



無料電子ブック

学習

HTML

Free unaffiliated eBook created from
Stack Overflow contributors.

#html

.....	1
1: HTML	2
.....	2
.....	2
Examples.....	2
.....	2
.....	2
.....	3
.....	3
.....	4
2: CSSHTML	6
.....	6
.....	6
Examples.....	6
.....	6
.....	6
.....	7
.....	7
3: Div	8
.....	8
.....	8
Examples.....	8
.....	8
.....	9
4: Doctypes	10
.....	10
.....	10
.....	10
Examples.....	10
Doctype.....	10
HTML 4.01Doctype.....	11

HTML 4.01	11
HTML 4.01 Transitional	11
HTML 4.01	11
HTML 5 Doctype	11
.....	11
Doctypes	12
HTML 3.2	12
HTML 2.0	12
5: HTML 5	13
.....	13
Examples	13
Html 5	13
6: HTML	14
Examples	14
HTML	14
.....	14
7: HTMLJavaScript	15
.....	15
.....	15
.....	15
Examples	15
JavaScript	15
JavaScript	15
JavaScript	16
Javascript	16
8: IFrame	17
.....	17
.....	17
.....	17
sandbox	18
Examples	18
.....	

.....	18
IFrame.....	19
"srcdoc".....	19
.....	19
9: SVG	21
.....	21
.....	21
Examples.....	21
SVGHTML.....	21
image.....	21
.....	21
SVG.....	21
CSSSVG.....	22
10:	24
.....	24
.....	24
Examples.....	25
= "".....	25
role = "alertdialog".....	25
= "".....	25
= "".....	25
role = "banner".....	26
= "".....	26
= "".....	26
= "".....	26
= "".....	27
= "".....	27
= "".....	27
= "contentinfo".....	27
= "".....	28
= "".....	28

role = "directory"	28
= ""	28
= ""	28
= ""	29
= ""	29
= ""	30
= ""	30
= "img"	30
= ""	30
= ""	30
= ""	31
role = "listitem"	31
= ""	31
role = "main"	31
= ""	31
= ""	32
= ""	32
role = "menubar"	32
= "menuitem"	32
= "menuitemcheckbox"	32
= "menuitemradio"	33
= ""	33
= ""	33
role = "option"	33
= ""	33
role = "progressbar"	34
= ""	34
= ""	34
role = "radiogroup"	34
= ""	34
= ""	35
= "rowheader"	35
= ""	35

= ""	36
role = "searchbox"	36
role = "separator"	36
= ""	36
= ""	36
= ""	37
role = "switch"	37
= ""	37
= ""	37
= ""	37
role = "tabpanel"	38
= ""	38
= ""	38
= ""	38
= ""	38
= ""	39
= ""	39
role = "treeitem"	39
11:	41
	41
	41
	41
Examples	41
	42
/	42
	43
JavaScript	43
	44
	44
	45
12:	46
	46

.....	46
Examples.....	46
.....	46
.....	47
.....	48
.....	48
srcset.....	48
srcset.....	48
srcset.....	49
.....	50
13:	51
.....	51
.....	51
.....	51
Examples.....	52
.....	52
.....	52
.....	52
14:	53
.....	53
.....	53
Examples.....	53
.....	53
2.....	53
15: ID.....	55
.....	55
.....	55
.....	55
.....	55
Examples.....	55
.....	55

CSS.....	56
ID.....	56
ID.....	57
.....	57
ID.....	57
.....	59
IDHTML.....	59
W3C.....	59
16:	61
.....	61
.....	61
Examples.....	61
Contenteditable.....	61
17:	63
.....	63
.....	63
.....	63
Examples.....	63
.....	63
Internet Explorer.....	64
.....	64
.....	64
.....	65
18:	66
.....	66
.....	66
.....	66
Examples.....	66
.....	66
.....	66
.....	66
.....	67

URL	67
19:	68
.....	68
.....	68
.....	68
Examples	68
.....	68
.....	68
20:	70
.....	70
Examples	70
.....	70
.....	70
.....	71
Nav.	72
.....	72
.....	72
.....	73
.....	73
.....	74
.....	74
.....	75
21:	76
.....	76
.....	76
Examples	76
.....	76
.....	76
.....	76
22:	78
.....	78
.....	

Examples.....78

.....78

.....78

23:79

.....79

.....79

.....79

Examples.....80

.....80

.....80

theadtbodytfootcaption.....81

.....82

.....83

24:85

.....85

.....85

Examples.....85

.....85

.....85

.....85

.....86

.....86

.....86

.....87

.....87

25:88

Examples.....88

.....88

HTML5.....88

26:89

.....89

.....	89
.....	89
.....	89
Examples.....	89
.....	89
.....	89
.....	90
.....	90
.....	90
.....	91
.....	91
27:	93
.....	93
Examples.....	93
.....	93
.....	93
URL cite.....	93
.....	93
URL cite.....	93
/.....	94
28:	95
.....	95
.....	95
.....	95
Examples.....	95
.....	95
.....	96
.....	96
.....	97
application-name.....	97
author.....	97
description.....	97

generator.....	97
keywords.....	97
.....	98
.....	99
.....	99
Facebook / Open Graph.....	99
Facebook /	99
Twitter.....	100
Google+ / Schema.org.....	100
.....	100
Web.....	100
29:	102
.....	102
.....	102
.....	102
Examples.....	103
` `` \/.	103
.....	104
.....	104
.....	104
30:	106
.....	106
.....	106
Examples.....	106
.....	106
.....	107
31:	108
.....	108
.....	108
.....	108
Examples.....	108
.....	108
.....	

.....	109
.....	110
.....	111
.....	111
32:	113
.....	113
.....	113
.....	113
Examples	113
CSS	113
JavaScript	114
.....	114
.....	114
.....	114
<noscript>	114
.....	115
CSS	115
Web	115
'media'	115
.....	116
dns-prefetchprefetchprerender	116
.....	116
DNS	116
.....	116
.....	116
33:	118
.....	118
.....	118
.....	118
.....	119

Examples.....	120
.....	120
.....	120
.....	121
value.....	121
checked.....	121
.....	121
.....	121
.....	122
.....	122
.....	122
.....	123
.....	123
.....	124
.....	124
/.....	124
.....	124
.....	124
.....	125
.....	125
.....	126
.....	126
E.....	126
.....	126
.....	127
[name].....	127
[type].....	127
[value].....	127
.....	127
.....	128
URL.....	128
.....	128

DateTime -	129
.....	129
.....	129
.....	129
.....	129
.....	130
.....	130
.....	130
DateTime.....	131
34:	132
.....	132
Examples.....	132
ForForm.....	132
.....	133
35:	134
.....	134
Examples.....	134
.....	134
MIME.....	134
36:	135
Examples.....	135
.....	135
HTML.....	135
37:	137
.....	137
.....	137
Examples.....	137
HTML.....	137
38:	138
.....	138
.....	138
Examples.....	138

.....	138
39:	140
.....	140
.....	140
.....	140
Examples.....	140
.....	140
.....	141
40:	142
.....	142
.....	142
Examples.....	142
.....	142
.....	142
Chrome / Safari / Opera.....	142
Firefox.....	143
.....	143
HTML.....	143
41:	145
.....	145
Examples.....	145
.....	145
.....	145
.....	145
.....	145
.....	146
.....	146
.....	147
.....	147
.....	148

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [html](#)

It is an unofficial and free HTML ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official HTML.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: HTMLをいめる

HTMLの language MARKUP Hyper TEXT Mは、XMLタグでにをけるのシステム。これは、WebページやWebアプリケーションのコンテンツをするためににされ、ネットワークですることができます。

テキストとはに、のバージョンのHTMLでは、イメージやビデオなど、さまざまなメディアがサポートされています。

バージョン

バージョン		
1.0	N / A	1994-01-01
2.0	RFC 1866	1995-11-24
3.2	W3C HTML 3.2	1997-11-14
4.0	W3C HTML 4.0	1998-04-24
4.01	W3C HTML 4.01	1999-12-24
5	WHATWG HTML リビングスタンダード	2014-10-28
5.1	W3C HTML 5.1	2016-11-01

Examples

こんにちは

き

HTMLの language MARKUP Hyper TEXT Mは、のコンテンツをすからされるマークアップシステムをします。マークアップとは、HTMLの、どのようにされるかをするものではなく、するものをするをします。ビジュアルはカスケードスタイルシートCSSによってされ、ブラウザによってされます。例えばfontのようなものをにするのは、「にれであり、がしてはならない」[1]。

HTMLはプログラミングと呼ばれることもありますが、がないためマークアップもありません。HTMLタグは、ページのコンテンツにとをします。

は、タグ <element_name> 、タグ </element_name> 、でまれたの、および

`<element_name>...content...</element_name>`のをみ`<element_name>...content...</element_name>`

タグやをたないHTMLがいくつかあります。これらを**ボイド**といいます。Voidには`<img>`、`<meta>`、`<link>`、`<input>`ます。

は、`video`、`audio`、`table`、`footer`などのコンテンツになキーワードとえることができます。

HTMLページは、にものでされ、その、ウェブブラウザによってみられ、され、でがみりまたはのコンテンツにレンダリングされる。

このドキュメントでは、とタグのいにすることがです。

`video`、`audio`、`table`、`footer`

タグ `<video>`、`<audio>`、`<table>`、`<footer>`、`</html>`、`</body>`

の

タグをしよう...

`<p>`タグはのをします。

には、タグとタグがあります。タグには、のが `<p>` でまれています。タグはタグとじで、との `</p>` のにスラッシュ / がされています。

コンテンツはの2つのタグのにります `<p>This is a simple paragraph.</p>`

なページの

のHTMLは、な**"Hello World"** Webページをします。

HTMLファイルはの**テキストエディタ**をしてできます。HTMLファイルとしてされるには、ファイルを`.html`または`.htm` [2]でするがあります。

、このファイルはのウェブブラウザでくことができます。

```
<!DOCTYPE html>
<html lang="en">
```

```
<head>
  <meta charset="UTF-8">
  <title>Hello!</title>
</head>

<body>
  <h1>Hello World!</h1>
  <p>This is a simple paragraph.</p>
</body>

</html>
```

なページブレイクダウン

このでされるタグはのとおりです。

タグ	
<!DOCTYPE>	ドキュメントでされるHTMLバージョンをします。これはHTML5です。 については、 doctypesのトピック をしてください。
<html>	ページをきます。タグのにマークアップはありません </html>。 langは、 ISOコード のはen をしてページのをします。 については、 コンテンツのトピック をしてください。
<head>	メインのブラウザウィンドウにされますが、にメタデータと呼ばれるHTMLドキュメントにするがまれていないヘッドを、きます。また、スタイルシートやスクリプトからのインポートをめることもできます。タグは</head>です。
<meta>	ブラウザにドキュメントにするいくつかのメタデータをえます。 charsetは エンコーディング をします。のHTMLは、ではないが、に UTF-8 をすべきである。 HTMLでは、 <meta>タグはタグをとしません。 については、 メタトピック をしてください。
<title>	ページのタイトル。このタグとタグ </title> のにかれたテキストは、ページのタブまたはブラウザのタイトルバーにされます。
<body>	ユーザーにされるドキュメントの、つまりページのすべてのまたはコンテンツをきます。タグ</body>にはコンテンツをししないでください。
<h1>	ページのレベル1し。 は、 し をしてください。
<p>	テキストのなをします。

1.↑ [HTML5,11.2](#)していない

2.↑ `.htm`は、の[DOS](#)の3のファイルのからされています。

オンラインでHTMLをいめるをむ <https://riptutorial.com/ja/html/topic/217/html>をいめる

2: CSSでのHTMLの

き

CSSはページのHTMLにスタイルをします。インラインスタイルは、タグでstyleをすることをい、におめです。スタイルシートは<style>タグをし、ページののをするためにされます。のスタイルシートはCSSのファイルをけり、にルールをする<link>タグをしてできます。このトピックでは、3つのファイルのすべてのについてします。

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

Examples

スタイルシートの

ドキュメントのhead linkをする

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

また、ウェブサイトからされたスタイルシートを、コンテンツネットワーク、つまりCDNをとってすることもできます。たとえば、ブートストラップ

```
<head>
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
integrity="sha384-BVYiisIFeK1dGmJRAkycuHAHRg320mUcww7on3RYdg4Va+PmSTsz/K68vbdEjh4u"
crossorigin="anonymous">
</head>
```

に、WebサイトでフレームワークのCDNサポートをつけることができます。

スタイルシート

また、<style>タグをしてCSSをにめることもできます。

```
<head>
  <style type="text/css">
    body {
      background-color: gray;
    }
  </style>
</head>
```

のスタイルシートもプログラムにめることができます。

```
<head>
  <style type="text/css">
    body {
      background-color: gray;
    }
  </style>

  <style type="text/css">
    p {
      background-color: blue;
    }
  </style>
</head>
```

インラインスタイル

`style`をすると、のスタイルすることができます。

```
<span style="color: red">This text will appear in red.</span>
```

これをけるようにしてください - CSSのポイントは、コンテンツをプレゼンテーションからすることです。

のスタイルシート

のスタイルシートをみむことはです

```
<head>
  <link rel="stylesheet" type="text/css" href="general.css">
  <link rel="stylesheet" type="text/css" href="specific.css">
</head>
```

のファイルとは、のファイルとをきすることにしてください。したがって、`general.css`がまれている、

```
body {
  background-color: red;
}
```

`specific.css`はがまれます

```
body {
  background-color: blue;
}
```

をすると、ドキュメントのがになります。

オンラインでCSSでのHTMLのをむ <https://riptutorial.com/ja/html/topic/4536/cssでのhtmlの>

3: Div

き

HTMLのdivは、のをカプセルするコンテナであり、Webページのをグループしてるためにできます。divはにもししていませんが、Webデザインのなツールです。このトピックでは、divのとアプリケーションについてします。

- `<div>example div</div>`

Examples

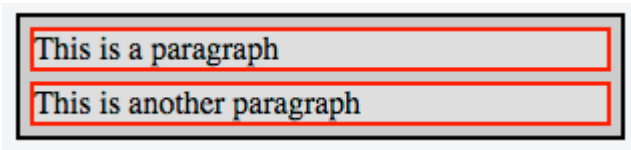
ネスティング

の<div>をの<div>にするのがなです。これは「ネスト」とばれ、をサブセクションにしたり、CSSスタイリングをしてをしたりします。

`<div class="outer-div">`は、2つの`<div class="inner-div">`をグループするためにされます。に`<p>`がまれています。

```
<div class="outer-div">
  <div class="inner-div">
    <p>This is a paragraph</p>
  </div>
  <div class="inner-div">
    <p>This is another paragraph</p>
  </div>
</div>
```

これにより、のがられますかりやすいようにCSSスタイルがされます。



インラインおよびブロックのネストをれにしながら、インラインとブロックがあることにしてください。ブロックは「バックグラウンドでをする」というですが、のネストされたはにのにされ、インラインはデフォルトでにしてされます

い<div>ネストをける

くにされるネストされたコンテナレイアウトは、いコーディングスタイルをします。

をくしたり、のをすると、そのようなHTMLコードがされることがあります。ののブラウザのほとんどのにCSS3のがあります。できるだけないHTMLをして、コンテンツとタグのをめ、ページの

みみをらし、エンジンのランキングをさせてください。

divセクションは6つのレイヤーよりもくれにしていけません。

な

<div>は、なるディビジョンをすのセマンティックなをちません。、HTMLドキュメントののをグループしてカプセルし、のコンテンツグループからします。したがって、<div>はそのによってもよくされます。

```
<div>
  <p>Hello! This is a paragraph.</p>
</div>
```

divは、**ブロックレベルです**。つまり、HTMLのブロックをり、ページのをめることをします。ブラウザには、のデフォルトのCSSルールがあります。

```
div {
  display: block;
}
```

divをのとして、のがでないときには、**World Wide Web ConsortiumW3C**によってくされています。divのわりになをすることで、のアクセシビリティがし、のがします。

たとえば、<article>、<section>をする、<nav>をするページのナビゲーション、および<fieldset>をするフォームコントロールのグループをしてブログをマークアップします。

divは、スタイルのや、のでをけるセクションののをりすのにです。

オンラインでDivをむ <https://riptutorial.com/ja/html/topic/1468/div>

4: Doctypes

き

Doctypes - 'document type'の。ブラウザがよりかりやすくするためにかれたHTMLのバージョンをするのにちます。 DoctypeはHTMLタグではなく、ドキュメントのにします。このトピックでは、さまざまなdoctypeのとをHTMLでします。

- <DOCTYPE [バージョンの]>

<!DOCTYPE>はHTMLタグではありません。ドキュメントがしているHTMLのバージョンをするためにされます。これは、DTDとばれます。

<!DOCTYPE>では、とはされません。 WebページのHTMLがかどうかをするには、 [W3Cのサービス](#) にアクセスしてください。

- なdoctypeがでないり、IEのいバージョンのはHTMLタグをサポートしていません。
- doctypeは、ブラウザがquirksモードをしていないことをするためにされていることがです。
。 [MDNの](#)。

Examples

Doctypeの

<!DOCTYPE>は、HTMLの、 <html>タグののにめるがあります。

5

[HTML 5 Doctype](#)については、 [HTML 5 Doctype](#)をしてください。

```
<!DOCTYPE html>
```

4.01

これらのがどうかについては、 [HTML 4.01 Doctype](#)をしてください。

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

フレームセット

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
"http://www.w3.org/TR/html4/frameset.dtd">
```

HTML 4.01のDoctype

HTML 4.01には、さまざまなのをでできるようにする、いくつかのなるタイプのdoctypeがされています。

HTML 4.01

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

すべてのHTMLとがまれますが、プレゼンテーションまたはのはまれず、フレームセットはされません。

HTML 4.01 Transitional

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

すべてのHTMLと、プレゼンテーションおよびのがまれますが、フレームセットはされません。

HTML 4.01フレームセット

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN"
"http://www.w3.org/TR/html4/frameset.dtd">
```

すべてのHTMLと、プレゼンテーションおよびをみます。フレームセットはされます。

HTML 5 Doctype

HTML5はSGMLについていないため、DTDへのをとしません。

HTML 5 Doctype

```
<!DOCTYPE html>
```

のなし

W3.org HTML 5 DOCTYPEによる

DOCTYPEは、のコンポーネントでされているがあります。

1. "<!DOCTYPE" にする ASCIIの とをしない。

したがって、のDOCTYPE もです。

```
<!doctype html>
<!dOcTyPe html>
<!DocTYpe html>
```

このSOのでは、にトピックについてします [またはのdoctype](#)

のDoctypes

HTML 3.2

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
```

HTML 3.2は、されているほとんどのブラウザでサポートされています。しかし、HTML 3.2では、スタイルシートのサポートがられており、フレームやなどのHTML 4はサポートされていません。

HTML 2.0

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML//EN">
```

HTML 2.0はブラウザでくサポートされていますが、 、フレーム、およびのほか、よくされるくのプレゼンテーションとがサポートされていません。

[オンラインでDoctypesをむ](#) <https://riptutorial.com/ja/html/topic/806/doctypes>

5: HTML 5キヤツシュ

マニフェストファイルはシンプルなテキストファイルで、ブラウザにキャッシュするものキャッシュしないものを行います。マニフェストファイルのされるファイルは、".appcache"です。マニフェストファイルにはの3つのセクションがあります。

キャッシュ・マニフェスト - このヘッダーのにリストされているファイルは、めてダウンロードされたにキャッシュされます。

NETWORK - このヘッダーのにリストされているファイルはサーバーへのをとし、してキャッシュされません

FALLBACK - このヘッダーのにリストされたファイルは、ページにアクセスできないのページをします。

Examples

Html 5キャッシュの

これはたちのindex.htmlファイルです

```
<!DOCTYPE html>
<html manifest="index.appcache">
<body>
    <p>Content</p>
</body>
</html>
```

のコードでindex.appcacheファイルをします

CACHE MANIFEST
index.html

キャッシュするファイルをきみます。index.htmlをロードしてオフラインモードにし、タブをリロードします。

ここでは、2つのファイルはじフォルダになければなりません

オンラインでHTML 5キャッシュをむ <https://riptutorial.com/ja/html/topic/8024/html-5キャッシュ>

6: HTML イベント

Examples

HTML フォームイベント

HTML フォームのアクションによってトリガされるイベントほぼすべてのHTMLにされますが、フォームでもされます

オンブラ	がフォーカスをつたをきばします
オン・チェンジ	のがされたをします。
oncontextmenu	コンテキストメニューがされたときにされるスクリプト
オンフォーカス	がフォーカスをしたをします。
オン	がユーザーをしたときにされるスクリプト
	がなときにされるスクリプト
onreset	フォームのリセットボタンがクリックされたときにします。
オンサーチ	ユーザーがフィールドにかをくとします<input = "search">の
onselect	でテキストがされたにします。
オンサブミット	フォームがされたときにする

