,select{*font-size:100%;}
a{color:#;}
a:link{text-decoration:underline;}
a:visited{text-decoration:underline;}
a:hover{text-decoration:none;cursor:pointer!important;}
a:active{text-decoration:underline;}
```

5. CSS markup rules |tag naming rule

[part name]Block[ALPHABET][number(00)] headingBlockA01 Sample

[part name]Block[ALPHABET][number(00)]

Part name	Description
heading	h1-h6
table	table
list	ul, ol, dl, li, dt, dd
link	text link with icon
caution	comment statement
caption	caption of images/tables/graph
imageLeft/imageRight	an image that appear inside the text with the text flowing around it.
btn	button

Part name	Description
bg	element that is set only background
border	element that is set only border
bgBorder	element that is set only border and background
formTable	table that used for form
anchor	Anchor link (internal page link)
indent	Elements that set horizontal margin
column	Column section
radius	Element that is decorated rounded corner

5. CSS markup rules |Class naming rule

[part name]Block[ALPHABET][number(00)]
headingBlockA01

- Use upper case from A to Z
- Change characters depend on design

[part name]Block[ALPHABET][number(00)]
headingBlockA01

- from 01 to 99
- Change number depend on
  color difference / margin difference of
  the same design element

headingBlockA02

headingBlockA01

headingBlockB01

headingBlockB02

5. CSS markup rules CSS unit

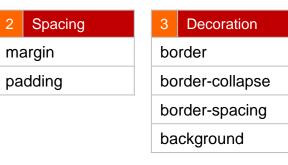| property	Units	Sample
font-size	%	150%
line-height	No Unit/the second decimal place	1.08
margin/padding	px	10px
background-position	px	50px 30px
background-position for icon like list style	em	1.3em
border	px	1px