キーボードイベント

onkeydown	ユーザーがキーをしたときにします。
onkeypress	ユーザーがキーをしたときにします。
onkeyup	ユーザーがキーをしたときにします。

オンラインでHTMLイベントをむ <https://riptutorial.com/ja/html/topic/10924/htmlイベント>

7: HTMLにJavaScriptコードをめる

- `<script type="text/javascript"> //some code </script>`
- `<script type="text/javascript" src="URL"></script>`
- `<script type="text/javascript" src="URL" async>//async code</script>`

パラメーター

src	JavaScriptファイルへのパスをします。URLまたはURLです。
type	MIMEタイプをします。これはHTML4ではですが、HTML5ではオプションです。
async	スクリプトをですることをしますスクリプトののみ。これはをとしませんXHTMLをく。
defer	ページのがしたときにスクリプトをするようにしますスクリプトののみ。これはをとしませんXHTMLをく。
charset	スクリプトファイルでされるエンコーディングをしますUTF-8。
crossorigin	がをどのようにするか
nonce	コンテンツセキュリティポリシーのチェックにされるノンスCSP3

<http://stackoverflow.com/documentation/javascript/503/document-object-model-dom>をするためにめみJavaScriptコードファイルがされているは、`<script></script>`タグをじる`</body>`タグをするか、JavaScriptメソッドまたはライブラリさまざまなブラウザをするjQueryなどをして、DOMがみまれ、できるようにします。

Examples

のJavaScriptファイルへのリンク

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

srcは、アンカーのhrefと同じようにします。URLまたはURLをできます。のは、HTMLドキュメントのじディレクトリのファイルにリンクしています。これは、HTMLのにある`<head>`タグのにされます

JavaScriptコードをむ

ファイルにリンクするわりに、JSコードをそのままHTMLにめることもできます。

```
<script>
// JavaScript code
</script>
```

にされる**JavaScript** ファイルをむ

```
<script type="text/javascript" src="URL" async></script>
```

Javascriptをにする

セキュリティのにより、クライアントブラウザがJavascriptをサポートしていないか、Javascriptのがになっているがあります。スクリプトがページでされるはずであることをユーザーにえるために、<noscript>タグをできます。<noscript>のは、のページでJavascriptがになっているときににされます。

```
<script>
  document.write("Hello, world!");
</script>
<noscript>This browser does not support Javascript.</noscript>
```

オンラインでHTMLにJavaScriptコードをめるをむ <https://riptutorial.com/ja/html/topic/3719/htmlにjavascriptコードをめる>

8: IFrame

パラメーター

name	のをし、aタグとともにしてiframeのsrcをします。
width	のをピクセルでします。
height	のさをピクセルでします。
src	フレームにされるページをします。
srcdoc	ブラウザがサポートしているとして、フレームにされるコンテンツをします。コンテンツはHTMLでなければなりません。
sandbox	されている、iframeのはのからのものとしてわれ、スクリプト、プラグイン、フォーム、ポップアップなどのはになります。スペースでられたのリストをすることによって、をにすることができます。なについては、のをしてください。
allowfullscreen	iframeのコンテンツにrequestFullscreen()をするかどうかをします。

iframeは、のドキュメントをのHTMLドキュメントにめむためにされます。

iframeをしてすることができます。

- じドメインののHTMLページ。
- のドメインののHTMLページを - じポリシー。
- PDFドキュメントIEにはいくつかのがあるかもしれませんが、[このような](#)はにちます。

ブックマークやナビゲーションにがあるため、iframeをのとしてしなければなりません。にiframeのオプションもあります。[このSOの](#)は、iframeのきみについてもっとするのにちます。

じの

のサイトはiframeをしてすることができません。これは、[じポリシー](#)とばれるポリシーをするためです。つまり、iframeがするサイトは、されるサイトとじドメインにするがあります。

このポリシーは、iFrameのにするコンテンツのにもされます。iFrameがのドメインのコンテンツにアクセスしているは、iFrameのコンテンツにアクセスしたりしたりすることはできません。

sandbox

`sandbox` をすると、`iframe` にながえられます。トークンのスペースリリストは、これらのをするためにできます。

<code>allow-forms</code>	フォームのをします。
<code>allow-pointer-lock</code>	JavaScriptポインタAPIをにします。
<code>allow-popups</code>	ポップアップは、 <code>window.open</code> または <code></code> をしてできます <code></code>
<code>allow-same-origin</code>	<code>iframe</code> ドキュメントは、のものがえられるのではなく、のをします。 <code>allow-scripts</code> とにすると、 <code>iframe</code> ドキュメントは、ドキュメントとじのものであればすべてのサンドボックスをできます。
<code>allow-scripts</code>	スクリプトをにします。 <code>iframe</code> ドキュメントとドキュメントは、 <code>postMessage()</code> APIをしてにできます。 <code>allow-same-origin</code> とにすると、 <code>iframe</code> ドキュメントは、ドキュメントとじからのすべてのサンドボックスをできます。
<code>allow-top-navigation</code>	<code>iframe</code> のコンテンツがトップレベルのドキュメントのをできるようにします。

Examples

インラインフレームの

「IFrame」というのはインラインフレームをします。あなたのページにのページをめることができます。これは、`base.html`なをすさいフレームを`base.html`ます。

```
<iframe src="base.html"></iframe>
```

フレームサイズの

`width`と`height`をしてをすることができます。はピクセルでされますHTML 4.01のパーセンテージはされますが、HTML 5ではCSSピクセルのみがされます。

```
<iframe src="base.html" width="800" height="600"></iframe>
```

IFrameでのアンカーの

、Iframeのウェブページのは、IframeのリンクをクリックするなどしてIframeからされます。ただし、IFrameのコンテンツをIFrameのからすることはです。 `href`がのURLにされ、 `target`がiframeの `name`にされているアンカータグをできます。

```
<iframe src="webpage.html" name="myIframe"></iframe>
<a href="different_webpage.html" target="myIframe">Change the Iframe content to
different_webpage.html</a>
```

"srcdoc"の

HTMLとしてiframeのなをするために、 `srcdoc`を `src`のわりにすることができます。これにより、テキストが「IFrame is cool」のIFrameがされます。

```
<iframe srcdoc="<p>IFrames are cool!</p>"></iframe>
```

`srcdoc`は、ブラウザでサポートされていない、IFrameではなく、バックのにちる `src`を、のは `src`と `srcdoc`がし、ブラウザでサポートされている、 `srcdoc`されます。

```
<iframe srcdoc="<p>IFrames are cool!</p>" src="base.html"></iframe>
```

のでは、ブラウザが `srcdoc`をサポートしていない、わりに `base.html` ページのをします。

サンドボックス

は、すべてのがになっているできないWebページをめみます

```
<iframe sandbox src="http://example.com/"></iframe>
```

ページでスクリプトをしてフォームを `allow-scripts` するには、 `allow-scripts`と `allow-forms` を `sandbox attribute`にし `sandbox attribute`。

```
<iframe sandbox="allow-scripts allow-forms" src="http://example.com/"></iframe>
```

Webページと同じドメインにできないコンテンツユーザーコメントなどがある、iframeをしてスクリプトをにし、ドキュメントがJavaScriptをしてそのコンテンツとやりりできるようにすることができます。

```
<iframe sandbox="allow-same-origin allow-top-navigation"
src="http://example.com/untrusted/comments/page2">
```

ドキュメントは、イベントリスナーをし、そのにわせてIFrameのサイズをできます。これは、 `allow-top-navigation`とともに、サンドボックスされたiframeをドキュメントのとしてさせることができます。

このサンドボックスはサニタイズのわりではありませんが、[な](#)のとしてできます。

また、このサンドボックスは、がiframeのソースをするようにすることによってされるがあることにもしてください。 [Content Security Policy](#) HTTPヘッダーをして、このをすることができます。

オンラインでIFrameをむ <https://riptutorial.com/ja/html/topic/499/iframe>

9: SVG

き

SVGはScalable Vector Graphicsです。 SVGは、Webのグラフィックスをするためにされます。

HTML `<svg>`は、SVGグラフィックスのコンテナです。

SVGには、パス、ボックス、、テキスト、およびグラフィックイメージをするためのいくつかのメソッドがあります。

SVGは、スケーラブルなベクトルをするためのXMLベースです。 HTMLドキュメントにきむことも、のSVGファイルからめむこともできます。 インラインSVGは、それぞれCSSとJavaScriptをしてリビルドとをうことができます。

SVGのブラウザサポートはさまざまですが、 [ここ](#)でできます。

よりなについては、 [SVGのドキュメント](#)をしてください。

Examples

SVG ファイルをHTMLにめむ

`<img>` `<object>`または`<object>`をして、SVGをめむことができます。さとのオプションですが、くおめします。

imageの

```

```

`<img>`をすると、CSSをしてSVGをスタイルしたり、JavaScriptをしてSVGをすることはできません。

オブジェクトの

```
<object type="image/svg+xml" data="attention.svg" width="50" height="50">
```

`<img>`とはなり、`<object>`はSVGをドキュメントにインポートするため、JavascriptとCSSをしてSVGをできます。

インラインSVG

SVGはHTMLにきむことができます。インラインSVGは、CSSとJavaScriptをしてスタイルをしてできます。

```
<body>
  <svg class="attention" xmlns="http://www.w3.org/2000/svg"
xmlns:xlink="http://www.w3.org/1999/xlink" viewBox="0 0 1000 1000" >
    <path id="attention" d="m571,767l0,-106q0,-8,-5,-13t-12,-51-108,0q-7,0,-12,5t-
5,13l0,106q0,8,5,13t12,61l08,0q7,0,12,-6t5,-13Zm-1,-208l10,-257q0,-6,-5,-10q-7,-6,-14,-6l-
122,0q-7,0,-14,6q-5,4,-5,12l9,255q0,5,6,9t13,31l03,0q8,0,13,-3t6,-9Zm-7,-522l428,786q20,35,-
1,70q-10,17,-26,26t-35,101-858,0q-18,0,-35,-10t-26,-26q-21,-35,-1,-70l429,-786q9,-17,26,-
27t36,-10t36,10t27,27Z" />
  </svg>
</body>
```

のインラインSVGは、するCSSクラスをしてスタイルをできます。

```
.attention {
  fill: red;
  width: 50px;
  height: 50px;
}
```

はのようになります。



CSSをしたSVGのめみ

のイメージとに、 `background-image` プロパティをしてSVGファイルをできます。

HTML

```
<div class="attention"></div>
```

CSS

```
.attention {
  background-image: url(attention.svg);
  background-size: 100% 100%;
  width: 50px;
  height: 50px;
}
```

データURLをして、をCSSファイルにめむこともできます。

```
background-image:
url(data:image/svg+xml,%3Csvg%20xmlns%3D%22http%3A%2F%2Fwww.w3.org%2F2000%2Fsvg%22%20xmlns%3Axlink%3D%
106q0%2C-8%2C-5%2C-13t-12%2C-51-108%2C0q-7%2C0%2C-12%2C5t-
5%2C13l0%2C106q0%2C8%2C5%2C13t12%2C61l08%2C0q7%2C0%2C12%2C-6t5%2C-13Zm-1%2C-208l10%2C-
257q0%2C-6%2C-5%2C-10q-7%2C-6%2C-14%2C-6l-122%2C0q-7%2C0%2C-14%2C6q-5%2C4%2C-
```

```
5%2C1219%2C255q0%2C5%2C6%2C9t13%2C31103%2C0q8%2C0%2C13%2C-3t6%2C-9Zm-7%2C-  
5221428%2C786q20%2C35%2C-1%2C70q-10%2C17%2C-26%2C26t-35%2C101-858%2C0q-18%2C0%2C-35%2C-10t-  
26%2C-26q-21%2C-35%2C-1%2C-701429%2C-786q9%2C-17%2C26%2C-27t36%2C-  
10t36%2C10t27%2C27Z%22%20%2F%3E%0D%0A%3C%2Fsvg%3E);
```

オンラインでSVGをむ <https://riptutorial.com/ja/html/topic/1183/svg>

10: アリア

- アリアライブ
- アリア
- アリアオートコンブリート
- アリアチェック
- アリア
- アリア
- aria-haspopup
- アリア - された
- アリア
- アリアラベル
- アリアレベル
- アリア - マルチライン
- アリア - マルチセレクトタイプ
- アリアオリエンテーション
- アリアプレス
- アリアリードオンリー
- アリアが
- アリア
- アリアソート
- アリアバリューマックス
- アリアバジエットミン
- アリアバリュー
- アリアバリューテキスト
- アリア
- アリア・ビジー
- アリア・ジロフェクト
- アリアドラッグ
- アリア
- アリアコントロール
- aria-describedby
- アリアフロー
- アリアラベルドビー
- アリア
- アリア・ポジシエット
- アリアセサイズ

ARIAは、リッチWebアプリケーションをにするためのです。ARIAにうことで、なスクリーンリーダーなどをもってコンテンツにアクセスするユーザーのアクセシビリティをめることができます。

Examples

ロール = "アラート"

で、はになをむメッセージ。

```
<div role="alert" aria-live="assertive">Your session will expire in 60 seconds.</div>
```

は `role="alert"` と `aria-live="assertive"` `role="alert"` にんでいます。これらはのですが、スクリーンリーダーではどちらかしかサポートしていません。をにすることにより、ライブがどおりにするをにすることができます。

ソース - Heydon Pickering ['なARIAの](#)

role = "alertdialog"

アラートメッセージをむダイアログの。フォーカスはダイアログのにします。

```
<div role="alertdialog">
  <h1>Warning</h1>
  <div role="alert">Your session will expire in 60 seconds.</div>
</div>
```

ロール = "アプリケーション"

Webドキュメントではなく、Webアプリケーションとしてされた。このでは、アプリケーションは2つのをするなです。

```
<div role="application">
  <h1>Calculator</h1>
  <input id="num1" type="text"> + <input id="num2" type="text"> =
  <span id="result"></span>
</div>
```

ロール = ""

ドキュメント、ページ、またはサイトのしたをするコンポジションでされるページのセクション。

デフォルトののARIAセマンティクスとするARIAロールおよび/または `aria-*` アトリビュートをするとはであり、これらのプロパティはすでにブラウザによってされているためされません。

```
<article>
  <h1>My first article</h1>
  <p>Lorem ipsum...</p>
```

```
</article>
```

セマンティックにして `role=article` をしますされていない、です

```
<div role="article">
  <h1>My first article</h1>
  <p>Lorem ipsum...</p>
</div>
```

`role=article` W3C エントリ

role = "banner"

ページのコンテンツではなく、ほとんどがサイトのコンテンツをむ。

```
<div role="banner">
  <h1>My Site</h1>

  <ul>
    <li><a href="/">Home</a></li>
    <li><a href="/about">About</a></li>
    <li><a href="/contact">Contact</a></li>
  </ul>
</div>
```

ロール = "ボタン"

クリックまたはされたときにユーザーがトリガーするアクションをにする。

```
<button role="button">Add</button>
```

ロール = "セル"

のコンテナのセル。

```
<table>
  <thead>
    <!-- etc -->
  </thead>
  <tbody>
    <td role="cell">95</td>
    <td role="cell">14</td>
    <td role="cell">25</td>
  </tbody>
</table>
```

ロール = "チェックボックス"

チェックするには、`true`、`false`、または`mixed`の3つがあります。

```
<p>
  <input type="checkbox" role="checkbox" aria-checked="false">
  I agree to the terms
</p>
```

ロール = "ヘッダー"

のヘッダーをむセル。

```
<table role="grid">
  <thead>
    <tr>
      <th role="columnheader">Day 1</th>
      <th role="columnheader">Day 2</th>
      <th role="columnheader">Day 3</th>
    </tr>
  </thead>
  <tbody>
    <!-- etc -->
  </tbody>
</table>
```

ロール = "コンボボックス"

のプレゼンテーション。ユーザーがにしてオプションをできるテキストボックスにしています。また、してのテキストをしいとしてリストにすることもできます。

```
<input type="text" role="combobox" aria-expanded="false">
```

、JavaScriptをして、みまたはリストのりのをします。

ロール = ""

DOMののレベルでメインコンテンツとになるようにされているが、メインコンテンツからしてもをなさないのサポートセクション。

```
<div role="complementary">
  <h2>More Articles</h2>

  <ul>
    <!-- etc -->
  </ul>
</div>
```

ロール = "contentinfo"

にするをむきなな。

```
<p role="contentinfo">
  Author: Albert Einstein<br>
```

```
Published: August 15, 1940
</p>
```

ロール = ""

やの。

```
<span role="term" aria-labelledby="def1">Love</span>
<span id="def1" role="definition">an intense feeling of deep affection.</span>
```

ロール = "ダイアログ"

ダイアログは、アプリケーションののをして、ユーザーにのやのをすようにされたアプリケーションウィンドウです。

```
<div role="dialog">
  <p>Are you sure?</p>
  <button role="button">Yes</button>
  <button role="button">No</button>
</div>
```

role = "directory"

ななど、グループのメンバーへののリスト。

```
<ul role="directory">
  <li><a href="/chapter-1">Chapter 1</a></li>
  <li><a href="/chapter-2">Chapter 2</a></li>
  <li><a href="/chapter-3">Chapter 3</a></li>
</ul>
```

ロール = "ドキュメント"

ウェブアプリケーションではなく、ドキュメントコンテンツとしてされたをむ。

```
<div role="document">
  <h1>The Life of Albert Einstein</h1>
  <p>Lorem ipsum...</p>
</div>
```

ロール = "フォーム"

ランドマークは、としてフォームをするためにするアイテムとオブジェクトのコレクションをんでいます。

にしいHTMLをして<form>、デフォルトARIAのセマンティクスを`role=form`してロールは、のネイティブのセマンティクスをオーバーライドするするなど、すでにセマンティックあるになをするべきではないとしてされていません。

デフォルトのARIAセマンティクスとするARIAロールおよびまたはaria-*アトリビュートをすることはであり、これらのプロパティはすでにブラウザによってされているためされません。

```
<form action="">
  <fieldset>
    <legend>Login form</legend>
    <div>
      <label for="username">Your username</label>
      <input type="text" id="username" aria-describedby="username-tip" required />
      <div role="tooltip" id="username-tip">Your username is your email address</div>
    </div>
    <div>
      <label for="password">Your password</label>
      <input type="text" id="password" aria-describedby="password-tip" required />
      <div role="tooltip" id="password-tip">Was emailed to you when you signed up</div>
    </div>
  </fieldset>
</form>
```

ではrole=formをしますされない、なもの

```
<div role=form>
  <input type="email" placeholder="Your email address">
  <button>Sign up</button>
</div>
```

ロール= "グリッド"

グリッドはテーブルのようなとにされたのデータのセルをむインタラクティブなコントロールです。

```
<table role="grid">
  <thead>
    <!-- etc -->
  </thead>
  <tbody>
    <!-- etc -->
  </tbody>
</table>
```

ロール= "グリッドセル"

グリッドまたはツリーグリッドのセル。

```
<table role="grid">
  <thead>
    <!-- etc -->
  </thead>
  <tbody>
    <tr>
      <td role="gridcell">17</td>
      <td role="gridcell">64</td>
```

```
<td role="gridcell">18</td>
</tr>
</tbody>
</table>
```

ロール = "グループ"

によってページサマリーまたはにまれないようにされたのユーザーインターフェースオブジェクト。

```
<div role="group">
  <button role="button">Previous</button>
  <button role="button">Next</button>
</div>
```

ロール = "し"

ページのセクションのし。

```
<h1 role="heading">Introduction</h1>
<p>Lorem ipsum...</p>
```

ロール = "img"

イメージをするのコレクションのためのコンテナ。

```
<figure role="img">
  
  <figcaption>This is my cat, Albert.</figcaption>
</figure>
```

ロール = "リンク"

アクティブにすると、ユーザーエージェントがそのリソースにナビゲートするとなるまたはリソースへの。

ほとんどの、 [デフォルトのARIAセマンティクス](#)にするARIAロールおよび/または aria-* をすることはであり、これらのプロパティはブラウザによってすでにされているためされません。

ソース - <https://www.w3.org/TR/html5/dom.html#aria-usage-note>

ロール = "リスト"

リストのグループ。

```
<ul role="list">
```

```
<li role="listitem">One</li>
<li role="listitem">Two</li>
<li role="listitem">Three</li>
</ul>
```

ロール = "リストボックス"

ユーザーがのリストから1つのをできるようにするウィジェット。

```
<ul role="listbox">
  <li>One</li>
  <li>Two</li>
  <li>Three</li>
</ul>
```

、JavaScriptをしてをします。

role = "listitem"

リストまたはディレクトリの。

```
<ul role="list">
  <li role="listitem">One</li>
  <li role="listitem">Two</li>
  <li role="listitem">Three</li>
</ul>
```

ロール = "ログ"

しいがのあるでされ、いがえるライブの。

```
<ul role="log">
  <li>User 1 logged in.</li>
  <li>User 2 logged in.</li>
  <li>User 1 logged out.</li>
</ul>
```

role = "main"

ドキュメントのな

```
<!-- header & nav here -->
<div role="main">
  <p>Lorem ipsum...</p>
</div>
<!-- footer here -->
```

ロール = "マーキー"

でないがにするライブの。

```
<ul role="marquee">
  <li>Dow +0.26%</li>
  <li>Nasdaq +0.54%</li>
  <li>S&amp;P +0.44%</li>
</ul>
```

ロール = ""

をすコンテンツ。

```

```

ロール = "メニュー"

ユーザーにのリストをするウィジェットの。

```
<ul role="menu">
  <li role="menuitem">New</li>
  <li role="menuitem">Open</li>
  <li role="menuitem">Save</li>
  <li role="menuitem">Close</li>
</ul>
```

role = "menubar"

され、はにされるメニューのプレゼンテーション。

```
<ul role="menubar">
  <li role="menuitem">File</li>
  <li role="menuitem">Edit</li>
  <li role="menuitem">View</li>
  <li role="menuitem">Help</li>
</ul>
```

ロール = "menuitem"

メニューまたはメニューバーにまれるのグループのオプション。

```
<ul role="menubar">
  <li role="menuitem">File</li>
  <li role="menuitem">Edit</li>
  <li role="menuitem">View</li>
  <li role="menuitem">Help</li>
</ul>
```

ロール = "menuitemcheckbox"

のある3つの、true、false、またはmixedのチェックなmenuitem。

```
<ul role="menu">
  <li role="menuitem">Console</li>
  <li role="menuitem">Layout</li>
  <li role="menuitemcheckbox" aria-checked="true">Word wrap</li>
</ul>
```

ロール = **"menuitemradio"**

menuitemradioのグループのチェックなmenuitem、そのうちの1つだけがにチェックすることができます。

```
<ul role="menu">
  <li role="menuitemradio" aria-checked="true">Left</li>
  <li role="menuitemradio" aria-checked="false">Center</li>
  <li role="menuitemradio" aria-checked="false">Right</li>
</ul>
```

ロール = **"ナビゲーション"**

ドキュメントやドキュメントをナビゲートするためのナビゲーションのコレクションはリンク。

```
<ul role="navigation">
  <li><a href="/">Home</a></li>
  <li><a href="/about">About</a></li>
  <li><a href="/contact">Contact</a></li>
</ul>
```

ロール = **"メモ"**

コンテンツがであるか、リソースのメインコンテンツのなセクション。

```
<p>Lorem ipsum...</p>
<p>Lorem ipsum...</p>
<p role="note">Lorem ipsum...</p>
```

role = "option"

リストのな。

```
<ul role="listbox">
  <li role="option">Option 1</li>
  <li role="option">Option 2</li>
  <li role="option">Option 3</li>
</ul>
```

ロール = **"プレゼンテーション"**

なネイティブ・ロールのセマンティクスがアクセシビリティAPIにマップされないエレメント。

```
<div style="float:left;">Some content on the left.</div>
<div style="float:right;">Some content on the right</div>
<div role="presentation" style="clear:both;"></div> <!-- Only used to clear floats -->
```

role = "progressbar"

いがかかるタスクのをする。

```
<progress role="progressbar" value="25" max="100">25%</progress>
```

ロール = "ラジオ"

ラジオののグループのチェックな。に1つのみをチェックできます。

```
<div role="radiogroup">
  <input role="radio" type="radio" aria-checked="true"> One<br>
  <input role="radio" type="radio" aria-checked="false"> Two<br>
  <input role="radio" type="radio" aria-checked="false"> Three
</div>
```

ロール = ""

がじるWebページやのきななセクションは、ページのやにめるほどですたとえば、スポーツスポーツイベントのをむページのなど。

```
<div role="region">
  Home team: 4<br>
  Away team: 2
</div>
```

role = "radiogroup"

ラジオボタンのグループ。

```
<div role="radiogroup">
  <input role="radio" type="radio" aria-checked="true"> One<br>
  <input role="radio" type="radio" aria-checked="false"> Two<br>
  <input role="radio" type="radio" aria-checked="false"> Three
</div>
```

ロール = ""

のコンテナのセルの。

```
<table>
  <thead>
    <!-- etc -->
  </thead>
  <tbody>
```

```

    <tr role="row">
      <!-- etc -->
    </tr>
  </tbody>
</table>

```

ロール= "グループ"

グリッドに1つのをむグループ。

```

<table>
  <thead role="rowgroup">
    <!-- etc -->
  </thead>
  <tbody role="rowgroup">
    <!-- etc -->
  </tbody>
</table>

```

ロール= "rowheader"

グリッドののヘッダーをむセル。

```

<table role="grid">
  <thead>
    <!-- etc -->
  </thead>
  <tbody>
    <tr>
      <th role="rowheader">Day 1</th>
      <td>65</td>
    </tr>
    <tr>
      <th role="rowheader">Day 2</th>
      <td>74</td>
    </tr>
  </tbody>
</table>

```

ロール= "スクロールバー"

コンテンツがににされているかどうかにかかわらず、のコンテンツのスクロールをするグラフィカルオブジェクト。

```

<div id="content1">Lorem ipsum...</div>
<div
  role="scrollbar"
  aria-controls="content1"
  aria-orientation="vertical"
  aria-valuemax="100"
  aria-valuemin="0"
  aria-valuenow="25">
  <div class="scrollhandle"></div>
</div>

```

```
</div>
```

ロール = ""

ランドマークで、アイテムとオブジェクトのコレクションをみ、としてをします。

```
<div role="search">
  <input role="searchbox" type="text">
  <button role="button">Search</button>
</div>
```

role = "searchbox"

をするためのテキストボックスの。

```
<div role="search">
  <input role="searchbox" type="text">
  <button role="button">Search</button>
</div>
```

role = "separator"

コンテンツのセクションまたはメニューアイテムのグループをけてするディバイダ。

```
<p>Lorem ipsum...</p>
<hr role="separator">
<p>Lorem ipsum...</p>
```

ロール = "スライダー"

ユーザーがされたのをするユーザー。

```
<div
  role="slider"
  aria-valuemax="100"
  aria-valuemin="0"
  aria-valuenow="25">
  <div class="sliderhandle"></div>
</div>
```

ロール = "スピンボタン"

ユーザがののからすることをするの。

```
<input
  role="spinbutton"
  aria-valuemax="100"
  aria-valuemin="0"
  aria-valuenow="25">
```

```
type="number"
value="25">
```

ロール = "ステータス"

コンテンツがユーザのであるが、ずしもステータスバーとしてされるはないがアラートをするのにはでないコンテナ。

```
<div role="status">Online</div>
```

role = "switch"

チェック/チェックされていないとはに、オン/オフをすチェックボックスの。

```
<select role="switch" aria-checked="false">
  <option>On</option>
  <option selected>Off</option>
</select>
```

ロール = "タブ"

ユーザーにされるタブコンテンツをするためのメカニズムをするグループラベル。

```
<ul role="tablist">
  <li role="tab">Introduction</li>
  <li role="tab">Chapter 1</li>
  <li role="tab">Chapter 2</li>
</ul>
```

ロール = "テーブル"

とにされたデータをむセクション。のは、インタラクティブではないのコンテナをととしています。

```
<table role="table">
  <thead>
    <!-- etc -->
  </thead>
  <tbody>
    <!-- etc -->
  </tbody>
</table>
```

ロール = "タブリスト"

tabpanelへのであるタブのリスト。

```
<ul role="tablist">
```

```

<li role="tab">Introduction</li>
<li role="tab">Chapter 1</li>
<li role="tab">Chapter 2</li>
</ul>

```

role = "tabpanel"

タブにけられたリソースのコンテナ。タブはタブリストにまれます。

```

<ul role="tablist">
  <li role="tab">Introduction</li>
  <li role="tab">Chapter 1</li>
  <li role="tab">Chapter 2</li>
</ul>
<div role="tabpanel">
  <!-- etc -->
</div>

```

ロール = "テキストボックス"

のテキストをそのとしてする。

```

<textarea role="textbox"></textarea>

```

ロール = "タイマー"

からの、またはまでのりをすカウンタをむライブの。

```

<p>
  <span role="timer">60</span> seconds remaining.
</p>

```

ロール = "ツールバー"

コンパクトなビジュアルでされたにされるファンクションボタンのまり。

```

<ul role="toolbar">
  <li></li>
  <li></li>
  <li></li>
  <li></li>
</ul>

```

ロール = "ツールチップ"

のをするコンテキストポップアップ。

```

<span aria-describedby="slopedesc">Slope</span>
<div role="tooltip" id="slopedesc">y=mx+b</div>

```

、ヒントはになります。JavaScriptをすると、ユーザーがしたのにユーザーがしたときに、ツールチップがれてされます。

ロール= "ツリー"

りたたむことができるサブレベルのネストされたグループをむがあるリストのタイプ。

```
<ul role="tree">
  <li role="treeitem">
    Part 1
    <ul>
      <li role="treeitem">Chapter 1</li>
      <li role="treeitem">Chapter 2</li>
      <li role="treeitem">Chapter 3</li>
    </ul>
  </li>
  <li role="treeitem">
    Part 2
    <ul>
      <li role="treeitem">Chapter 4</li>
      <li role="treeitem">Chapter 5</li>
      <li role="treeitem">Chapter 6</li>
    </ul>
  </li>
  <li role="treeitem">
    Part 3
    <ul>
      <li role="treeitem">Chapter 7</li>
      <li role="treeitem">Chapter 8</li>
      <li role="treeitem">Chapter 9</li>
    </ul>
  </li>
</ul>
```

ロール= "ツリーグリッド"

ツリーとじでをしてりたたむことができるグリッド。

role = "treeitem"

ツリーのオプション。ツリーので、ツリーレベルのサブツリーのグループがまれている、またはりたたむことができます。

```
<ul role="tree">
  <li role="treeitem">
    Part 1
    <ul>
      <li role="treeitem">Chapter 1</li>
      <li role="treeitem">Chapter 2</li>
      <li role="treeitem">Chapter 3</li>
    </ul>
  </li>
  <li role="treeitem">
    Part 2
```

```
<ul>
  <li role="treeitem">Chapter 4</li>
  <li role="treeitem">Chapter 5</li>
  <li role="treeitem">Chapter 6</li>
</ul>
</li>
<li role="treeitem">
  Part 3
  <ul>
    <li role="treeitem">Chapter 7</li>
    <li role="treeitem">Chapter 8</li>
    <li role="treeitem">Chapter 9</li>
  </ul>
</li>
</ul>
```

オンラインでARIAをむ <https://riptutorial.com/ja/html/topic/2734/アリア>

11: アンカーとハイパーリンク

き

アンカータグは、々のWebページをリンクするためににされますが、ののドキュメントのなるをリンクするや、アプリケーションをするにもできます。このトピックでは、さまざまなにおけるHTMLアンカータグのとアプリケーションについてします。

- `<a href="URL or anchor">Link Text</a>`

パラメーター

パラメーター	
href	アドレスをします。URLまたはURL、またはアンカーの <code>name</code> にすることができます。URLは、 http://example.com/ のようなウェブサイトのなURLです。URLは、じWebサイトののディレクトリおよび/またはドキュメントをしています。たとえば、 <code>/about-us/</code> はルートディレクトリ <code>/</code> の "about-us"ディレクトリをします。ドキュメントをにせずにのディレクトリをすると、Webサーバーはそのディレクトリにドキュメント "index.html"をします。
hreflang	<code>href</code> でリンクされているリソースのをしますこれはこのリソースとともにするがあります。HTML5では BCP 47 、HTML 4 では RFC 1766 のをします。
rel	のドキュメントとリンクされているドキュメントとのをします。HTML5では、を でするか、Microformats wikiに するがあります 。
target	しいタブやウィンドウなど、リンクをくをします。なは、 <code>_blank</code> 、 <code>_self</code> 、 <code>_parent</code> 、 <code>_top</code> 、および <code>framename</code> 。このようなるいをする事は、ウェブサイトのユーザーのコントロールにするため、おめしません。
title	リンクにするをします。は、カーソルがリンクをするとき、ツールヒントのテキストとしてされることがもいです。このはリンクにされず、ほぼすべてのHTMLタグでできます。
download	ユーザーがハイパーリンクをクリックしたときにターゲットがダウンロードされるようにします。のは、ダウンロードしたファイルのになります。されるにはなく、ブラウザはにしいファイルをし、ファイル、 <code>.pdf</code> などにします。をすると、のファイルがされます。

Examples

のサイトへのリンク

これはなである[<a> nchor](#)の

```
<a href="http://example.com/">Link to example.com</a>
```

これは、`href` ハイパーテキストでされているURL `http://example.com/`のURLにアンカーテキスト "Link to example.com"をつハイパーリンクをします。それはのようになります

[example.comへのリンク](#)

このリンクがWebサイトにつながることをすために、`external` リンクタイプをできます。

```
<a href="http://example.com/" rel="external">example site</a>
```

HTTPのプロトコルをするサイトにリンクすることができます。たとえば、FTPサイトにリンクするには、

```
<a href="ftp://example.com/">This could be a link to a FTP site</a>
```

この、このアンカータグは、ユーザーのブラウザがHTTPHypertext Transfer ProtocolではなくFTPFile Transfer Protocolをしてexample.comするようにしているがなりexample.com。

これはFTPサイトへのリンクになるがあります

しいタブ/ウィンドウでリンクをく

```
<a href="example.com" target="_blank">Text Here</a>
```

`target`は、リンクをくをします。これを`_blank`にすると、ブラウザーにしいタブまたはウィンドウユーザーごとでくようにします。

セキュリティ

する`target="_blank"`オープニングサイトにへのアクセスできます`window.opener`そのページが、そのアクセスしてすることができますJavaScriptをしてオブジェクトを`window.opener.location` ページのを、にサイトをマルウェアやフィッシングするユーザーをリダイレクトします。

コントロールしていないページでこれをするときは、リンクに`rel="noopener"`をして、`window.opener`オブジェクトがリクエストとともにされないようにします。

のところ、Firefoxは`noopener` サポートしていませんので、のをるには`rel="noopener noreferrer"`をするがあります。

アンカーへのリンク

アンカーは、HTMLページののタグにジャンプするためにできます。 `<a>` タグは、 `id` をつのをすことができます。 IDについては、 [クラスとIDにするドキュメントをしてください](#)。 アンカーはにページのサブセクションにジャンプするためにされ、ヘッダータグとともにされます。

くのトピックでページ `page1.html` をしたとします。

```
<h2>First topic</h2>
<p>Content about the first topic</p>
<h2>Second topic</h2>
<p>Content about the second topic</p>
```

のセクションをしたら、ページののにのセクションへのクイックリンクまたはブックマークきのをすることができます。

トピックに `id` をしたは、そのトピックにリンクすることができます

```
<h2 id="Topic1">First topic</h2>
<p>Content about the first topic</p>
<h2 id="Topic2">Second topic</h2>
<p>Content about the second topic</p>
```

これにアンカーをできます

```
<h1>Table of Contents</h1>
<a href="#Topic1">Click to jump to the First Topic</a>
<a href="#Topic2">Click to jump to the Second Topic</a>
```

これらのアンカーは、Webページ `page1.html` にもされています。 したがって、ページとアンカーをすることで、あるページからのページにリンクすることができます。

```
Remember, you can always <a href="page1.html#Topic1">look back in the First Topic</a> for supporting information.
```

JavaScriptをするリンク

に `javascript:` プロトコルをして、のリンクとしてくのではなく、JavaScriptとしてテキストをします。

```
<a href="javascript:myFunction();">Run Code</a>
```

`onclick` をってじことをすることもできます

```
<a href="#" onclick="myFunction(); return false;">Run Code</a>
```

`return false; #` へのリンクがクリックされたときにページがにスクロールしないようにするがあり

ます。そのにしたいすべてのコードをずめてください。りはさらなるコードのをめるでしょう。

またすべきは、をめることができます!ページがにスクロールするのをぐためにハッシュタグのにこれは、なスラッグがあれば、それがする `id="#"`をつけることができなかったため、ページのどこでもスクロールしないリンクがするためです。なスラッグ `#scrollsNowhere`などをしてもしじをすることができます。この、`return false;`ではありません

```
<a href="#" onclick="myFunction();">Run Code</a>
```

あなたはこれをうべきですか

えはノーほぼです。このようなJavaScriptをインラインですることはかなりいことです。ページのをし、わりにをバインドするなJavaScriptソリューションをすることをしてください。 [イベントのリッスン](#)

また、このがリンクではなくボタンであるかどうかをしてください。そのは、`<button>`をするがあります。

じサイトのページにリンクする

[パス](#)をして、じWebサイトのページにリンクすることができます。

```
<a href="/example">Text Here</a>
```

のでは、サーバーのルートディレクトリ `/`にあるファイルの `example`れます。

このリンクが<http://example.com>にある、の2つのリンクがユーザーをじにさせます

```
<a href="/page">Text Here</a>
<a href="http://example.com/page">Text Here</a>
```

のは、`example.com`ルートディレクトリにある `page` ファイルにし `example.com`。

メールクライアントをするリンク

な

`href`のが `mailto:`まる、それはクリックにメールクライアントをこうとします

```
<a href="mailto:example@example.com">Send email</a>
```

これにより、メールアドレス `example@example.com` がしくされたメールのになります。

CCとBCC

ccまたはbcc-recipientsのアドレスをするには、のをします。

```
<a href="mailto:example@example.com?cc=john@example.com&bcc=jane@example.com">Send email</a>
```

と

メールのとをすることもできます

```
<a href="mailto:example@example.com?subject=Example+subject&body=Message+text">Send email</a>
```

これらのはURLエンコードされているがあります。

mailto:をしてリンクをクリックすると、オペレーティングシステムでされているデフォルトのメールクライアントをくか、するクライアントをするようにめられます。のアドレスのにされたすべてのオプションがすべてのメールクライアントでサポートされているわけではありません。

をダイヤルするリンク

hrefのがtel:でまる、デバイスは、クリックするとをダイヤルします。これは、モバイルデバイスやSkypeやFaceTimeなどのソフトウェアをしているコンピュータ/タブレットでし、をかけることができます。

```
<a href="tel:11234567890">Call us</a>
```

ほとんどのデバイスとプログラムは、ダイヤルしようとしているをらかなのでするようにユーザーにします。

オンラインでアンカーとハイパーリンクをむ <https://riptutorial.com/ja/html/topic/254/アンカーとハイパーリンク>

12: イメージ

- `<img src="" alt="">`

パラメーター

パラメーター	
src	のURLをします。
srcset	なれば、ディスプレイ、モニタなどでは、
sizes	ブレイクポイントのサイズ
crossorigin	がをどのようにするか
usemap	するイメージマップの
ismap	イメージがサーバーイメージマップであるかどうか
alt	なんらかのでできないにされるテキスト
width	のをしますオプション。
height	のさをしますオプション。

Examples

イメージの

ページにをするには、`image`タグをします。

イメージタグ `img` にはタグがありません。 `img` タグにえるな2つののは `src`、ソース、 `alt`。これはをするテキストです。

```

```

ウェブURLからをすることもできます

```

```

はにHTMLページにされず、はHTMLページにリンクされます。 `<img>` タグは、のをします。

base64をしてページにをめむこともです

```

```

ヒントイメージをのドキュメントにリンクするには、に<a>タグにタグをネストします。

のとさ

widthと**height**は、ではされていません。らのはずっとしくなっています。

イメージのサイズは、イメージタグのwidthとheightをしてできます。

```

```

のwidthとheightをすることで、のサイズをしていても、ページのにするヒントがブラウザにされます。イメージのサイズがされていないは、イメージがロードされたにブラウザがページのレイアウトをするがあり、ロードにページが「ジャンプ」するがあります。

4.1

イメージにCSSピクセルまたはイメージののパーセンテージのとさをえることができます。

これらのはすべてです。

```




```

5

イメージのとさはCSSピクセルでするがあります。パーセンテージはもはやなではありません。に、のがされているは、アスペクトをする3つののいずれかにしなければなりません。ですが、をよりきなサイズにばすためにwidthとheightをししないでください。

これらのはです

```



```

このはおめできません

```

```

これらのはです

```

```

```

```

テキストの

Alt-Textは、のあるユーザーやエンジンでスクリーンリーダーがします。そのため、あなたのになaltテキストをくことができます。

イメージをaltできえても、テキストはしくされます。えば

```
<!-- Incorrect -->
 An anonymous user wrote:
<blockquote>Lorem ipsum dolor sed.</blockquote>
<a href="https://google.com/"></a> /
<a href="https://google.com/"></a>
```

イメージがなければ、これはのようになります。

ユーザーavatarユーザーがいた

Lorem ipsum dolor sed。

アイコンの / アイコン

これをするには

- アバターのaltテキストをします。このは、きるユーザーであることをすになアイコンのをしますが、このはにテキストです。 1
- アイコンのalt-textから「icon」をします。これがアイコンであることがかっていれば、そののをえることはできません。

```
<!-- Correct -->
 An anonymous user wrote:
<blockquote>Lorem ipsum dolor sed.</blockquote>
<a href="https://google.com/"></a> /
<a href="https://google.com/"></a>
```