5. CSS markup rules CSS Order of Properties

Mark up CSS property according to the designated arranged order of the following

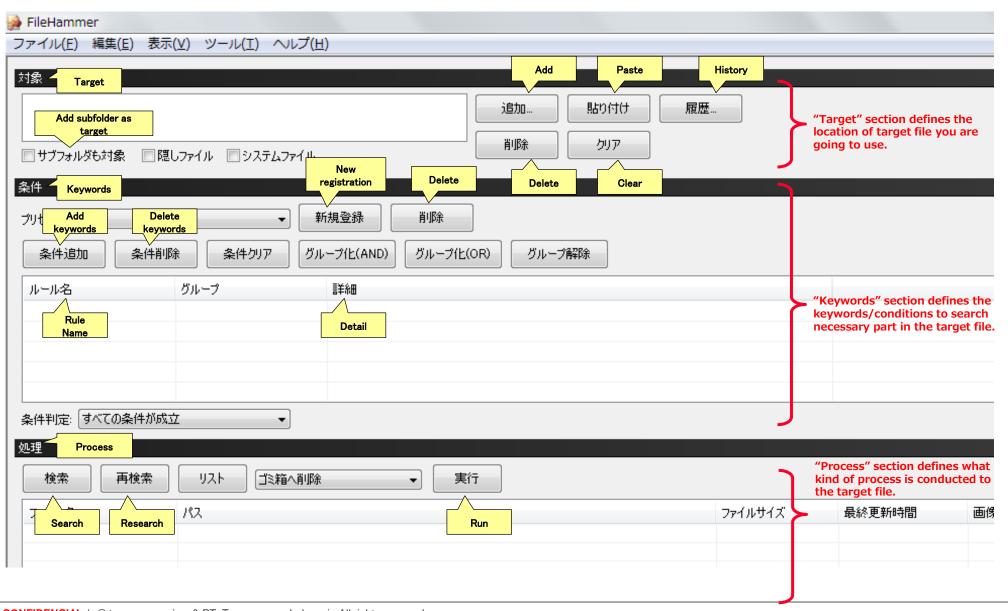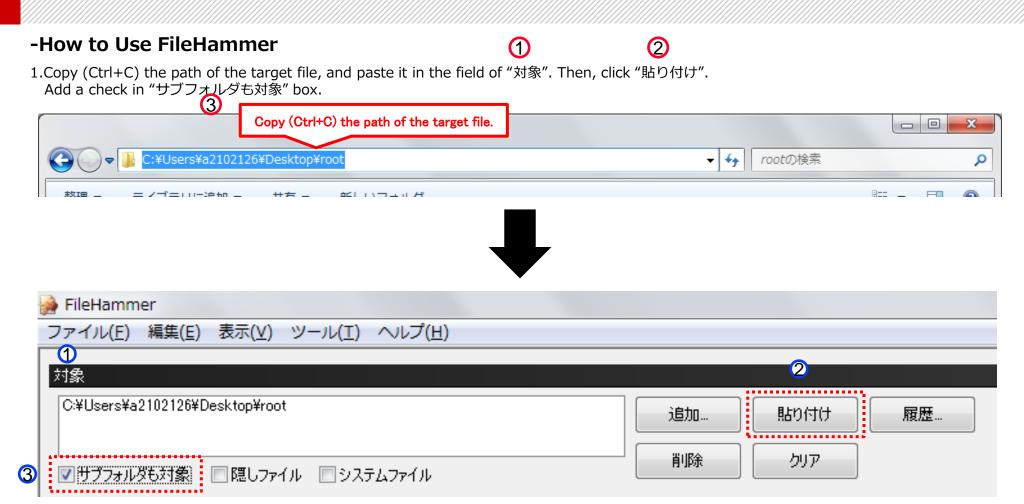Start   End

1 Appearance	2 Spacing	3 Decoration	4 Text decoration	5 List Style	6 Others
display	margin	border	color	list-style	ime-mode
float	padding	border-collapse	font		content
clear		border-spacing	text-align		cursor
position		background	vertical-align		zoom
Top / right / bottom / left			line-height		opacity
z-index			white-space		filter
min-width			text-indent		
width			text-decoration		
min-height			letter-spacing		
height			word-break		
overflow			word-wrap		

6. Compile Files

File Hammer

FileHammer is a software for copying and deleting the file which is searched with multiple keywords.
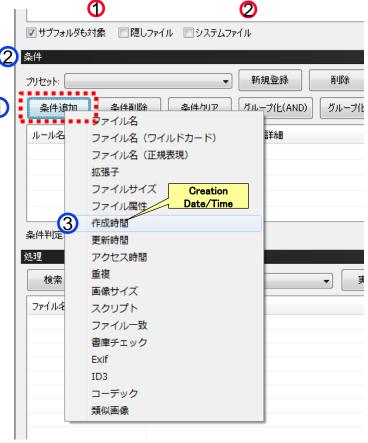*Send only Updated Files to TCI

6. Compile Files

6. Compile Files

-How to Use FileHammer

①   ②

1.Copy (Ctrl+C) the path of the target file, and paste it in the field of "対象". Then, click "貼り付け".
  Add a check in "サブフォルダも対象" box.

③

Copy (Ctrl+C) the path of the target file.

C:¥Users¥a2102126¥Desktop¥root

FileHammer

ファイル(F) 編集(E) 表示(V) ツール(T) ヘルプ(H)

①

対象

②

C:¥Users¥a2102126¥Desktop¥root

追加...   貼り付け   履歴...

削除   クリア

③ ☑サブフォルダも対象 ☐隠しファイル ☐システムファイル

6. Compile Files

-How to use FileHammer

③

2.Click "条件追加" button in the "条件" section and select "作成時間"
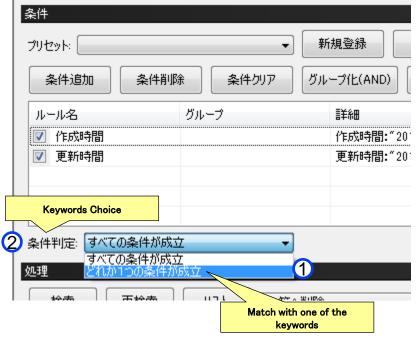
① ②

3.For example, you've delivered the files on November 16, and you want to extract the files that you've created or added after November 17. In this case, please set up the dialogue box of "条件：作成時間" as below and click "OK".