ユーザーがいた

Lorem ipsum dolor sed。

/

1のaltをめることとそれをにすることのにないがあります。のaltは、イメージがコンテンツのなではないことをしますこのはであり、それはりのをするがないなるイメージです。したがって、レンダリングからすることができます。しかし、altがないことは、イメージがコンテンツのなであり、レンダリングにテキストがしないことをします。

srcsetをしたイメージ

サイズのあるsrcsetの

```

```

sizesはメディアクエリーのようなもので、がビューポートにどれくらいのスペースをしているかをします。

- ビューポートが1200pxよりきい、はに580pxですたとえば、コンテンツはコンテナで1200pxです。のがマージンをいた。
- ビューポートが640pxと1200pxのにある、はビューポートの48をめますたとえば、のスケールはページで、ビューポートのからマージンをいたのをとります。
- ビューポートがのサイズ640pxの、はビューポートの98をめますたとえば、のスケールはページで、ビューポートのからをいたです。のについては、メディアをするがあります。

srcsetはに々がなとそのサイズをブラウザにえています。

- img/hello-300.jpgはimg/hello-300.jpg、
- img/hello-600.jpgはimg/hello-600.jpgですが、
- img/hello-900.jpgは900pxです
- img/hello-1200.jpgは1200ピクセルです

srcはにのソースです。srcsetをする、ブラウザがsrcsetサポートしていない、srcはをsrcsetます。

サイズのないsrcsetの

```

```

srcsetは、なイメージのリストを、デバイスピクセル_x descriptorでします。

- デバイスピクセルが1のは、img/hello-300.jpgしてimg/hello-300.jpg
- デバイスのピクセルが2のは、img/hello-600.jpgしてimg/hello-600.jpg
- デバイスのピクセルが3のは、img/hello-1200.jpgしてimg/hello-1200.jpg

srcはにのソースです。srcsetをする、ブラウザがsrcsetサポートしていない、srcはをsrcsetます

。

をいた

コード

```
<picture>
  <source media="(min-width: 600px)" srcset="large_image.jpg">
  <source media="(min-width: 450px)" srcset="small_image.jpg">
  
</picture>
```

なるでなるをするには、のにすように、タグにソースタグをしてすべてのをめるがあります。

- のが600ピクセルをえるでは、large_image.jpgがされます
- > 450ピクセルのでは、small_image.jpgがされます
- ののでは、default_image.jpgがされます

オンラインでイメージをむ <https://riptutorial.com/ja/html/topic/587/イメージ>

13: イメージマップ

- `<img usemap="#[map-name]">`
- `<map name="[map-name]"></map>`
- `<area>`

パラメーター

タグ	
	に、 <code></code> するイメージマップのをします。の <code></code> がされます。
usemap	マップの <code>name</code> にハッシュがされています。たとえば、 <code>name="map"</code> 、イメージには <code>usemap="#map"</code> がです。
<map>	
name	それをするマップの。イメージの <code>usemap</code> です。
<area>	は <code><area></code> のです。とき <code>href</code> されている、る <code><area></code> リンク、 <code><area></code> また、アンカータグのすべてのをサポート <a> く ping。それらを MDN のでてください。
alt	イメージがサポートされていないにするテキスト。これは <code>href</code> が <code><area></code> もされているにのみです。
coords	なをする。 <code>shape="polygon"</code> 、コンマでられた "x、y"ペアのリストにするがあります <code>shape="polygon" coords="x1, y1, x2, y2, x3, y3, ..."</code> 。 <code>shape="rectangle"</code> 、 <code>left</code> , <code>top</code> , <code>right</code> , <code>bottom</code> するがあります。 <code>shape="circle"</code> は、 <code>centerX</code> , <code>centerY</code> , <code>radius</code> するがあります。
href	ハイパーリンクのURLされている。これをする、 <code><area></code> はハイパーリンクをしません。
shape	<code><area></code> の。 することができる <code>default</code> する <code>coords</code> なし、 <code>circle</code> は <code>circrectangle</code> または <code>rect</code> のため、 <code>shape="polygon"</code> は <code>poly</code> コーナーポイントによってされたのについて。

- のパラメータリストは、MDNドキュメント `<map>`と `<area>`からされています。
- GIMPなどのをするエディタをして、イメージマップのをよりなのなどですることがです。しかし、それはのような、イメージマップ・ジェネレータをすることはにになるかもしれない、[この1](#)。

Examples

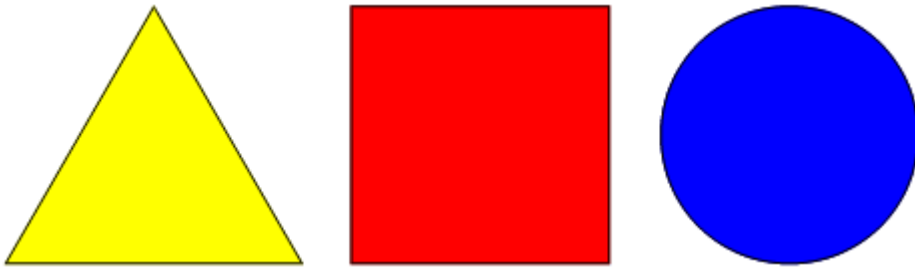
イメージマップの

イメージマップは、クリックながらはハイパーリンクとしてするイメージです。

イメージはタグでされ、マップは<map>タグで<area>タグでされ、クリックをします。 usemap とname をして、イメージとマップをバインドします。

な

ののがクリックになるようにマップをするには

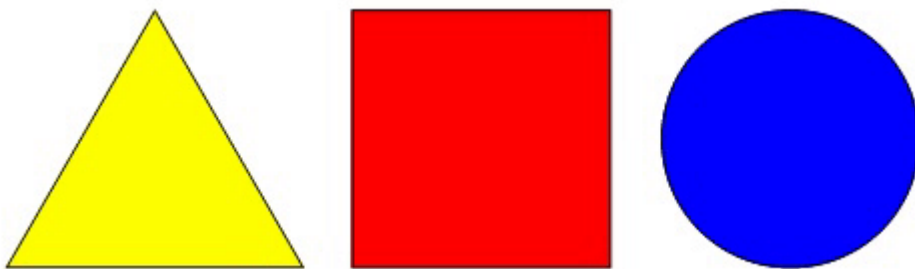


コードはのようになります。

```

<map name="shapes">
  <area shape="polygon" coords="79,6,5,134,153,134">
  <area shape="rectangle" coords="177,6,306,134">
  <area shape="circle" coords="397,71,65">
</map>
```

カーソルがポインタになったときに、ブラウザがをしていることがわかります。 JSFiddleの[ライブ・デモ](#)をいただくか、のデモンストレーションをご覧ください



オンラインでイメージマップをむ <https://riptutorial.com/ja/html/topic/3819/イメージマップ>

14: キャンバス

パラメーター

さ	キャンバスのさをします。
	キャンバスのをします。

- このタグは、Internet Explorerのバージョンが9のにはありません。ブラウザのについては、caniuse.comをしてください。
- `canvas`はグラフィックスのコンテナにぎず、グラフィックスののはJavaScriptによってわれます。

Examples

な

`canvas`はHTML5でグラフィックをするためにされました。

```
<canvas id="myCanvas">
  Cannot display graphic. Canvas is not supported by your browser (IE<9)
</canvas>
```

のでは、300×150ピクセルのなHTML `<canvas>`がされます。

キャンバスをして、、グラフ、、**JavaScript**などのなゲームのなどのらしいものをできます。

`canvas`の2D レイヤーサーフェスオブジェクトは、`CanvasRenderingContext2D`とばれ

`CanvasRenderingContext2D`。または`.getContext("2d")`メソッドをして`.getContext("2d")`から`HTMLCanvasElement`します。

```
var ctx = document.getElementById("myCanvas").getContext("2d");
// now we can refer to the canvas's 2D layer context using `ctx`

ctx.fillStyle = "#f00";
ctx.fillRect(0, 0, ctx.canvas.width, ctx.canvas.height); // x, y, width, height

ctx.fillStyle = "#000";
ctx.fillText("My red canvas with some black text", 24, 32); // text, x, y
```

[jsFiddle](#)の

2つのを

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <title>Draw two rectangles on the canvas</title>
  <style>
    canvas{
      border:1px solid gray;
    }
  </style>
  <script async>
    window.onload = init; // call init() once the window is completely loaded
    function init(){
      // #1 - get reference to <canvas> element
      var canvas = document.querySelector('canvas');

      // #2 - get reference to the drawing context and drawing API
      var ctx = canvas.getContext('2d');

      // #3 - all fill operations are now in red
      ctx.fillStyle = 'red';

      // #4 - fill a 100x100 rectangle at x=0,y=0
      ctx.fillRect(0,0,100,100);

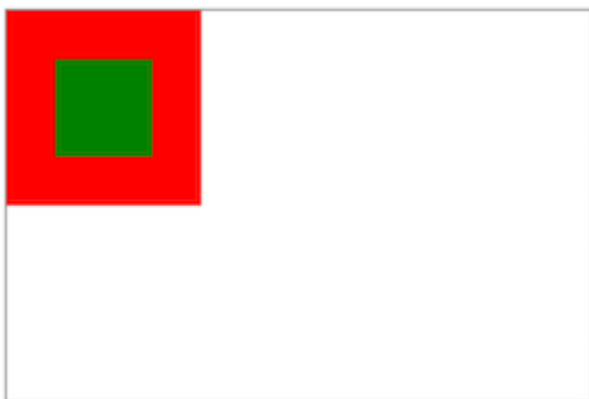
      // #5 - all fill operations are now in green
      ctx.fillStyle = 'green';

      // #6 - fill a 50x50 rectangle at x=25,y=25
      ctx.fillRect(25,25,50,50);

    }
  </script>
</head>
<body>
  <canvas width=300 height=200>Your browser does not support canvas.</canvas>
</body>
</html>

```

このはのようになります。



オンラインでキャンバスをむ <https://riptutorial.com/ja/html/topic/1162/キャンバス>

15: クラスとID

き

クラスとIDにより、スクリプトやスタイルシートのHTMLをしやすくなります。classは、1つのタグででき、CSSによってスタイリングにされます。IDはのをすることをしています。つまり、じIDをして2しないでください。IDはにJavaScriptとリンクでされ、CSSではされません。このトピックでは、HTMLのクラスとIDのなにをするなとをします。

- class = "class1 class2 class3"
- id = "uniqueid"

パラメーター

パラメータ	
クラス	のクラスをします。
id	のIDをしますじコンテキストです

- classとidはどちらもグローバルであるため、のHTMLにりてることができます。
- クラスはアルファベットAZまたはazでまり、そのに、ハイフン、アンダースコアをけることができます。
- HTML5では、classとidをのでできます。HTML 4.0.1では、<base>、<head>、<html>、<meta>、<param>、<script>、<style>、<title>タグはできませんでした。
- は1つのクラスをつことができます。クラスはスペースでられており、スペースをむことはできません。
- はIDを1つしかたず、そのコンテキストWebページででなければなりません。IDにはスペースをめることもできません。

Examples

にクラスをえる

クラスは、りてられているのです。クラスにをりてるには、classをします。

```
<div class="example-class"></div>
```

にのクラスをりてるには、クラスをスペースでります。

```
<div class="class1 class2"></div>
```

CSSでのクラスの

クラスは、そのようなのをすべてをすることなく、のをスタイリングするためにできます。たとえば、これらの2つの`span`はになるスタイリングをつことができます。

```
<span></span>
<span class="special"></span>
```

じのクラスは、ページのののにえることができ、それらはすべてそのクラスにけられたスタイリングをけります。CSSでをしないう、これはにです。

たとえば、2つのがあり、いずれも`highlight`クラスです。

```
<div class="highlight">Lorem ipsum</div>
<span class="highlight">Lorem ipsum</span>
```

CSSがのような、ののテキストにがされます。

```
.highlight { color: green; }
```

しかし、クラスの`highlight`で`div`をターゲットにしたいは、のようなをすることができます

```
div.highlight { color: green; }
```

それにもかかわらず、CSSでスタイリングするときは、クラスをつ `div.highlight` ではなく、クラス `.highlight` のみをするをおめします。

のセレクトとに、クラスはネストすることができます

```
.main .highlight { color: red; } /* Descendant combinator */
.footer > .highlight { color: blue; } /* Child combinator */
```

クラスセクタをさせて、のクラスのみわせをつだけをするともできます。たとえば、これがHTMLの

```
<div class="special left menu">This text will be pink</div>
```

そして、こののピンクのピンクにをけるために、CSSでのことをうことができます

```
.special.left.menu { color: pink; }
```

にIDをえる

のIDは、ででなければなりません。そのは、リンクするアンカーをする、スクリプトをする、またはスタイリングCSSをするのにをにすることです。

```
<div id="example-id"></div>
```

が2つのなるのにけられていても、じにじIDをつ2つのをつべきではありません。たとえば、のコードはしくありません。

```
<div id="example-id"></div>
<span id="example-id"></span>
```

ブラウザはこのコードをレンダリングするためにをくしますが、CSSでスタイリングするかJavaScriptでをするとしないがすることがあります。

CSSでIDでをするには、IDのに#ます。

```
#example-id { color: green; }
```

されたページのIDをつにジャンプするには、URLにをけて#をします。

```
http://example.com/about#example-id
```

これはほとんどのブラウザでサポートされており、JavaScriptやCSSををするはありません。

したIDにする

じIDをつのをつことは、トラブルシューティングがしいです。HTMLパーサーは、どんなでもページのレンダリングをみます。、エラーはしません。しかし、そのペースは、ったふるまいのWebページにわるがあります。

このでは

```
<div id="aDiv">a</div>
<div id="aDiv">b</div>
```

CSSセレクトはまだします

```
#aDiv {
  color: red;
}
```

しかし、JavaScriptはのをできません。

```
var html = document.getElementById("aDiv").innerHTML;
```

この、htmlはのdivコンテンツ("a")ます。

IDの

`id`のにするのは`id`とおりです。

1. ドキュメントでなければなりません
2. をむことはできません
3. なくとも1つのをむがあります

したがって、はすべての、ちょうど1、ちょうど、をむことができます。ちょうど。

したがって、これらはです

```
<div id="container"> ... </div>
<div id="999"> ... </div>
<div id="#%LV-||"> ... </div>
<div id="____V"> ... </div>
<div id="□□"> ... </div>
<div id="♥">... </div>
<div id="{ }">... </div>
<div id="©"> ... </div>
<div id="♠♥♦♣~¥"> ... </div>
```

これはです

```
<div id=" "> ... </div>
```

じにまれている、これもです。

```
<div id="results"> ... </div>
<div id="results"> ... </div>
```

4.01

`id`はでまるがあります。のにくことができるのは、

- AZ / az
- 09
- ハイフン "-"
- アンダースコア _
- コロン ":"
- ピリオド "."

のHTML5のセクションののグループのをすると、なものは1つだけです。

```
<div id="container"> ... </div>
```

これらもです

```
<div id="sampletext"> ... </div>
<div id="sample-text"> ... </div>
<div id="sample_text"> ... </div>
<div id="sample:text"> ... </div>
```

```
<div id="sample.text"> ... </div>
```

りしますが、またはでまらないはです。

クラスのために

クラスのはにidとじです。いは、classがドキュメントであるはないということです。

のをすると、これはじではではありませんが、

```
<div id="results"> ... </div>
<div id="results"> ... </div>
```

これはにです

```
<div class="results"> ... </div>
<div class="results"> ... </div>
```

なIDとクラスのがHTMLのでどのようにわれるか

のルールとは、HTMLのコンテキストでされることにしてください。

idやclassに、やをすると、CSS、JavaScript、などのコンテキストでがするがあります。

たとえば、のidはHTML5ですが、

```
<div id="9lions"> ... </div>
```

...それはCSSです

4.1.3 とケース

CSSでは、セレクトアの、クラス、およびIDをむには、[a-zA-Z0-9]およびISO 10646U + 00A0、ハイフン - およびアンダースコア _; **2**のハイフン、または**1**のハイフンですることはできません。 がえられた

ほとんどの、やなをつでをエスケープすることができます。

W3Cの

- [3.2.5.1 id](#)
- [3.2.5.7 class](#)
- [6.2 SGMLの](#)

オンラインでクラスとIDをむ <https://riptutorial.com/ja/html/topic/586/クラスとid>

16: グローバルアトリビュート

パラメーター

class	のクラスを1つします。 クラスとID をしてください。
contenteditable	のをできるかどうかをします。
contextmenu	ユーザーがをクリックしたときにされるコンテキストメニューをします。
dir	のテキストのテキストをします。
draggable	をドラッグできるかどうかをします。
hidden	ページでされていないのをにします。
id	のをします。 クラスとID をしてください。
lang	のとそのテキストのをします。 コンテンツ をしてください。
spellcheck	のをスペル/でチェックするかどうかをします。
style	のインラインCSSスタイルのセットをします。
tabindex	タブのキーボードショートカットによって、ページのがナビゲートされるをします。
title	にするをします。には、マウスオーバーのツールチップテキストののです。
translate	のコンテンツをするかどうかをします。

グローバルアトリビュートは、ドキュメントのどのエレメントにもなアトリビュートです。

Examples

Contenteditable

```
<p contenteditable>This is an editable paragraph.</p>
```

をクリックすると、そのはテキストフィールドとにできます。

`contenteditable`がにされていない、はからそれをします。したがって、コンテンツのなのすべてのテキストもですが、のテキストのはにすることができます。

```
<p contenteditable>
  This is an editable paragraph.
  <span contenteditable="false">But not this.</span>
</p>
```

なのないテキストには、からされたテキストカーソルもあります。

オンラインでグローバルアトリビュートをむ <https://riptutorial.com/ja/html/topic/2811/グローバルアトリビュート>

17: コメント

き

のプログラミング、マークアップ、およびマークダウンとに、HTMLのコメントは、ユーザーインターフェイスにをえることなく、のをのにします。ただし、のとはなり、HTMLコメントは、Internet ExplorerのHTMLをするためにできます。このトピックでは、HTMLコメントとそのアプリケーションをするについてします。

- `<!-- Comment text -->`

`<!--`でまり、`-->`わるものはコメントです。コメントにする2つのダッシュ `--` をめることはできません。また、に2つのダッシュでわらなければなりませんつまり、`---`がっています。

コメントはウェブページにはされず、CSSでスタイルすることはできません。ページのは、HTMLにをつけることや、にのコンテンツをすためにすることができます。

ダイナミックまたはインタラクティブなページの、コンテンツをにしてするには、HTMLコメントではなくJavaScriptとCSSをします。

JavaScriptをしてHTMLコメントノードのコンテンツをすることができ、これらのノードをドキュメントからに、、できますが、ページのにはしません。

HTMLコメントはページのソースコードのであるため、りのページとにブラウザにダウンロードされます。ソースコードは、、ウェブブラウザのメニューオプション「ソースの」または「ページソースの」をしてできます。

Examples

コメントの

HTMLコメントをして、やのに、コードののポイントについてノートをすることができます。 `<!--`でまり、`-->`でわることができる

```
<!-- I'm an HTML comment! -->
```

それらはのコンテンツにインラインでみむことができます

```
<h1>This part will be displayed <!-- while this will not be displayed -->.</h1>
```

また、のにまたがってをすることもできます。

```
<!-- This is a multiline HTML comment.  
Whatever is in here will not be rendered by the browser.
```

```
You can "comment out" entire sections of HTML code.  
-->
```

ただし、のようになHTMLタグにすることはできません。

```
<h1 <!-- testAttribute="something" -->>This will not work</h1>
```

これは、なHTMLがさなど<h1 <!-- testAttribute="something" -->ブロックは、のタグとえられるh1シングルにいて、そのにまれるのいくつかのな、と>もしないじ。

XMLまたはSGMLとHTMLをしようとするツールとののために、あなたのコメントのボディは、2つのダッシュをめることはできません--

Internet Explorerのきコメント

きコメントは、なるバージョンのMicrosoft Internet Explorerのコードをカスタマイズするためにできます。たとえば、なるHTMLクラス、スクリプトタグ、またはスタイルシートをすることができます。きコメントは、Internet Explorerバージョン5から9でサポートされています。Internet Explorerののバージョンとしいバージョン、およびIEのすべてのブラウザは、「レベル」とみなされ、きコメントをのHTMLコメントとしています。

レベルし

レベルでされたコメントは、のHTMLコメントのようにえるコンテンツをカプセルすることによってします。IE5から9まではそれをきコメントとしてんで、それにじてコンテンツをまたはします。のブラウザでは、コンテンツはになります。

```
<!--[if IE]>  
    Revealed in IE 5 through 9. Commented out and hidden in all other browsers.  
<![endif]-->  
  
<!--[if lt IE 8]>  
    Revealed only in specified versions of IE 5-9 (here, IE less than 8).  
<![endif]-->  
  
<!--[if !IE]>  
    Revealed in no browsers. Equivalent to a regular HTML comment.  
<![endif]-->  
  
<!--  
    For purposes of comparison, this is a regular HTML comment.  
-->
```

レベルにさらされた

これらはレベルののコメントとはなりません。のコメントにはきコメントのみがまれています。きコメントをサポートしていないブラウザは、にそれらをし、それらののりのコンテンツをします。

```

<!--[if IE]>-->
    The HTML inside this comment is revealed in IE 5-9, and in all other browsers.
<!--<![endif]-->

<!--[if IE 9]>-->
    This is revealed in specified versions of IE 5-9, and in all other browsers.
<!--<![endif]-->

<!--[if !IE]>-->
    This is not revealed in IE 5-9. It's still revealed in other browsers.
<!--<![endif]-->

```

インラインのをコメントアウトする

インラインは`span`や`a`などは、ドキュメントに1つのをみます。マークアップみにくいやしないにするのにいをけるために、をコメントアウトすることができます。

```

<!-- Use an HTML comment to nullify the newline character below: -->
<a href="#">I hope there will be no extra whitespace after this!</a><!--
--><button>Foo</button>

```

インラインのにコメントをれずにしてみると、それらのにスペースが1つあります。には、スペースをピックアップすることがまれる。

コード

```

<!-- Use an HTML comment to nullify the newline character below: -->
<a href="#">I hope there will be no extra whitespace after this!</a><!--
--><button>Foo</button>
<hr>
<!-- Without it, you can notice a small formatting difference: -->
<a href="#">I hope there will be no extra whitespace after this!</a>
<button>Foo</button>

```

[I hope there will be no extra whitespace after this!](#)

[I hope there will be no extra whitespace after this!](#)

オンラインでコメントをむ <https://riptutorial.com/ja/html/topic/468/コメント>

18: コンテンツ

- `<element lang="language_code">` - コードは、[\[ISO 639-1\]](https://en.wikipedia.org/wiki/ISO_639-1) https://en.wikipedia.org/wiki/ISO_639-1 のにするがあります - >

`lang`のは、な**BCP 47** タグまたは がの でなければなりません。

BCP 47 タグは、[IANA サブタグレジストリ](#) にリストされています。

アクセシビリティ

する WCAG 2.0 はのとおりです。

- [3.1.1 ページの](#)
- [3.1.2 の](#)

する WCAG 2.0 のテクニックはのとおりです。

- [H57html のの](#)
- [H58 をしてのをする](#)

Examples

`lang` は、とテキストのをするためにされます。

```
<p lang="en">The content of this element is in English.</p>
```

```
<p lang="en" title="The value of this attribute is also in English.">The content of this element is in English.</p>
```

はされます

```
<div lang="en">
  <p>This element contains English content.</p>
  <p title="This attribute, too.">Same with this element.</p>
</div>
```

のをつ

をきすることができます

```
<p lang="en">This English sentence contains the German word <span lang="de">Hallo</span>.</p>
```

なるでのの

の `lang` を `applet`、`base`、`basefont`、`br`、`frame`、`frameset`、`hr`、`iframe`、`meta`、`param`、`script` HTML 4.0 のかられた をすることで、のを "き" できます

```
<p lang="en" title="An English paragraph">
  <span lang="de" title="A German sentence">Hallo Welt!</span>
</p>
```

`html` にドキュメントのな をすることはい です

```
<html lang="en">
```

ドキュメントにの `lang` がされていないは、すべて つまり、のとのテキストがそのであることをし ます。

ドキュメントにののがまれている、これらののを「きする」の `lang` をするがあります。

URL

ハイパーリンクをする `<a>` および `<area>` `hreflang` をすることはです。それは、リンクされたリソー スのをします。されたはな [BCP 47](#) [1] タグでなければなりません。

```
<p>
  <a href="example.org" hreflang="en">example.org</a> is one of IANA's example domains.
</p>
```

-
1. ↑IETF ネットワーキンググループ IETF、2009、[をすための RFC 5646 タグ](#)

オンラインでコンテンツをむ <https://riptutorial.com/ja/html/topic/737/コンテンツ>

19: コンピュータコードをマークアップする

- `<pre>Formatted text</pre>`
- `<code>Inline Code</code>`

`code`は、「コンピュータがする」 [HTML5](#) など、あらゆるのに使うことができます。

- ソースコード
- マークアップ/プログラミング、などからの
- ファイル

のは、 `var` をできます。

コンピュータの、 `samp` をできます。

ユーザーの、 `kbd` をできます。

Examples

によって

にコンピュータコードたとえば、HTMLのがまれているは、 `code` をしてそれをマークアップします。

```
<p>The <a> element creates a hyperlink.</p>
```

ブロックする

そして

コードの、`<pre>`がなは、 `pre` を `code` とみわけてし `code`。

```
<pre>
  <code>
    x = 42
    if x == 42:
      print "x is ...      ... 42"
  </code>
</pre>
```

`< <`などのように、HTMLでなをつをエスケープしなければならないので、HTMLコードのブロックをするには `<p>This is a paragraph.</p>`、のようになります。

```
<pre>
  <code>
    &lt;p>This is a paragraph.&lt;/p>
```

```
</code>  
</pre>
```

オンラインでコンピュータコードをマークアップするをむ

<https://riptutorial.com/ja/html/topic/1836/コンピュータコードをマークアップする>

20: セクショニング

HTML5では、がセクションとしてリストされていません。

Examples

`<article>`には、ブログ、ユーザーコメント、またはRSSのようなページのコンテキストでできるインタラクティブなウィジェットなどのコンテンツがまれています。

- がれになっている、ノードのはにするがあります。

の `article` とコメント `article` をむブログ `section` は、このようにえるかもしれません。

```
<section>
  <!-- Each individual blog post is an <article> -->
  <article>
    <header>
      <h1>Blog Post</h1>
      <time datetime="2016-03-13">13th March 2016</time>
    </header>

    <p>The article element represents a self contained article or document.</p>
    <p>The section element represents a grouping of content.</p>

    <section>
      <h2>Comments <small>relating to "Blog Post"</small></h2>

      <!-- Related comment is also a self-contained article -->
      <article id="user-comment-1">
        <p>Excellent!</p>
        <footer><p>...</p><time>...</time></footer>
      </article>
    </section>
  </article>

  <!-- ./repeat: <article> -->

</section>

<!-- Content unrelated to the blog or posts should be outside the section. -->
<footer>
  <p>This content should be unrelated to the blog.</p>
</footer>
```

なをける

ページのなヘッダー、フッター、ナビゲーションバーなどをくがなるグループです。 `<article>` `<main>` を `<main>` にきえることはできません。

```
<article>
```

```
<p>This doesn't make sense, this article has no real `context`.</p>
</article>
```

わりに、を`<main>`できえて、これがこのページの`main`コンテンツであることをします。

```
<main>
  <p>I'm the main content, I don't need to belong to an article.</p>
</main>
```

のをするは、のデバイスとHTML5のブラウザでしいとレンダリングのために`<main>` [ARIAルール](#)をしてください。

```
<section role="main">
  <p>This section is the main content of this page.</p>
</section>
```

ノート

- `<main>`のは、`<article>`ではできません。

`<article>`のHTML5をむには、[ここをクリックしてください](#)

メインエレメント

`<main>`には、Webページのメインコンテンツがまれています。このコンテンツは々のページにのもので、サイトののにはしないでください。ヘッダー、フッター、ナビゲーション、ロゴなどのりしコンテンツはのにされます。

- `<main>`は、たった1ページで1だけするがあります。
- `<main>`は、`article`、`aside`、`footer`、`header`または`nav`のとしてめることはできません。

のでは、のブログ およびやコメントなどのをしています。

```
<body>
  <header>
    <nav>...</nav>
  </header>

  <main>
    <h1>Individual Blog Post</h1>
    <p>An introduction for the post.</p>

    <article>
      <h2>References</h2>
      <p>...</p>
    </article>

    <article>
      <h2>Comments</h2> ...
    </article>
  </main>
```

```
<footer>...</footer>
</body>
```

- ブログのは<main>にまれており、これがこのページのメインコンテンツであることをしていますしたがって、ウェブサイトです。
- <header>タグと<footer>タグは、<main>のです。

ノート

HTML5では、<main>はセクションではなく、グループとしてされます。

- [ARIAロール](#) `main` デフォルト、`presentation`

メインコンテンツとしてされることをしたのへの `role="main"` [ARIAロール](#)のは、HTML5をサポートしていないユーザエージェントをし、そうでないにはよりくのコンテキストをすることをおめします。

<main>はデフォルトでメインのをっているため、するはありません。

[<main>のHTML5をむには、ここをクリックしてください](#)

Nav

<nav>はにウェブサイトのメインナビゲーションブロックをむセクションにするためのもので、ウェブページのののアンカーなどやのページへのリンクをむことができます。

インラインアイテム

は、ハイパーリンクのインラインセットをします。

```
<nav>
  <a href="https://google.com">Google</a>
  <a href="https://www.yahoo.com">Yahoo!</a>
  <a href="https://www.bing.com">Bing</a>
</nav>
```

にじてリストをする

コンテンツがアイテムのリストをすは、リストアイテムをしてこれをし、ユーザーエクスペリエンスをさせます。

ここで `role="navigation"` にしてください。

```
<nav role="navigation">
  <ul>
    <li><a href="https://google.com">Google</a></li>
    <li><a href="https://www.yahoo.com">Yahoo!</a></li>
    <li><a href="https://www.bing.com">Bing</a></li>
  </ul>
</nav>
```

なをける

`<footer>`には、サイトののへのリンクリストFAQ、TCなどがあります。この、フッターだけです。リンクを`<footer>` `<nav>`でさらにラップするはありません。

```
<!-- the <nav> is not required in the <footer> -->
<footer>
  <nav>
    <a href="#">...</a>
  </nav>
</footer>

<!-- The footer alone is sufficient -->
<footer>
  <a href="#">...</a>
</footer>
```

ノート

- `<main>`のは、`<nav>`

`role="navigation"><nav>`へのARIAのは、HTML5をサポートしていないユーザエージェントをし、そうしたコンテキストのコンテキストをすることをおめします。

```
<nav role="navigation"><!-- ... --></nav>
```

スクリーンリーダー がサイトをナビゲートできるようにするソフトウェア

スクリーンリーダーのようなユーザーエージェントは、`<nav>`をにじてなっています。

- ページのレンダリングに`<nav>`のをくすることができます
- それはのレンダリングをらせるがあります
- ユーザーのニーズにわせてのでページをさせることができます
のために`<nav>`のテキストリンクをきくする。

`<nav>`のなHTML5をむには、ここをクリックしてください

セクションエレメント

`<section>`は、コンテンツをテーマにグループするためのなセクションをします。すべてのセクシ

ョンは、セクションのとしてしてできるがあり `section`。

- `<article>`の`<section>`をすることも、そのものです。
- すべてのセクションにテーマ このリージョンをするし
- `<section>`をなスタイリングの'コンテナ'としてしないでください。
スタイリングをするコンテナがなは、わりに`<div>`します。

のでは、のチャプターをつ1つのブログをしています。チャプターはセクションテーマごとにグループされたコンテンツのセットで、セクションのしてできますです。

```
<article>
  <header>
    <h2>Blog Post</h2>
  </header>
  <p>An introduction for the post.</p>
  <section>
    <h3>Chapter 1</h3>
    <p>...</p>
  </section>
  <section>
    <h3>Chapter 2</h3>
    <p>...</p>
  </section>
  <section>
    <h3>Comments</h3> ...
  </section>
</article>
```

ノート

は、のをシンジケートすることがにかなっている、 **article**をするがあります。

`<main>`のHTML5をむには、[ここをクリックしてください](#)

ヘッダ

`<header>`は、もいのセクションコンテンツまたはセクションルートコンテンツをします。
`<header>`には、またはナビゲーションののグループがまれています。

ヘッダーはセクショニングコンテンツではありません。それはしいセクションをしません。

```
<header>
  <p>Welcome to...</p>
  <h1>Voidwars!</h1>
</header>
```

このでは、`<article>`に`<header>`ます。

```
<article>
  <header>
    <h1>Flexbox: The definitive guide</h1>
  </header>
  <p>The guide about Flexbox was supposed to be here, but it turned out Wes wasn't a Flexbox expert either.</p>
</article>
```

W3C

フッター

<footer>には、ページのフッターがまれます。

pタグをむ<footer>のをにします。

```
<footer>
  <p>All rights reserved</p>
</footer>
```

オンラインでセクショニングをむ <https://riptutorial.com/ja/html/topic/311/セクショニング>

21: タブインデックス

パラメーター

	はフォーカスですが、シーケンシャルなキーボードではできません
0	はキーボードシーケンシャルナビゲーションでフォーカスですが、はプラットフォームでされています
ポジティブ	は、シーケンシャルなキーボードナビゲーションをしてフォーカスでアクセスでなければなりません。それはなは、によってされますシーケンシャルは、 <code>tabindex</code> するにいます

`tabindex`のは、W3Cセクションごとに32767をえてはならない17.11.1されたデフォルトが-1でない

0、な、または`tabindex`をたないは、キーボードナビゲーションののにのインデックスをつのにくがあります。

Examples

タブけにをする

```
<div tabindex="0">Some button</div>
```

にじて、ネイティブHTML `button`または`a`タグをしてください。

タブけからをする

```
<button tabindex="-1">This button will not be reachable by tab</button>
```

はタブからはされますが、フォーカスはできません。

カスタムタブオーダーをするしない

```
<div tabindex="2">Second</div>
<div tabindex="1">First</div>
```

のは、をそれぞれののタブのにします。のない `tabindex="0"`または`button`や`a`などのネイティブは、のあるのにされます。

ポジティブは、タブのされるるいをし、スクリーンリーダーにっている々をさせるがあるため、おめしません。DOMをべえることでなをりしてみてください。

オンラインでタブインデックスをむ <https://riptutorial.com/ja/html/topic/2594/タブインデックス>

22: データ

- `<element data-custom-name="somevalue">`

パラメーター

らかの のをとしてします。

Examples

データの

HTML5の`data-*`は、HTMLにデータをするなをします。されたデータは、JavaScriptをしてみみまたはできます

```
<div data-submitted="yes" class="user_profile">
  ... some content ...
</div>
```

- データのは`data-*`。つまり、データのは`data-`のにます。このをすると、にアクセスできます。
- のデータ `json` をむは、 `data-*` をしてできます。

いブラウザのサポート

のすべてのブラウザでサポートされているHTML5にデータがされましたが、HTML5よりのいブラウザではデータがされません。

ただし、HTMLでは、ブラウザでされないはしておかなければならず、ブラウザはページのレンダリングにします。

Webはこのをして、HTMLののではないであるをしています。たとえば、`<img>`タグのに`value`がなく、グローバルではないため、`bellow`の`value`はとなされます。

```

```

つまり、いブラウザではデータはサポートされていませんが、ききし、じのJavaScript `setAttribute` メソッドと `getAttribute` メソッドをしておよびできますが、のブラウザでのみサポートされるしい `dataset` プロパティはできません。

オンラインでデータをむ <https://riptutorial.com/ja/html/topic/1182/データ>

23: テーブル

き

HTMLの<table>は、Webのがセルのをつ2テーブルにのデータテキスト、、リンク、のテーブルなどをするをにします。

- <table></table>
- <thead></thead>
- <tbody></tbody>
- <tfoot></tfoot>
- <tr></tr>
- <th></th>
- <td></td>

さまざまなテーブルとそのコンテンツがわけてテーブルモデルをします。<table>は、モデル/データのコンテナです。テーブルには、そのによってえられた、、およびセルがあります。とはグリッドをします。のセルは、ならずそのグリッドをにカバーするがあります。のリストは、テーブルモデルのさまざまなをしています。

- <table> - モデル/データのコンテナ。<table>は、のでのをつデータをします。
- <caption> - のキャプションまたはタイトルのようにfigcaptionれてfigure
- <col> - なし
- <colgroup> - のグループ
- <thead> - テーブルヘッダ1つだけ
- <tbody> - の/も
- <tfoot> - テーブルフッター1つだけ
- <tr> - テーブル
- <th> - ヘッダーセル
- <td> - テーブルデータセル

には、はのデータをするためのものです。スプレッドシートとにをなさせるデータをおよびするとえることができます。

レイアウトにテーブルをすることはおめしません。わりに、display: tableをむレイアウトとにCSSをしdisplay: table。

<table>レイアウトのにしてでにされているなの1つはHTMLメールです。Outlookをむのメールクライアントは、Microsoftがをつた、いレンダリングエンジンにロールバックします。マイクロソフトがIEをOSのにしないためには、Outlookのレンダリングエンジンをトライデントののバージョンににロールバックしました。このロールバックはのWebテクノロジーをサポートしていないため、HTMLメールに<table>ベースのレイアウトをすることは、ブラウザ/プラットフォーム/クライアントのをするのです。

Examples

シンプルテーブル

```
<table>
  <tr>
    <th>Heading 1/Column 1</th>
    <th>Heading 2/Column 2</th>
  </tr>
  <tr>
    <td>Row 1 Data Column 1</td>
    <td>Row 1 Data Column 2</td>
  </tr>
  <tr>
    <td>Row 2 Data Column 1</td>
    <td>Row 2 Data Column 2</td>
  </tr>
</table>
```

これにより、3つの `<tr>` からなる `<table>` がされます。1のヘッダーセル `<th>` と2のコンテンツセル `<td>` です。 `<th>` はのヘッダーであり、 `<td>` はのデータです。あなたは `<td>` または `<th>` にあなたが
むものをくことができます。

し1/し1	し2/し2
1のデータ1	1のデータ2
2データ1	2のデータ2

またはにまたがる

のセルは、 `colspan` および `rowspan` をして、のまたはにまたがることができます。これらのは `<th>` お
よび `<td>` にできます。

```
<table>
  <tr>
    <td>row 1 col 1</td>
    <td>row 1 col 2</td>
    <td>row 1 col 3</td>
  </tr>
  <tr>
    <td colspan="3">This second row spans all three columns</td>
  </tr>
  <tr>
    <td rowspan="2">This cell spans two rows</td>
    <td>row 3 col 2</td>
    <td>row 3 col 3</td>
  </tr>
  <tr>
    <td>row 4 col 2</td>
    <td>row 4 col 3</td>
  </tr>
```

</table>

は

row 1 col 1	row 1 col 2	row 1 col 3
This second row spans all three columns		
This cell spans two rows	row 3 col 2	row 3 col 3
	row 4 col 2	row 4 col 3

とがしているテーブルをしないでください。なHTMLであり、なるWebブラウザでなるがされるためです。

`rowspan` =セルにまたがるをすでない。こののデフォルトは1です ₁。ゼロ ₀ のは、セルがのからテーブルのの `<thead>`、`<tbody>`、または`<tfoot>` までされることをします。

`colspan` =のセルにまたがるカラムをすでない。こののデフォルトは1です ₁。ゼロ ₀ のは、セルがされているグループ`<colgroup>`ののまでセルがされることをします。

thead、tbody、tfoot、captionのあるテーブル

HTMLは、テーブルに`<thead>`、`<tbody>`、`<tfoot>`、および`<caption>`もします。これらは、セマンティックをテーブルにしたり、のCSSスタイリングのためのをしたりするのにです。

1つのページにまらないテーブルをするとき、ほとんどのブラウザはすべてのページで`<thead>`をりします。

らなければならないのがあります。すべてのがりにしないことにしてください。のは、4つのをどのようにするかをしています。

```
<table>
  <caption>Table Title</caption> <!--| caption is the first child of table |-->
  <thead> <!--=====| thead is after caption |-->
    <tr>
      <th>Header content 1</th>
      <th>Header content 2</th>
    </tr>
  </thead>

  <tbody> <!--=====| tbody is after thead |-->
    <tr>
      <td>Body content 1</td>
      <td>Body content 2</td>
    </tr>
  </tbody>

  <tfoot><!--| tfoot can be placed before or after tbody, but not in a group of tbody. |-->
  <!--| Regardless where tfoot is in markup, it's rendered at the bottom. |-->

  <tr>
```

```

        <td>Footer content 1</td>
        <td>Footer content 2</td>
    </tr>
</tfoot>

</table>

```

ののは2されます。のにはスタイルがなく、2のにはbackground-color、color、border*のCSSプロパティがいくつかされています。スタイルはなガイドとしてされ、のトピックのなではありません。

Table Title	
Header content 1	Header content 2
Body content 1	Body content 2
Footer content 1	Footer content 2

Table Title	
Header content 1	Header content 2
Body content 1	Body content 2
Footer content 1	Footer content 2

	スタイルがされます
<caption>	いのテキスト。
<thead>	のにのテキスト。
<tbody>	いのテキスト。
<tfoot>	のにテキスト。
<th>	オレンジのボーダー。
<td>	いボーダー。

グループ

やのグループにスタイルをすることがあります。なのために、をグループすることもできます。これをうには、<colgroup> <col>と<col>をします。

オプションの<colgroup>タグをすると、をグループできます。<colgroup>は、<table>でなければならず、<caption>ので、コンテンツ例えば、<tr>、<thead>、<tbody>などのになければなりません。