6. Compile Files

-How to Use FileHammer

③

4. Click "条件追加" button in the "条件" section, and select "更新時間".

① ②

② 条件

① 条件追加

③ 更新時間

Update Date/Time

5.For example, you've delivered the files on November 16, and you want to extract the files that you've updated after November 17. In this case, please set up the dialogue box of "条件：更新時間" as below and click "OK".

Keywords：Update Date/Time

条件：更新時間

時間

Define Date/Time

現在の時刻

時間を指定

2014 年 11 月 17 日 0 時 0 分 現在の時間

Year Month Day Hour Minute

判定

以降 詳細: 0 分 有効時間: 年:月:日:時:分

After

OK キャンセル

6. Compile Files

-How to Use FileHammer

①   ②

6.Select "どれか一つの条件が成立" from "条件判定" section. In this way, you can extract the files which were created or updated after November 17. ***CAUTION: The created date of file will be updated every time the file is zipped/unzipped or downloaded from the shared server. Thus, be careful when you put the keywords "created/updated date", you may fail to extract the target files.**

条件

プリセット: [▼]   [新規登録]

[条件追加] [条件削除] [条件クリア] [グループ化(AND)]

ルール名	グループ	詳細
☑ 作成時間		作成時間:"201
☑ 更新時間		更新時間:"201

Keywords Choice

② 条件判定: [すべての条件が成立   ▼]
        すべての条件が成立
        どれか1つの条件が成立   ①

処理

Match with one of the keywords

③   ④

7. As clicking "検索" button in "処理" section, the list of files matching with the keywords will be displayed.

④ 処理

[検索] ③ [索] [リスト] [ゴミ箱へ削除 ▼] [実行]

Find

ファイル名	パス	ファイルサイズ	最終更新時間	画(
☑ ◉ aogare.html	C¥Users... 102126¥Desktop¥root¥earthgarden¥zukan¥byouki¥aogare.html	4,038	2014/11/17 05:23	
☑ ◉ aomushi.html	...¥Desktop¥root¥earthgarden¥zukan¥gaichu¥aomushi.html	4,453	2014/11/17 06:53	
☑ ◉ ari.html	C¥Users... arthgarden¥zukan¥gaichu¥ari.html		2014/11/17 05:26	
☑ ◉ azamiuma.html	C¥Users... arthgarden¥zukan¥gaichu¥azamiuma.html		2014/11/17 05:22	
☑ ◉ barazoumushi.h...	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥zukan¥gaichu¥barazoumushi.html	4,977	2014/11/17 06:54	
☑ ◉ basho.html	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥knowledge¥basho.html	8,366	2014/11/17 05:23	
☑ ◉ bouchu.html	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥faq¥bouchu.html	5,151	2014/11/17 03:20	
☑ ◉ byoki.html	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥howto¥kaju¥blueberry¥byoki.html	6,064	2014/11/17 05:27	
☑ ◉ byoki.html	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥howto¥yasai¥kyabetsu¥byoki.html	8,970	2014/11/17 06:26	
☑ ◉ byoki.html	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥howto¥yasai¥rucola¥byoki.html	6,585	2014/11/17 06:38	
☑ ◉ byoki.html	C¥Users¥a2102126¥Desktop¥root¥earthgarden¥howto¥yasai¥tamanegi¥byoki.html	8,853	2014/11/17 03:06	

Path

Last Updated Date/Time

File Name

レディ

6. Compile Files

-How to Use FileHammer

①

8.Selct how to process the target files from the pull-down menu. If you want to extract the updated files, select "コピー先（階層構造を維持)"

Pull-down menu to select processing details

Copy Destination(with keeping hierarchical structure)

The files with checkmark are selected for processing..