```

<table>

```

```

    <colgroup span="2"></colgroup>
    <colgroup span="2"></colgroup>
    ...
</table>

```

オプションの<col>タグをすると、グループをせずに々のまたはのをできます。<col>はオプションですが、するは<colgroup>のになければなりません。

```

<table>
  <colgroup>
    <col id="MySpecialColumn" />
    <col />
  </colgroup>
  <colgroup>
    <col class="CoolColumn" />
    <col class="NeatColumn" span="2" />
  </colgroup>
  ...
</table>

```

<colgroup> <col>と<col>には、のCSSスタイルをできます。

- border
- background
- width
- visibility
- display display: none
 - display: none;にはディスプレイからがされ、がそのようなセルがしないかのようにされます

は、[HTML5のデータ](#)をしてください。

し

thようにのとのしをすのによくされます。

```

<table>
  <thead>
    <tr>
      <td></td>
      <th>Column Heading 1</th>
      <th>Column Heading 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th>Row Heading 1</th>
      <td></td>
      <td></td>
    </tr>
  </tbody>
</table>

```

```

        </tr>
        <tr>
            <th>Row Heading 2</th>
            <td></td>
            <td></td>
        </tr>
    </tbody>
</table>

```

これは、`scope` をしてアクセシビリティをさせることができます。のは、のようにされます。

```

<table>
  <thead>
    <tr>
      <td></td>
      <th scope="col">Column Heading 1</th>
      <th scope="col">Column Heading 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <th scope="row">Row Heading 1</th>
      <td></td>
      <td></td>
    </tr>
    <tr>
      <th scope="row">Row Heading 1</th>
      <td></td>
      <td></td>
    </tr>
  </tbody>
</table>

```

`scope` は、されたとしてられています。つまり、ののセットからをつことができます。このセットにまれるもの

- `col`
- `row`
- `colgroup`
- `rowgroup`
- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/th#attr-scope>
- <https://www.w3.org/TR/WCAG20-TECHS/H63.html>

オンラインでテーブルをむ <https://riptutorial.com/ja/html/topic/274/テーブル>

24: テキストフォーマット

き

ほとんどのHTMLタグはのにされますが、HTMLではテキストのタグをしてテキストののにのテキストスタイルをします。このトピックには、、、き、およびテキストなどのHTMLテキストのがまれています。

- `<abbr>Abbreviation</abbr>`
- `<b>Bold Text</b>`
- `<del>Deleted Text</del>`
- `<em>Emphasized Text</em>`
- `<i>Italic Text</i>`
- `<ins>Inserted Text</ins>`
- `<mark>Marked (or Highlighted) Text</mark>`
- `<s>Stricken Text</s>`
- `<strong>Strong Text</strong>`
- `<sub>Subscript Text</sub>`
- `<sup>Superscript Text</sup>`
- `<u>Underlined Text</u>`

Examples

、

のテキストには、`<strong>`または`<b>`タグをします。

```
<strong>Bold Text Here</strong>
```

または

```
<b>Bold Text Here</b>
```

いはですか。 `<strong>`はのテキストにしてにまたはにであることをすためにされ、`<b>`はそのようなことをし、ににするがあるテキストをします。

`<b>`テキストみげプログラムをするは、そのりののとはなるをいすことはありません。そののをすることなく、をくだけです。ただし、`<strong>`をすると、じプログラムがテキストがらかなのであることをえるために、なるのをすことができます。

イタリックのテキスト

テキストをにするには、`<em>`または`<i>`タグをします。

```
<em>Italicized Text Here</em>
```

または

```
<i>Italicized Text Here</i>
```

いはですか。 `<em>`は、テキストがされるべきをするがあることをすためにされ、 `<i>`にそのののテキストからはずすテキストをします。

あなたはセンテンスのアクションをしたいたとえば、1はイタリックでそれをするることによって、そうかもしれません`<em>` 「あなただけすでにをしますか」

しかし、あなたがにイタリックでいてるやをしていたは、 `<i>` "はでロミオとジュリエットをまなければなりませんでした。

きテキスト

`<u>`はHTML 4ではされていましたが、HTML 5ではがされました - テキストされたテキストをするためです。このようなレンダリングをして、ページのスペルのったテキストをしたり、ののマークをすることができます。

```
<p>This paragraph contains some <u>mispelled</u> text.</p>
```

ハイライト

`<mark>`はHTML5でしく、のテキストを「のコンテキストでののために」マーキングまたはするためにされます。 [1](#)

もなは、ユーザーがクエリをし、がのクエリをしてされたです。

```
<p>Here is some content from an article that contains the <mark>searched query</mark> that we are looking for. Highlighting the text will make it easier for the user to find what they are looking for.</p>
```

Here is some content from an article that contains the **searched query** that we are looking for. Highlighting he text will make it easier for the user to find what they are looking for.

なはのにいテキストですが、これはCSSでできます。

、、またはつけられた

テキストをみとしてマークするには、 `<ins>`タグをし`<ins>`。

```
<ins>New Text</ins>
```

テキストをみとしてマークするには、 `<del>` タグをします。

```
<del>Deleted Text</del>
```

テキストをりすには、 `<s>` タグをします。

```
<s>Struck-through text here</s>
```

きとき

テキストをきまたはきにオフセットするには、 `<sup>` および `<sub>` タグをできます。

きをするには

```
<sup>superscript here</sup>
```

きをするには

```
<sub>subscript here</sub>
```

をとしてマークするには、 `<abbr>` タグをします。

```
<p>I like to write <abbr title="Hypertext Markup Language">HTML</abbr>!</p>
```

する、 `title` は、そのようなのをするためにされます。

オンラインでテキストフォーマットをむ <https://riptutorial.com/ja/html/topic/526/テキストフォーマット>

25: ナビゲーションバー

Examples

ナビゲーションバー

ナビゲーションバーはにリンクのリストなので、ulとliはナビゲーションリンクをむためにされます。

```
<ul>
  <li><a href="#">Home</a></li>
  <li><a href="#">About</a></li>
  <li><a href="#">Contact</a></li>
</ul>
```

HTML5ナビゲーションバー

HTML5のnavをしてナビゲーションバーをするには、navタグのリンクをみます。

```
<nav>
  <a href="#">Home</a>
  <a href="#">About</a>
  <a href="#">Contact</a>
</nav>
```

オンラインでナビゲーションバーをむ <https://riptutorial.com/ja/html/topic/10725/ナビゲーションバー>

26: フォーム

き

をグループしてデータをするために、HTMLはフォームをしてとをカプセルします。これらのフォームは、されたメソッドのデータをサーバーまたはハンドラーがするページにするをいます。このトピックでは、データをしてするにHTMLフォームをするについてします。

- `<form method="post|get" action="somePage.php" target="_blank|_self|_parent|_top|frameName">`

パラメーター

accept-charset	フォームのにされるエンコードをします。
action	フォームのにフォームデータをするをします。
autocomplete	フォームがオートコンプリートをオンにするかオフにするかをします。
enctype	フォームデータをサーバーにするにフォームデータをどのようにエンコードするかをしますmethod = "post"のみ。
method	フォームデータPOSTまたはGETをするときにするHTTPメソッドをします。
name	フォームのをします。
novalidate	にフォームをしなないようにします。
target	フォームをしたにしたをするをします。

`<form>`は、をサーバーにするフォーム `<button>` `<fieldset>` `<input>` `<label>` `<output>` `<select>` `<textarea>` をむセクションをします。タグ `<form>` とタグ `</form>` のがです。

Examples

する

アクション

actionは、フォームがされたときにされるアクションをします。これは、、されたをし、それをするスクリプトにつながります。のままにすると、じファイルにされます

```
<form action="action.php">
```

メソッド

methodは、GETまたはPOSTのいずれかのフォームのHTTPメソッドをするためにされます。

```
<form action="action.php" method="get">
<form action="action.php" method="post">
```

GETメソッドは、にIDやでをけるなど、データをするために、またはクエリをするためにされます。 GETメソッドは、actionでされたURLにフォームデータをします。

```
www.example.com/action.php?firstname=Mickey&lastname=Mouse
```

POSTメソッドは、データをスクリプトにするときにされます。 POSTメソッドは、フォームデータをアクションURLにするのではなく、をしてします。

フォームからデータをしくするには、をするがあります。
としてのフィールドのをしてにそのをしてみましょう

```
<input type="text" name="lastname" value="Mouse">
```

そのの

```
<form action="action.php" method="post" target="_blank" accept-charset="UTF-8"
enctype="application/x-www-form-urlencoded" autocomplete="off" novalidate>

<!-- form elements -->

</form>
```

フォームタグのターゲット

targetは、フォームをしたにけるをするをすまたはキーワードをします。

targetは、ブラウジングコンテキストタブ、ウィンドウ、またはインラインフレームなどのまたはキーワードをします。

ターゲットをつタグから

```
<form target="_blank">
```

_blank	はしいウィンドウまたはタブにされます

—	はじフレームにされますこれがデフォルトです
—	はフレームにされます
—	はウィンドウのにされます
フレーム	はされたiframeにされます

targetは**HTML 4.01**では です。ターゲットは**HTML5**でサポートされています。

フレームとフレームセットは**HTML5**ではサポートされていないため、**_parent**、**_top**、**framename**のはに**iframe**でされます。

ファイルのアップロード

formタグのenctypeをmultipart/form-dataすることにより、やファイルをサーバにアップロード/することができmultipart/form-data。 enctypeは、サーバーにするにフォームデータをどのようにエンコードするかをします。

```
<form method="post" enctype="multipart/form-data" action="upload.php">
  <input type="file" name="pic" />
  <input type="submit" value="Upload" />
</form>
```

いくつかのフィールドをグループする

フォームをするには、いくつかのフィールドをグループしてフォームレイアウトのにてることができます。これは、タグをしてうことができます。ここではそのをします。

フィールドセットについて、LEGEND TEXTタグをしてセットのをすることができます

```
<form>
  <fieldset>
    <legend>1st field set:</legend>
    Field one:<br>
    <input type="text"><br>
    Field two:<br>
    <input type="text"><br>
  </fieldset><br>
  <fieldset>
    <legend>2nd field set:</legend>
    Field three:<br>
    <input type="text"><br>
    Field four:<br>
    <input type="text"><br>
  </fieldset><br>
  <input type="submit" value="Submit">
</form>
```

1st field set:

Field one:

Field two:

2nd field set:

Field three:

Field four:

ブラウザのサポート

Chrome、IE、Edge、FireFox、Safari、Operaのバージョンでもタグをサポートしています

オンラインでフォームをむ <https://riptutorial.com/ja/html/topic/1160/フォーム>

27: マークアップの

`cite`と`blockquote`は、`、`のし、スクリプトの、インスタントメッセージの、およびなるプレイヤーがであるそののをすすべきではありません。

Examples

によって

`q`は、のであるもりにできます。

```
<p>She wrote <q>The answer is 42.</q> and everyone agreed.</p>
```

4.01

はしないでください。ユーザーエージェントはHTML 4.01でrespするがあります。HTML 4.0ではそれらをにレンダリングするがあります。

5

はしないでください。ユーザーエージェントはにそれらをレンダリングします。

ソースURL `cite`

`cite`をして、されたソースのURLをすることができます

```
<p>She wrote <q cite="http://example.com/blog/hello-world">The answer is 42.</q> and everyone agreed.</p>
```

ソースがしている、ハイパーリンクするがありますので、ブラウザはに、このURLをさないことにしてくださいほかにを。 _a

ブロックする

`blockquote`は、ブロックレベルのにできます。

```
<blockquote>
  <p>The answer is 42.</p>
</blockquote>
```

ソースURL `cite`

`cite`をして、されたソースのURLをすることができます

```
<blockquote cite="http://example.com/blog/hello-world">
  <p>The answer is 42.</p>
</blockquote>
```

ソースがしている、ハイパーリンクするがありますので、ブラウザはに、このURLをさないこと
にしてくださいほかにをこのリンクをどこにするかについてのセクションの/をしてください。 ^a

/

4.01

citation / attributionは`blockquote`のであってではありません

```
<blockquote cite="http://example.com/blog/hello-world">
  <p>The answer is 42.</p>
</blockquote>
<p>Source: <cite><a href="http://example.com/blog/hello-world" rel="external">Hello
World</a></cite></p>
```

とをグループする`div`をすることはできますが、それらをにけるはありません。

`cite`は、されたソースのにできますただし、ではできません。

5

citation / attributionえば、ソースURLをえるハイパーリンクは、`blockquote`にれることができます
が、そのは`cite`テキストののまたは`footer`になければなりません。

```
<blockquote cite="http://example.com/blog/hello-world">
  <p>The answer is 42.</p>
  <footer>
    <p>Source: <cite><a href="http://example.com/blog/hello-world" rel="external">Hello
World</a></cite></p>
  </footer>
</blockquote>
```

`cite`は、されたソースの、またはののためにできます。

オンラインでマークアップのをむ <https://riptutorial.com/ja/html/topic/2943/マークアップの>

28: メタ

き

HTMLのメタタグは、キーワード、90のフィールドをむにするなをする。このトピックでは、これらのタグのについてします。

- `<meta name="metadata name" content="value">`
- `<meta http-equiv="pragma directive" content="value">`
- `<meta charset="encoding label">`

メタタグは、HTMLのメタデータをするためのHTMLタグです。メタタグはheadにるがあります。ページにはのメタタグがまれています。

メタタグのkeywordsは、ロボットによってされるものではありません。ほとんどのエンジンは、どのキーワードがウェブページのコンテンツにしているかをします。それはわれて、もあなたはもはやキーワードのメタタグをめるべきではないことをう。

ページのメタデータはにブラウザのスケーリングなどやエンジンGoogle、Yahoo、Bingがするウェブクロールのスパイダーでされます。

このでは、`<meta name>`でするされたメタデータと、`<meta http-equiv>`でするためのされたメタデータプラグマディレクティブをしています。しかし、インターネットWebクローラ、オーサリングツール、ソーシャルシェアリングサービスなどのくのサービスはメタデータのなポイントとして`<meta name>`をします。これらのいくつかは、[のwikiページにされています](#)。

Examples

コード

charsetは、HTMLのエンコーディングをし、なエンコーディング windows-1252、ISO-8859-2、Shift_JIS、UTF-8 であるがあります。UTF-8 Unicodeはもくされており、しいプロジェクトにするがあります。

5

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

```
<meta charset="ISO-8859-1">
```

すべてのブラウザはに`<meta charset>`フォームをしています、らかのでページがなHTML 4.01であるがあるは、わりにのものをできます。

```
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
```

```
<meta http-equiv="content-type" content="text/html; charset=ISO-8859-1">
```

ブラウザができるなエンコードラベルをすべてするには、「[エンコード](#)」もしてください。

リフレッシュ

5ごとにページをするには、この`meta`を`head`にします。

```
<meta http-equiv="refresh" content="5">
```

これはなコマンドですが、ユーザーエクスペリエンスにをえるため、しないことをおめします。ページをあまりににすると、ページがしなくなることがあり、ページのにスクロールすることがよくあります。ページののをにするがあるは、ページのをするだけでよいができます。

モバイルレイアウトコントロール

なモバイルにされたサイトでは、のように`<meta name="viewport">`タグをします。

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

`viewport`は、しているデバイスについて、ページのとスケーリングをするについてブラウザにします。

のでは、`content="width=device-width"`ブラウザは、ののにページのをレンダリングすることをします。そのがあるので、もし480px wide、ブラウザのウィンドウになります480px wide。 `initial-scale=1`は、のズームこのは1で、ズームしないことをしますをします。

このタグがサポートするはのとおりです。

width	デバイスのビューポートの。 1 480のような <code>device-width</code> またはのピクセル
height	デバイスのビューポートのさ。 2 <code>device-height</code> またはピクセルのの 600
initial-scale	ページがみまれるときののズーム。 1.0はズームしません。
minimum-scale	がページをズームできる。 1.0はズームしません。
maximum-scale	がページでズームできる。 1.0はズームしません。
user-scalable	デバイスのズームインとズームアウトをします。は <code>yes</code> または <code>no</code> です。 <code>no</code> にすると、ユーザーはWebページをすることができません。デフォルトは <code>yes</code> です。ブラウザではこのルールをできます。

ノート

1 `width` プロパティは、デバイスの幅をピクセル `width=600` または `device-width` `width=device-width` のいずれかでできます。

2に、`height` プロパティは `pixels` `height=600` または `device-height` `height=device-height` でできます。これはデバイスの高さをします。

ページ

`application-name`

ページがすWebアプリケーションのをします。

```
<meta name="application-name" content="OpenStreetMap">
```

Webアプリケーションでないは、`application-name` メタタグをしないでください。

`author`

ページのをする

```
<meta name="author" content="Your Name">
```

は1つだけです。

`description`

ページのをします。

```
<meta name="description" content="Page Description">
```

`description` メタタグは、さまざまなエンジンでされ、でWebページにインデックスをけることができます。、メタタグにまれるは、エンジンののページ/ウェブサイトのメインタイトルのにされるいです。Googleは、の2025だけをします。

`generator`

```
<meta name="generator" content="HTML Generator 1.42">
```

ドキュメントのにされたソフトウェアパッケージの1つをします。マークアップがにされるページにのみされます。

`keywords`

エンジンのキーワードをするカンマリ

```
<meta name="keywords" content="Keyword1, Keyword2">
```

keywordsメタタグは、エンジンがウェブページにするクエリをするためにされることがあります。として、あまりにもくのをしないことをおめします。このメタタグをけにするほとんどのエンジンは、の20だけをけするためです。もなキーワードをにれてください。

ロボット

いくつかのなエンジンでサポートされているrobotsは、エンジンのスパイダーがページのインデックスをできるかどうか、ページのリンクをたどるかどうかをします。

```
<meta name="robots" content="noindex">
```

このでは、すべてのエンジンにページをにしないようにしています。そののされるはのとおりです。

/	
all	デフォルト。 index, followしindex, follow。 のをしてください。
noindex	ページのインデックスをしないでください。
nofollow	このページのリンクをたどらないでください
follow	ページのリンクをたどることができます。 のをしてください。
none	noindex, nofollowしいnoindex, nofollow。
noarchive	このページのキャッシュされたバージョンをにさせないでください。
nocache	Bingのようないくつかのボットがするnoarchive。
nosnippet	にこのページのスニペットをしないでください。
noodp	のタイトルやスニペットの Open Directoryプロジェクト から、このページのメタデータをしないでください。
notranslate	にこのページのをしないでください。
noimageindex	このページのにはインデックスをしないでください。
unavailable_after [RFC-850 date/time]	したにこのページをにしないでください。 /は、 RFC 850 でするが あり ます 。

にindexやfollowをすることfollow、すべてのエンジンがにされているとしているので、なではありません。robots.txtファイルのみに、エンジンはにされていないものだけをします。エンジンがうことがされていないことだけをべるだけでindex, ..., noindexすべてのエンジンがじようにわれるわけではないindex, ..., noindexつてのものえは、index, ..., noindex

の

iOSなどのモバイルプラットフォームでは、がにされ、tel:リンクにわります。このはにですが、ISBNコードやそののをとしてすることがあります。

モバイルSafariやそのWebKitベースのモバイルブラウザで、のとをにするには、のメタタグがです。

```
<meta name="format-detection" content="telephone=no">
```

ソーシャルメディア

オープングラフは、サイトのマークアップにまれるのをするメタデータのです。これにより、Facebookなどのウェブサイトは、ウェブサイトにするよりくかなをされたですることができます。このは、ユーザーがFacebookのOGメタデータをむウェブサイトへのリンクをするとにされます。

Facebook / Open Graph

```
<meta property="fb:app_id" content="123456789">
<meta property="og:url" content="https://example.com/page.html">
<meta property="og:type" content="website">
<meta property="og:title" content="Content Title">
<meta property="og:image" content="https://example.com/image.jpg">
<meta property="og:description" content="Description Here">
<meta property="og:site_name" content="Site Name">
<meta property="og:locale" content="en_US">
<meta property="article:author" content="">
<!-- Facebook: https://developers.facebook.com/docs/sharing/webmasters#markup -->
<!-- Open Graph: http://ogp.me/ -->
```

- [Facebook Open Graph Markup](#)
- [オープングラフプロトコル](#)

Facebook / インスタント

```
<meta charset="utf-8">
<meta property="op:markup_version" content="v1.0">

<!-- The URL of the web version of your article -->
<link rel="canonical" href="http://example.com/article.html">

<!-- The style to be used for this article -->
<meta property="fb:article_style" content="myarticlestyle">
```

- [Facebookをする](#)
- [インスタントアーティクルフォーマットリファレンス](#)

Twitterはメタデータにのマークアップをします。このメタデータは、ツイートにサイトへのリンクがまれているときのツイートのをするためのとしてされます。

Twitter

```
<meta name="twitter:card" content="summary">
<meta name="twitter:site" content="@site_account">
<meta name="twitter:creator" content="@individual_account">
<meta name="twitter:url" content="https://example.com/page.html">
<meta name="twitter:title" content="Content Title">
<meta name="twitter:description" content="Content description less than 200 characters">
<meta name="twitter:image" content="https://example.com/image.jpg">
```

- [Twitterカードスタートガイド](#)
- [Twitterカードバリデーター](#)

Google+ / Schema.org

```
<link href="https://plus.google.com/+YourPage" rel="publisher">
<meta itemprop="name" content="Content Title">
<meta itemprop="description" content="Content description less than 200 characters">
<meta itemprop="image" content="https://example.com/image.jpg">
```

リダイレクト

ウェブページにリダイレクトがなことがあります。

たとえば、5にexample.comにリダイレクトするには

```
<meta http-equiv="refresh" content="5;url=https://www.example.com/" />
```

これは、されたWebサイトこのは5にexample.comにあなたをります。

リダイレクトののにをするがあるは、のをするだけで;url=はをします。

Webアプリケーション

あなたのウェブアプリまたはウェブサイト、アプリケーションのショートカットアイコンをのホームに**するようし**、AndroidChromeの「**ホームに**」メニューをしてアプリを「**アプリモード**」でさせることができます。

のメタタグは、フルスクリーンモードアドレスバーなしでWebアプリケーションをきます。

Android Chrome

```
<meta name="mobile-web-app-capable" content="yes">
```

IOS

```
<meta name="apple-mobile-web-app-capable" content="yes">
```

メタタグのステータスバーとアドレスバーのをすることもできます。

Android Chrome

```
<meta name="theme-color" content="black">
```

IOS

```
<meta name="apple-mobile-web-app-status-bar-style" content="black">
```

オンラインでメタをむ <https://riptutorial.com/ja/html/topic/1264/メタ>

29: メディア

パラメーター

	のをピクセルでします。
さ	のさをピクセルでします。
<source>	オーディオファイルまたはビデオファイルのリソースをします。
トラック	メディアのテキストトラックをする
コントロール	コントロールをする
	メディアのをにする
ループ	りしのサイクルでメディアをする
ミュートされた	なしでメディアをする
ポスター	ビデオがみまれるまでするをりてます。

ブラウザでのサポート

	クロム	Firefox Gecko	インターネット トエクスペ ローラ	オペラ	サファ リ
なサポート	3.0	3.51.9.1	9.0	10.50	3.1
<audio> WAVEのPCM	はい	3.51.9.1	サポートなし	10.50	3.1
<audio> WebMのVorbis	はい	4.02.0	サポートなし	10.60	3.1
<audio> MSEをしたWebMでの Vorbis / Opusのストリーミン グ		36.0			
<audio> OggのVorbis	はい	3.51.9.1	サポートなし	10.50	サポ ートなし
<audio> MP3	はい	はい	9.0	はい	3.1
<audio>					

	クロム	Firefox Gecko	インターネッ ト 익스プロ ーラ	オペラ	サファ リ
MP4のMP3	はい				
<audio> MP4のAAC	はい	はい	9.0	はい	3.1
<audio> OggのOpus	27.0	15.015.0			
<video> Web8のVP8とVorbis	6.0	4.02.0	9.0	10.60	3.1
<video> Web9のVP9とOpus	29.0	28.028.0	はい		
<video> MSEでWebMをストリーミングする		42.042.0			
<video> OggのTheoraとVorbis	はい	3.51.9.1	サポートなし	10.50	サポートなし
<video> MP4のH.264とMP3	はい	はい	9.0	はい	はい
<video> MP4のH.264とAAC	はい	はい	9.0	はい	3.1
そのの	サポートなし	サポートなし	サポートなし	サポートなし	3.1

Examples

``と``オーディオ/ビデオコンテンツをする

HTMLまたは<audio>をして、ビデオ/オーディオコンテンツをドキュメントにめみます。ビデオ/オーディオは、1つまたはのビデオ/オーディオソースをむ。ソースをするには、srcまたは<source>をします。ブラウザはもしたものをします。

オーディオタグの

```

<!-- Simple video example -->
<video src="videofile.webm" autoplay poster="posterimage.jpg">
  Sorry, your browser doesn't support embedded videos,
  but don't worry, you can <a href="videofile.webm">download it</a>
  and watch it with your favorite video player!
</video>

<!-- Video with subtitles -->
<video src="foo.webm">
  <track kind="subtitles" src="foo.en.vtt" srclang="en" label="English">
  <track kind="subtitles" src="foo.sv.vtt" srclang="sv" label="Svenska">
</video>

```

```
<!-- Simple video example -->
<video width="480" controls poster="https://archive.org/download/WebmVp8Vorbis/webmvp8.gif" >
  <source src="https://archive.org/download/WebmVp8Vorbis/webmvp8.webm" type="video/webm">
  <source src="https://archive.org/download/WebmVp8Vorbis/webmvp8_512kb.mp4" type="video/mp4">
  <source src="https://archive.org/download/WebmVp8Vorbis/webmvp8.ogv" type="video/ogg">
  Your browser doesn't support HTML5 video tag.
</video>
```

オーディオタグの

```
<!-- Simple audio playback -->
<audio src="http://developer.mozilla.org/@api/deki/files/2926/=AudioTest_(1).ogg" autoplay>
  Your browser does not support the <code>audio</code> element.
</audio>

<!-- Audio playback with captions -->
<audio src="foo.ogg">
  <track kind="captions" src="foo.en.vtt" srclang="en" label="English">
  <track kind="captions" src="foo.sv.vtt" srclang="sv" label="Svenska">
</audio>
```

オーディオ

HTML5は、Webページにオーディオファイルをめむためのしいをします。

`<audio>`をして、ページにオーディオファイルをめむことができます。

```
<audio controls>
  <source src="file.mp3" type="audio/mpeg">
  Your browser does not support the audio element.
</audio>
```

ビデオ

`<video>`をして、Webページにビデオをめむこともできます。

```
<video width="500" height="700" controls>
  <source src="video.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>
```

ビデオヘッダーまたは

ループでされ、コントロールやサウンドがないビデオをする。ビデオヘッダーやにです。

```
<video width="1280" height="720" autoplay muted loop poster="video.jpg" id="videobg">
  <source src="video.mp4" type="video/mp4">
  <source src="video.webm" type="video/webm">
  <source src="video.ogv" type="video/ogg">
</video>
```

このCSSは、ビデオをみめないのフォールバックをします。 videoののフレームをpost video.jpg としてすることをおめします。

```
#videobg {  
  background: url(video.jpg) no-repeat;  
  background-size: cover;  
}
```

オンラインでメディアをむ <https://riptutorial.com/ja/html/topic/2111/メディア>

30: ラベル

- `<label>Example <input type="radio" name="r"></label>` //コントロールをりす
- `<label for="rad1">Example</label> <input id="rad1" type="radio" name="r">` //for

パラメーター

にと って	ターゲットIDへの。つまり <code>for="surname"</code>
	HTML5 、 <code>form</code> をむフォームへの。ラベルは <code><form></code> にあることがされます。 <code>form="someFormId"</code> がされている、これにより、ドキュメントのどこにでも Label をすることが出来ます。

Examples

な

ラベルきのシンプルなフォーム...

```
<form action="/login" method="POST">

  <label for="username">Username:</label>
  <input id="username" type="text" name="username" />

  <label for="pass">Password:</label>
  <input id="pass" type="password" name="pass" />

  <input type="submit" name="submit" />

</form>
```

5

```
<form id="my-form" action="/login" method="POST">

  <input id="username" type="text" name="username" />

  <label for="pass">Password:</label>
  <input id="pass" type="password" name="pass" />

  <input type="submit" name="submit" />

</form>

<label for="username" form="my-form">Username:</label>
```

ラベルについて

`<label>`は、フォームアクションをするためにされます。

ユーザーインターフェイスのでは、タイプ `radio` や `checkbox` のようなターゲット/をにするためにされ `checkbox` 。

ラッパーとしての `<label>`

それは、のアクションをむことができる

```
<label>
  <input type="checkbox" name="Cats">
  I like Cats!
</label>
```

ターゲット `input` がテキストをクリックすると、その/がわりります

としての `<label>`

`for` をする `for`、コントロールを `label` としてするはありませんが、 `for` はそのIDとするがあります

```
<input id="cats" type="checkbox" name="Cats">
<label for="cats" >I like Cats!</label>
```

`<label>`にのコントロールをしないでください

オンラインでラベルをむ <https://riptutorial.com/ja/html/topic/1704/ラベル>

31: リスト

き

HTMLは、きリスト、けられていないリスト、およびリストの3つのでリストをします。きリストは、リストのをすをし、けられていないリストは、されたでをリストするためのきなどのみシンボルをし、リストは、でをリストするためにインデントをします。このトピックでは、HTMLマークアップでのこれらのリストのとみわせについてします。

- `<ol> ordered list items </ol>`
- `<ul> unordered list items </ul>`
- `<li> list item (ordered and not) </li>`
- `<dl> description list items </dl>`
- `<dt> description list title </dt>`
- `<dd> description list description </dd>`

`<ul style="list-style-type:disc">`ように、リストスタイルのCSSプロパティを`<ul>`タグにして、リストアイテムのマークにするアイコンのをすることができます。の`list-style-types`がされています

- `"list-style-typedisc"`はデフォルトのドットです。
- `"list-style-typecircle"`はりつぶされていないです。
- `"list-style-typesquare"`はりつぶされたです。
- `"list-style-typenone"`はくマークをしません。

また、`<ol> <ol type="1">`ように、けのをするために、`<ol>`タグにタイプをすることもできます。のタイプがされています

- `type = "1"`がデフォルトののです。
- `type = "A"`はをアルファベットにします
- `type = "a"`はをアルファベットにします
- `type = "I"`はのローマをします
- `type = "i"`はのローマをします

Examples

けられていないリスト

けられていないリストは`<ul>`タグででき、リストは`<li>`タグでできますのを。

```
<ul>
  <li>Item</li>
  <li>Another Item</li>
  <li>Yet Another Item</li>
</ul>
```

これにより、きのリストがされますデフォルトスタイル。

-
- のアイテム
- もうつのアイテム

のがでないのリストをするには、`ul`をするがあります。のをするとリストがしくないは、`<ol>`をするがあります。

オーダーリスト

`<ol>`タグをしてきリストをすることができ、リストは`<li>`タグをしてののようになります。

```
<ol>
  <li>Item</li>
  <li>Another Item</li>
  <li>Yet Another Item</li>
</ol>
```

これによりきのリストがされますデフォルトのスタイルです。

- 1.
2. のアイテム
3. もうつのアイテム

でをする

きリストのリストにされるをするには、いくつかのがあります。のは、`start`をして`start`をする`start`です。リストはこのされたからし、どおり1ずつしけます。

```
<ol start="3">
  <li>Item</li>
  <li>Some Other Item</li>
  <li>Yet Another Item</li>
</ol>
```

これによりきのリストがされますデフォルトのスタイルです。

- 3.
4. そののアイテム
5. もうつのアイテム

のリストをののにすることもできます。されたをつものよりのリストは、リストがどこにあったかをして、そののから1ずつインクリメントしけます。

```
<li value="7"></li>
```

また、リストの`value`をすることで、よりいでけをすることで、きリストののシステムをきするこ

ともできます。したがって、リストがすでに7にっていて、4でリストがつかった、そのリストは4としてされ、そのからびカウントをけます。

```
<ol start="5">
  <li>Item</li>
  <li>Some Other Item</li>
  <li value="4">A Reset Item</li>
  <li>Another Item</li>
  <li>Yet Another Item</li>
</ol>
```

したがって、のでは、5,6,4,5,6のパターンにうリストがされます。これは、のよりもさいからめて、リストの6をします。

startおよびvalueは、きリストがローマまたはとしてされるようにされていても、のみをくれます。

5

あなたは、olにreversedをすることでをにすることができます

```
<ol reversed>
  <li>Item</li>
  <li>Some Other Item</li>
  <li value="4">A Reset Item</li>
  <li>Another Item</li>
  <li>Yet Another Item</li>
</ol>
```

のは、しいポッドキャストのエピソードやプレゼンテーションなど、にリストにしているに、もしいをにするにちます。

のタイプをする

リストマーカーにされるのタイプは、typeをしてにできます

```
<ol type="1|a|A|i|I">
```

タイプ		
1	デフォルト - 10	1,2,3,4
a	アルファベット	あいうえお
A	アルファベット	あいうえお
i	ローマ	i、ii、iii、iv
I	ローマ	I、II、III、IV

アイテムのリストをするには、`ol`をしてください。アイテムがにけされ、をするがあります。のをしてもリストがしくないは、`<ul>`をするがあります。

リスト

リストまたはHTML5よりにびされたリストは、`dl`でできます。これは、とのグループでされ、`dt`にがえられ、は`dd`にえられます。

```
<dl>
  <dt>name 1</dt>
  <dd>value for 1</dd>
  <dt>name 2</dt>
  <dd>value for 2</dd>
</dl>
```

ライブデモ

- グループには、のやのをすをつことができます。

```
<dl>

  <dt>name 1</dt>
  <dt>name 2</dt>
  <dd>value for 1 and 2</dd>

  <dt>name 3</dt>
  <dd>value for 3</dd>
  <dd>value for 3</dd>

</dl>
```

ライブデモ

ネストされたリスト

リストをネストすると、リストのサブをすことができます。

```
<ul>
  <li>item 1</li>
  <li>item 2
    <ul>
      <li>sub-item 2.1</li>
      <li>sub-item 2.2</li>
    </ul>
  </li>
  <li>item 3</li>
</ul>
```

- アイテム1
- 2
 - サブアイテム2.1
 - サブアイテム2.2

- アイテム3

ネストされたリストは、`li` のでなければなりません。

さまざまなリストをれにすることもできます

```
<ol>
  <li>Hello, list!</li>
  <li>
    <ul>
      <li>Hello, nested list!</li>
    </ul>
  </li>
</ol>
```

オンラインでリストをむ <https://riptutorial.com/ja/html/topic/393>/リスト

32: リソースのリンク

き

くのスクリプト、アイコン、およびスタイルシートをHTMLマークアップにきむことができますが、これらのリソースをのファイルにめてにリンクするがです。このトピックでは、スタイルシートやスクリプトなどのリソースをHTMLドキュメントにリンクするについてします。

- `<link rel="link-relation" type="mime-type" href="url">`
- `<script src="path-to-script"></script>`

パラメーター

charset	リンクされたドキュメントのエンコーディングをします。
crossorigin	がクロスオリジンをするをします。
href	リンクされたドキュメントのをします。
hreflang	リンクされたドキュメントのテキストのをします。
media	リンクされたドキュメントをするデバイスをします。のデバイスについてスタイルシートをするによくされます
rel	です。のドキュメントとリンクされているドキュメントとのをします。
rev	リンクされたドキュメントとのドキュメントとのをします。
sizes	リンクされたリソースのサイズをします。 <code>rel="icon"</code> ときのみ、
target	リンクされたドキュメントのみみをします。
type	リンクされたドキュメントのメディアタイプをします。
integrity	ブラウザがそのをできるように、リンクされたリソースのbase64でエンコードされたハッシュsha256、sha384、またはsha512をします。

Examples

CSSスタイルシート

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

なは、HTMLのにある<head>タグのにCSS <link>タグをすることです。このでは、CSSがロードされるまで、スタイルのされていないHTMLをするのではなく、CSSがにロードされ、ロードのページにされます。HTML5はCSSをサポートするため、HTML5ではtypeはです。

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

そして

```
<link rel="stylesheet" href="path/to.css">
```

HTML5でもじことをする。

@importではありませんが、なCSSで@importステートメントをするもあります。このような

```
<style type="text/css">
  @import("path/to.css")
</style>

<style>
  @import("path/to.css")
</style>
```

JavaScript

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

なはJavaScriptをすることです<script>ちょうどじるにタグ</body>タグ。スクリプトをにロードすると、サイトのビジュアルがよりくされ、JavaScriptがまだみまれていないとやりとりするのをげます。

```
<script src="path/to.js" async></script>
```

のとして、ロードのJavascriptコードがページのにでない、にロードしてページのみみをすることができます。asyncをすると、ブラウザのがしてみまれ、にダウンロードされると、JavascriptファイルをするためにHTMLがされます。

えき

```
<script src="path/to.js" defer></script>
```

スクリプトは、スクリプトとていますが、HTMLがにされたにのみがされるがなります。スクリプトは、スクリプトとに、にロードされることがされています。

<noscript>

```
<noscript>JavaScript disabled</noscript>
```

<noscript>は、ユーザーがスクリプトをにしている、またはブラウザがスクリプトのをサポートしていないにされるコンテンツをします。 <noscript>タグは、 <head>または<body>いずれかにできます。

ファビコン

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

PNGファイルにはMIMEタイプのimage/pngをし、アイコン *.ico ファイルにはimage/x-iconをします。いは、 [このSOの](#)をしてください。

ウェブサイトのルートにあるfavicon.icoというのファイルは、<link>タグをとせずになみまれます。このファイルがされた、ブラウザのキャッシュをするのがく、になるがあります。

CSS

```
<link rel="alternate stylesheet" href="path/to/style.css" title="yourTitle">
```

のブラウザでは、スタイルシートがされているにされます。デフォルトではされませんが、はブラウザのでできます

Firefoxでは、`[[> [ページスタイル]サブメニュー`をしてスタイルシートをできるようになりました。また、Internet ExplorerではこのInternet Explorer 8もサポートされています。このをしてくださいバージョン48。 Webページは、ユーザーがスタイルをりえるためののユーザーインターフェイスをすることもできます。

[MDN Docs](#)

Web フィード

rel="alternate"をして、Atom / RSSフィードのをします。

```
<link rel="alternate" type="application/atom+xml" href="http://example.com/feed.xml" />
<link rel="alternate" type="application/rss+xml" href="http://example.com/feed.xml" />
```

[RSSフィード](#)と[Atomic RSS](#)のMDNドキュメントをしてください。

リンク 'media'

```
<link rel="stylesheet" href="test.css" media="print">
```

Mediaは、どのタイプのメディアにどのスタイルシートをするかをします。 `print` をすると、ページのスタイルシートのみがされます。

こののは、 [メディア mediatype](#) CSS [メディアクエリ](#)にのいずれかにすることができます。

と

たとえば、ページがなのである、 `prev`と`next` をしてにあるページをポイントすることができます。

```
<link rel="prev" href="http://stackoverflow.com/documentation/java/topics">
<link rel="next" href="http://stackoverflow.com/documentation/css/topics">
```

リソースのヒント **dns-prefetch**、`prefetch`、`prerender`

`preconnect` は、DNSをするで `dns-prefetch` にています。ただし、TCPハンドシェイクとオプションのTLSネゴシエーションもいます。これはなです。

```
<link rel="preconnect" href="URL">
```

DNSプリフェッチ

ブラウザーにURLのDNSをするようするので、そのURLのすべてのアセットがよりにみられます。

```
<link rel="dns-prefetch" href="URL">
```

プリフェッチ

のリソースをプリフェッチするがあることをブラウザーにし、よりにロードできるようにします。

```
<link rel="prefetch" href="URL">
```

DNS-Prefetchはドメインのみをし、プリフェッチはされたリソースをダウンロード/します。

レンダリング

ブラウザがバックグラウンドでURLをしてレンダリングするようにするので、ユーザーがそのURLにするのにユーザーにできます。これはなです。

```
<link rel="prerender" href="URL">
```

オンラインでリソースのリンクをむ <https://riptutorial.com/ja/html/topic/712>/リソースのリンク

33:

き

インタラクティブなWebシステムのなコンポーネントであるタグは、ユーザーからののをけるようにされたHTMLです。なるタイプのは、されたフォーマットにするようにされたデータをし、パスワードにセキュリティをすることができる。

- `<input type="" name="" value="">`

パラメーター

パラメータ	
クラス	のクラスをします。
id	のIDをします。
タイプ	するコントロールのタイプをします。がされているhidden、text、tel、url、email、password、date、time、number、range、color、checkbox、radio、file、submit、image、reset、およびbutton。がされていない、またはがな、またはブラウザがされたタイプをサポートしていないは、デフォルトでtextになります。
	のをします。
	をにするがあることをすブール。なコントロールはできません。フォームにされず、フォーカスをけることができません。
チェックした	typeのがradioまたはcheckboxの、このBooleanのは、コントロールがデフォルトでされていることをします。それのはされます。
	HTML5のファイルまたはをすることができることをします fileおよびemailタイプのにのみされfile
プレースホルダ	HTML5コントロールにできるをユーザーにします。プレースホルダテキストにまたはをめないでください

パラ メー タ	
オー トコ ンプ リー ト	HTML5 コントロールのをブラウザがにできるかどうかをします。
みり	がでないことをすブール。みりコントロールはとしてフォームのにされますが、フォーカスはされません。 HTML5 type のが hidden 、 range 、 color 、 checkbox 、 radio 、 file または button いずれかにされている、このはされ button 。
	HTML5 フォームがされるためには、がするがあること、またはをチェックするがあることをします
alt	イメージがされないのテキスト。
オー トフ オー カス	ページがみまると <input> にフォーカスがられるはずです。
	<input> のをします。
ステ ップ	step は、なをします。 number 、 range 、 date 、 datetime-local 、 month 、 time 、 week タイプでし week 。

のHTML5 voidとに、 <input> はであり、 <input /> とくことができます。 HTML5ではこのスラッシュはありません。

HTMLでなタイプはのとおりで。

- button
- checkbox
- file
- hidden
- image
- password
- radio
- reset
- submit
- text デフォルト

5

は、HTML5のとしてたにされたタイプです。これらのタイプのは、すべてのWebブラウザでサポ

ートされていません。がサポートされていない、inputはデフォルトでtextになりtext。

- color
- date
- datetime および
- datetime-local
- email
- month
- number
- range
- search
- tel
- time
- url
- week

どのブラウザがどのタイプをサポートしているかをべるには、 caniuse.comにすることができます。

Examples

チェックボックスとラジオボタン

チェックボックスとラジオボタンは

もなチェックボックスまたはラジオボタンは、それぞれtypeがcheckboxまたはradio <input>です。