*CATION

If you select the wrong processing details, the target files will be deleted or moved to wrong destination. **Please carefully make the selection.**

Copy Destination(without keeping hierarchical structure)

Move Destination(without keeping hierarchical structure)

Delete

Copy Destination(with keeping hierarchical structure)
*Select this one

Move Destination(with keeping hierarchical structure)

6. Compile Files

-How to Use FileHammer

①

9.After selecting "コピー先（階層構造を維持）" from the pull-down menu, click "run" button and select the folder of copy destination.

Run

①

Select the folder as copy destination.

10.After clicking "OK" button, the target files (matching with the keywords) will be copied to the folder which you set as the copy destination.

*The hierarchical structure of the copied files/folder will be kept as long as you've selected "Copy Destination (with keeping hierarchical structure)".

6. Remove Thumb.db

Thumb.db is an image chache which makes thumbnail viewing faster, this is automatically created in windows vista ,7,8 whenever images are encountered in a folder. It's usually hidden but can appear, disappear and is often impossible to delete.

Please disable the windows from auto-generating thumbs.db

Open windows explorer → Organize → folder and search options

6. Remove Thumb.db

Choose View → uncheck show icons, never thumbnails

Note:
Please make sure that computer is already set for **showing All hidden file**.

How to set:
Choose Hidden files and folders → Show hidden folder and drives

Click Apply

FilelistCreator

FilelistCreator is a software for creating the list of files. It is useful to create the list of delivery files.

/ © transcosmos inc. & PT. Transcosmos Indonesia All rights reserved.

66

6. Compile Files

-Features

FilelistCreator is a software for creating the list of files. It is useful to create the list of delivery files.

Specify the folder location in which you want to make a list of files.

You can select the contents of the file list here. Select File Name, File Format, and File Path.

In order to display all the paths, please set up the "File Path Options" as below.

The preview of the file list will be shown as above.

You can select the format of the file list. Select ".txt" format.

6. Compile Files

-Features

After the result file is created, delete your local (computer) path, only use root path standard

-Before

Use local path, please remove it

-After

Use root path standard

7. How To Compile Files for Delivery

7. Compile Files

For every delivery, please send only the files you've worked on or updated. Use FileHammer to extract the updated files, and make the list of updated files by using FilelistCreator. Then, upload the file list to WebDAV along with the deliverable files. The directory is as below.

WebDAV

- **Client Name (i.e. earth-chem)**
  - **Project Name (i.e EarthGarden_SiteRenewal)**
    - **Project Management Number (i.e. IND-SPT-005)**
      - **Indonesia**
        - **Purpose_yymmdd_001 (i.e. Delivery_140922_001)** ← **Zip this folder and upload it to webDAV as a deliverable.**
          - **Data Type (i.e. design, html, etc…)**
            - **root**
            - **QA Sheet**
            - **List of deliverable files**

〈In case of delivering HTML files…〉
Please save the files in the root folder which contains the existing hierarchical structure, then upload the root folder as shown on the left. In addition, please include the QA Sheet and the list of deliverable files every time you deliver the files.

8. How to Use Checking Tools For Recruit Project

Last updated : 6/3/20015

transcosmos
Indonesia

The Marketing Chain Management Company

1. Instruction

This check tools is additional checking for recruit project only, after finish check using QA check sheet, please use this check tools to check the result of recruit project

1. Check tools sheets form

For recruit error checking please open "【オフショア用】メール配信_チェックリスト".xls at this directory:

Z:¥TCID¥01_案件¥02_本案件¥Recruit¥00_common¥03_checksheets¥リクルートCRMテスト環境検出ツール

Check only for **point 8** and **point 11**
*no need to check all of them

No	【メールHTML納品時】運用チーム向け（確認チェックシート）	
	企画内容チェック項目	運用者
3	ヘッダー、フッターの内容があっているか	☐
	-ヘッダー、フッターの内容がその媒体になっているか	☐
	-注意文言（例；配信停止、プラポリ、お問合せ先）などは各媒体の通常メルマガで使っている文言か	☐
4	URLにVOSコードはついているか。	
	-メール内のURLについてVOSは入っているか。入っていなければ企画者へ確認	☐
5	メールのファイルサイズは正しいか。	
	メールのファイルサイズが100KB以内であるか（ポイントお得メールは特に注意） ※ポイントお得メール以外の媒体は各媒体の指定するファイルサイズに準じる	☐
7	リンクエラーがないか	☐
	パスの書き方が絶対パスになっているか（相対パスではないことを確認）	☐
8	メールのURLに、以下のテスト環境のURLが入っていないかをチェックツールを使って確認	
	d1.d2.d6. stg, test,dev1,wwwtst,wwwtst1,wwwtst2,dev1-point,cdev1-recruit-card, d1.golfNULLGWDOCOMO	☐
10	テキストメールの半角スペース確認	☐
	テキストの場合は、リンクの前後は改行になっているか	☐
11	機種依存文字の確認	
	機種依存文字が入っていないか(機種依存文字はエイケアセットできません) 機種依存文字とは、丸付き文字・括弧付き文字・ローマ数字など。　例）①②③ⅠⅡ♥""'" 機種依存文字チェッカー　http://www.submit.ne.jp/tool/uniquechar/check.html	☐

【オフショア用】メール配信_チェックリスト".xls

Please download the check tools from this location
(Z:¥TCID¥01_案件¥02_本案件¥Recruit¥00_common¥03_checksheets¥リクルートCRMテスト環境検出ツール)

Checktools

Create new folder at your project directory , Give the folder name, for the example "**checktools**"
Copy the check tools and HTML page that you want to check into "**checktools**" directory

Rename The HTML file into alfabetic (If the HTML name using japanese word)

2.2 How To check Point 8

Double Click at check tools will show the common prompt terminal interface

Drag and drop your HTML file from directory "**checktools**" to common prompt terminal and press enter

2.3 How To check Point 8

After push enter button, the tools will generate result file into "**checktools**" directory

Name	Date modified	Type	Size
03	17/02/2015 14:47	HTML Document	41 KB
ífîŝîïë┴	24/02/2015 15:05	Text Document	0 KB
テスト環境検出ツール	18/12/2013 14:42	Windows Batch File	2 KB

Please check based on this file, If there are no mistake, The file will shown as a blank page and If there are any mistake the file will show the mistake

ífîŝîïë┴ - Notepad

File   Edit   Format   View   Help

sample.html <!--cdev1-recruit-card-->

* For example the picture above are the result file with error message

	メールのURLに、以下のテスト環境のURLが入っていないかをチェックツールを使って確認	
8	d1.d2.d6. stg, test,dev1,wwwtst,wwwtst1,wwwtst2,dev1-point cdev1-recruit-card, d1.golfNULLGWDOCOMO	☐

3.1 How To check Point 11

Please open this URL http://www.submit.ne.jp/tool/uniquechar/check.html

Copy and paste the HTML work to the website

機種依存文字チェッカー

下の枠の中に、テキストを記入してチェックボタンを押して下さい。
機種依存文字（環境によって文字化けしてしまう文字）が入っていると赤く表示されます。

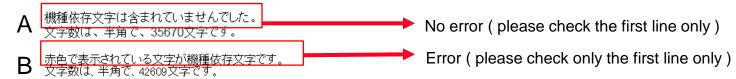The result will show and make sure there are no red mark

Click 機種依存文字をチェック Button

After click button will show the result, If the result is "A" it's mean no error
But if the result is "B" it's mean the file has some error

Result:

A 機種依存文字は含まれていませんでした。
文字数は、半角で、35670文字です。 → No error (please check the first line only)

B 赤色で表示されている文字が機種依存文字です。
文字数は、半角で、42609文字です。 → Error (please check only the first line only)

Thank You