```
<input type="checkbox">
<input type="radio">
```

のスタンドアローンのチェックボックスは、yes-or-noのようなバイナリオプションにされます。チェックボックスはしているため、ユーザーはチェックボックスのグループできなだけくのをできます。つまり、1つのチェックボックスをオンにしても、チェックボックスグループののチェックボックスのチェックはされません。

ラジオボタンは、グループにあるすべてのボタンでじnameをしてされるグループのラジオボタンでグループされていないは、わりにチェックボックスをすることをしますでされます。ラジオボタンののはにです。つまり、ユーザはラジオボタンのグループから1つののみをできます。ラジオボタンがチェックされると、にチェックされたじnameのラジオボタンはチェックされなくなります。

```
<input type="radio" name="color" id="red" value="#F00">
<input type="radio" name="color" id="green" value="#0F0">
<input type="radio" name="color" id="blue" value="#00F">
```

されると、ラジオボタンはチェックされていないまたはりつぶされたチェックされているとして

されます。チェックボックスは、チェックマークなしまたはチェックマークとしてされます。ブラウザとオペレーティングシステムによっては、のがなくなることがあります。

チェックボックスとラジオボタンには、そのをするためのがあります。

value

のとに、`value`は、フォームにボタンにけるをします。ただし、チェックボックスとラジオボタンは、をし`on`、のをするのではなく、にデフォルトで`on`になるというです。`value`は、ボタンのにはされません。

checked

`checked`は、チェックボックスまたはラジオボタンのをします。これはブールのものでありすることができます。

これらはそれぞれ、なラジオボタンをします。

```
<input checked>
<input checked="">
<input checked="checked">
<input checked="ChEcKeD">
```

`checked`れていないボタンののは、`checked`がないことです。

```
<input type="radio">
<input type="checkbox">
```

`<form>`リセットすると、チェックボックスとラジオボタンが`checked`のになります。

アクセシビリティ

ラベル

ボタンにコンテキストをえ、ボタンがであるかをユーザにすために、それぞれにラベルがです。これは`<label>`をしてボタンをラップすることでうことができます。また、ラベルをクリックするので、するボタンをします。

```
<label>
  <input type="radio" name="color" value="#F00">
  Red
</label>
```

または`for`がボタンの`id`にされた`<label>`をしています。

```
<input type="checkbox" name="color" value="#F00" id="red">
<label for="red">Red</label>
```

ボタン グループ

ラジオボタンはグループのラジオボタンにするため、ラジオボタンのグループにラベルまたはコンテキストをするのがです。

グループにラベルをけるには、`<fieldset>` `<legend>`に`<legend>`をむラジオボタンをめるがあります。

```
<fieldset>
  <legend>Theme color:</legend>
  <p>
    <input type="radio" name="color" id="red" value="#F00">
    <label for="red">Red</label>
  </p>
  <p>
    <input type="radio" name="color" id="green" value="#0F0">
    <label for="green">Green</label>
  </p>
  <p>
    <input type="radio" name="color" id="blue" value="#00F">
    <label for="blue">Blue</label>
  </p>
</fieldset>
```

チェックボックスは、するチェックボックスのグループをするフィールドセットととのでグループすることもできます。ただし、チェックボックスはにではないため、じをしないようにしてください。これをうと、じキーにしてのをするフォームがされ、すべてのサーバーがじでこれをするわけではありませんの。チェックボックスにはのをけるか、かっこ `[]` をして、そのキーののをするがあることをします。どちらのをするかは、フォームデータのクライアントまたはサーバーのによってなります。また、ブラウザとスクリーンリーダーのみわけによっては、フィールドセットのフィールドののがみられるため、をくしておくがあります。

された

```
<input type="hidden" name="inputName" value="inputValue">
```

のはユーザーにはされませんが、フォームがされたときにそのがサーバーにされます。

パスワード

```
<input type="password" name="password">
```

が`password`である`type`をつ`input`は、`type=text`とのテキストフィールドをしますが、ユーザーがするとテキストはされません。

```
<input type="password" name="password" placeholder="Password">
```

プレースホルダテキストはプレーンテキストでされ、ユーザーがをするとにきされます。



ブラウザやシステムによっては、パスワードフィールドのデフォルトをして、にしたをすることもできます。



する

```
<input type="submit" value="Submit">
```

は、クリックされるとにあるフォームをするボタンをします。

よりにスタイルすることができる、またはのをむボタンがなは、 `<button>` をすることもできます。

```
<button type="submit">
   Submit
</button>
```

ファイル

```
<input type="file" name="fileSubmission">
```

ファイルをすると、ローカルファイルシステムからのページでするファイルをできます。 `form` とみわせてすると、ユーザーはファイルをサーバーにアップロードできるようになりますについては、「[ファイルのアップロード](#)」を。

のでは、ユーザーが `file` をして `file` システムからファイルをし、そのファイルを `upload_file.php` というのサーバーのスクリプトにアップロードすることができ `upload_file.php` 。

```
<form action="upload_file.php" method="post" enctype="multipart/form-data">
  Select file to upload:
  <input type="file" name="fileSubmission" id="fileSubmission">
  <input type="submit" value="Upload your file" name="submit">
</form>
```

のファイル

する `multiple` ユーザーが **1** つののファイルをすることができますを

```
<input type="file" name="fileSubmission" id="fileSubmission" multiple>
```

ファイルをける

Acceptは、ユーザーができるファイルのをします。 .png、 .gif、 .jpeg。

```
<input type="file" name="fileSubmission" accept="image/x-png,image/gif,image/jpeg" />
```

HTMLのは、のなについてブラウザによってにわれます。 JavaScriptののにまたはにきえることができます。こののは、にされたHTTPをしてユーザーによってされるがあるため、サーバーのにわるものではありません。は、フォームをしようとしたときにのみするため、JavaScriptをしていないりがわれるように、すべてのきがフォームになればなりません。またはみりのはをトリガーしません。

いくつかのしいタイプ email、 url、 tel、 dateなどはにされ、のはありません。

5

をパスするためにフィールドをするがあることをすには、 required をします。

```
<input required>
```

/

さのをすには、 minlength と maxlength をします。ほとんどのブラウザでは、ユーザーがボックスにをえるをすることをぎ、をみるでもエントリがにならないようにします。

```
<input minlength="3">
<input maxlength="15">
<input minlength="3" maxlength="15">
```

の

min および max をして、ユーザーがタイプ number または range にできるのをする

```
Marks: <input type="number" size="6" name="marks" min="0" max="100" />
Subject Feedback: <input type="range" size="2" name="feedback" min="1" max="5" />
```

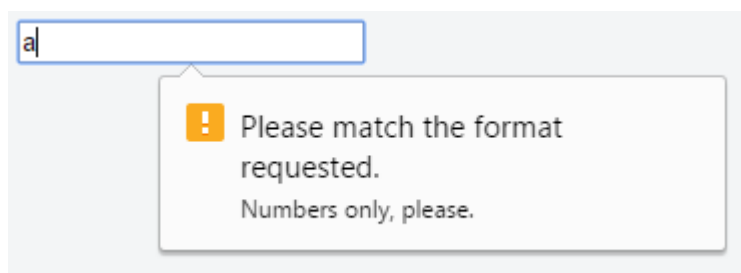
5

パターンにする

さらにするには、 pattern をして、をすためにするがあるをします。フィールドがしないは、メッセージにまれる title することもできます。

```
<input pattern="\d*" title="Numbers only, please.">
```

このフィールドになをむフォームをしようとする、のメッセージがGoogle Chromeバージョン51にされます。



のブラウザーのにはサポートされていますが、すべてのブラウザーでなパターンのメッセージがされるわけではありません。

[CanIUse](#)のサポートをし、それにじてしてください。

5

ファイルタイプをける

`file`タイプのフィールドの、ビデオ、、オーディオ、のファイル、またはの[メディアタイプ](#)など、のタイプのファイルのみをけることができます。えは

```
<input type="file" accept="image/*" title="Only images are allowed">
```

カンマでのをすることができます。

```
<input type="file" accept="image/*,.rar,application/zip">
```

`novalidate`にを `form` または `formnovalidate` ボタンへのは、フォームのをぐことができます。えは

```
<form>
  <input type="text" name="name" required>
  <input type="email" name="email" required>
  <input pattern="\d*" name="number" required>

  <input type="submit" value="Publish"> <!-- form will be validated -->
  <input type="submit" value="Save" formnovalidate> <!-- form will NOT be validated -->
</form>
```

フォームには、ドラフトを「」するためになフィールドがありますが、ドラフトを「」するはありません。

リセット

```
<input type="reset" value="Reset">
```

タイプ `reset` のはボタンをします。このボタンはクリックされると、フォームにまれるすべてのを

デフォルトにリセットします。

- フィールドのテキストは、またはデフォルト `value` をしてにリセットされます。
- メニューのオプションは、 `selected` がないり `selected` ます。
- すべてのチェックボックスとラジオボックスは、 `checked` がないり、されます。

リセットボタンは、をすために `<form>` のにあるか、または `form` をして `<form>` にされているがあります。ボタンはこのフォームののみをリセットします。

5

```
<input type="number" value="0" name="quantity">
```

が `number` である `type` をつ `Input` は、のををすにするためのなコントロールをします。

このフィールドでは、しいがいていとはりません。のでできるすべてのシンボルをします。たとえば、ユーザは `ele-,0` ようなをでき `ele-,0` 。

```
<input type="tel" value="+8400000000">
```

が `tel` である `type` をつ `input` は、をするための1のプレーンテキストコントロールをします。

Eメール

5

`<input type="email">` は、メールアドレスをむフィールドにされます。

```
<form>
  <label>E-mail: <label>
  <input type="email" name="email">
</form>
```

メールアドレスは、ブラウザのサポートによってににされます。

ボタン

```
<input type="button" value="Button Text">
```

ボタンをして、フォームをせずにページでアクションをトリガすることができます。さらににスタイルすることができたり、のをむボタンがなは、 `<button>` をすることもできます。

```
<button type="button">Button Text</button>
```

ボタンは、 、 "onclick" イベントでされます

```
<input type="button" onclick="alert('hello world!')" value="Click Me">
```

または

```
<button type="button" onclick="alert('hello world!')">Click Me</button>
```

[name]

フォームデータとにされるボタンのname。

[type]

ボタンのtype。

なはのとおりです。

submit ボタンはフォームデータをサーバーにします。これは、がされていない、またはがのまたははなににされたのデフォルトです。

reset ボタンはすべてのコントロールをにリセットします。

button このボタンにはデフォルトのはありません。イベントのにトリガされる、のイベントにけられたクライアントのスクリプトをつことができます。

menu このボタンは、されたをしてされたポップアップメニューをきます。

[value]

ボタンのです。

5

ボタンの

form	ボタンがするフォームのIDをします。 されていないは、そのフォームするにします。
formaction	フォームデータのをします。 このボタンをしてフォームがされたとき。
formenctype	フォームデータのエンコードをします。 このボタンをしてサーバーにします。 formmethod="post"でのみできます。
formmethod	するHTTPメソッドPOSTまたはGETをします。

	このボタンをしてフォームデータをするとき
formnovalidate	にフォームデータをしないことをします。
formtarget	したをするをします。 このボタンをしてフォームをした

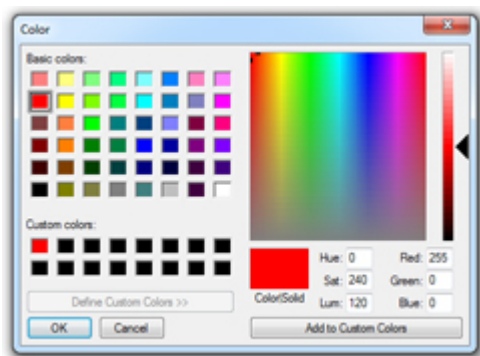
5

```
<input type="color" name="favcolor" value="#ff0000">
```

サポートするブラウザでは、`type`のが`color`である`input`は、`color`のとしい`color`をつボタンのようなコントロールをしますがされていないはデフォルトはで、な16です。



このボタンをクリックすると、オペレーティングシステムのカラーウィジェットがき、ユーザーはカラーをできます。



このタイプをサポートしていないブラウザのフォールバックは、の`type=text`です。

URL

5

```
<input type="url" name="Homepage">
```

これは、URLアドレスをむフィールドにされます。

ブラウザのサポートによっては、`url`フィールドはににされます。

いくつかのスマートフォンは`url`をし、`".com"`をURLにさせるためにキーボードにします。

5

```
<input type="date" />
```

あなたがをするためのピッカーがにポップアップされます。 FirefoxまたはInternet Explorerではサポートされていません。

DateTime - ローカル

5

```
<input type="datetime-local" />
```

ブラウザのサポートにして、とをのためのとのツールがにポップアップされます。

```
<input type="image" src="img.png" alt="image_name" height="50px" width="50px"/>
```

。テキストをするには、srcをしてイメージのソースをし、altをするがあります。 heightとwidthをして、イメージのサイズをピクセルですることができます。

5

```
<input type="range" min="" max="" step="" />
```

ながでないをするためのコントロール。

		デフォルト
	の	0
	の	100
ステップ	ごとにする。	1

5

```
<input type="month" />
```

ブラウザのサポートにして、をするコントロールがされます。

5

```
<input type="time" />
```

timeは、このををすをけるものとしてマークします。は[RFC 3339](#)でされており、

```
19:04:39
08:20:39.04
```

、AndroidEdge、Chrome、Opera、Chromeのすべてのバージョンでtype = "time"がサポートされています。Android Browserのバージョン、に4.4がサポートされています。Safari for iOSはなサポートをし、、、およびステップのをサポートしていません。

5

```
<input type="week" />
```

ブラウザーのサポートにして、コントロールにのとタイムゾーンのないをするためのコントロールがされます。

テキスト

もなタイプとtypeがされていないのデフォルト。このタイプは、がからにされたのテキストフィールドをします。それのはすべてこれにできます。<input>は、ユーザーがデータをできるコントロールをするために<form>でされます。

```
<input type="text">
```

またはtypeをせずに、defaultをして

```
<input>
```

テキストフィールドのデフォルトのは20です。これはのようにsizeのをすることでできます

```
<input type="text" size="50">
```

sizeは、CSSでをするのとはまったくなります。をすると、がにワイドでなければならないのピクセル、のなどがされます。sizeをすると、sizeれているフォントとののについて、りてるをします。

sizeをしても、ボックスにできるはにされず、ボックスののみがされます。さをするについては、[の](#)をしてください。

フィールドは1のテキストのみをします。のテキストにしてのテキストがなは、わりに<textarea>をします。

サーチ

5

は、テキストにされます。ほとんどのブラウザでは、テキストのスペースののがされます

```
<input type="search" name="googlesearch">
```

DateTime グローバル

valueが " **datetime** "のtypeをつinputは、のをグローバルなタイムゾーンきをすにするためのコントロールをします。

```
<fieldset>
  <p><label>Meeting time: <input type=datetime name="meeting.start"></label>
</fieldset>
```

される

- グローバル
-
-
-
- タイプ
- オートコンプリート
- オートフォーカス
- リスト
-
- ステップ
- みり
- な

オンラインでをむ <https://riptutorial.com/ja/html/topic/277/>

34:

パラメーター

グローバル	のHTML5でな。これらののなドキュメントについては、 MDNグローバル をしてください。
	のをす。フォームとして、 <code>document.forms</code> プロパティをして、そのでをできます。このは、フォームにをするためにもされます。
にとつて	フォームのIDのスペースでられたリスト <code><inputs id="inp1"></code> for value is "inp1" は、がをすることをします。
	にけられた <code><form></code> をす。はににあるは <code><form></code> 、このは、はまだにしていることをします <code><form></code> とったのコレクションやをけ <code><form></code> 。

Examples

ForとFormをした

のデモでは、`<output>`の`[for]` `[form]`と`[form]`のについて`[form]`ます。`<output>`はするためにJavaScriptがであることにしてください。このですように、インラインJavaScriptはフォームでよくされます。`<input>`は`type="number"`、その`value`はではなく、テキストです。したがって、`value`sをするがあるは、`parseInt()`、`parseFloat()`、`Number()`などのメソッドをして、`value`にするがあり`parseInt()`。

ライブデモ

```
<!--form1 will collect the values of in1 and in2 on 'input' event.-->
<!--out1 value will be the sum of in1 and in2 values.-->

<form id="form1" name="form1" oninput="out1.value = parseInt(in1.value, 10) +
parseInt(in2.value, 10)">

  <fieldset>

    <legend>Output Example</legend>

    <input type="number" id="in1" name="in1" value="0">
    <br/>
    +
    <input type="number" id="in2" name="in2" value="0">

  </fieldset>

</form>
```

```
<!--[for] attribute enables out1 to display calculations for in1 and in2.-->
<!--[form] attribute designates form1 as the form owner of out1 even if it isn't a
descendant.-->

<output name="out1" for="in1 in2" form="form1">0</output>
```

をつ

```
<output name="out1" form="form1" for="inp1 inp2"></output>
```

オンラインでをむ <https://riptutorial.com/ja/html/topic/723/>

35: めみ

パラメーター

パラメーター	
src	リソースのアドレス
type	めみリソースのタイプ
width	
height	

Examples

な

`embed` タグはHTML5のです。このは、はHTMLアプリケーションまたはコンテンツのポイントをします。

```
<embed src="myflash.swf">
```

MIMEタイプの

MIMEタイプは、`type`をしてするがあります。

```
<embed type="video/mp4" src="video.mp4" width="640" height="480">
```

オンラインでめみをむ <https://riptutorial.com/ja/html/topic/8123/めみ>

36: エンティティ

Examples

の

のはHTMLにされているがあり、のHTMLコードをけるがあるため、することはできません。たとえば、ソースコードにの<>をしようとする、にしないがじることがあります。に、ソースコードにかれているは、HTMLにりにされないことがあります。☎のようなものは、ASCIIセットではできません。

このために、エンティティがされます。これらはentity_name;というentity_name;またはentity_number;。なHTMLエンティティのいくつかをにします。

キャラクター		エンティティ	エンティティ
""		 	
"<"		<	<
">"	よりきい	>	>
""	アンパサンド	&	&
" - "	emダツシュ	—	—
" - "	エンダツシュ	–	–
"©"		©	©
"®"		®	®
"TM"		™	™
「☎」		☎	☎

したがって、

©2016 Stack Exchange Inc.

のHTMLコードがされます。

```
<b>&copy; 2016 Stack Exchange Inc.</b>
```

HTMLのエンティティ

htmlでWebページをするには、くのシンボルやがですが、にをすると、のがされているのhtmlコードやキーボードでのができなくなることがあります。したがって、をするために、にコードでなるシンボルをできるようにするために、w3 orgはたちに「エンティティ」をします。

エンティティは 'Entity Name' - entity_name;でされています。と 'エンティティ' - entity_number;ページになシンボルをするには、2つのどちらかをするがあります。

いくつかのエンティティのリストは<https://dev.w3.org/html5/html-author/charref>にあります。

'magnifying glass'にエンティティをしたな

```
<input type="text" placeholder="  &#128269; Search"/>
```

それは

オンラインでエンティティをむ <https://riptutorial.com/ja/html/topic/5229/エンティティ>

37:

き

はもなHTMLです。このトピックでは、のHTMLでのについてします。

パラメーター

カラム	カラム
<p>	をする
 	1のをします。
<pre>	あらかじめフォーマットされたテキストをする

Examples

HTML

HTML <p>はをします

```
<p>This is a paragraph.</p>
<p>This is another paragraph.</p>
```

-

HTMLをどのようにするかはわかりません。

または、およびサイズされたウィンドウは、なるをします。

HTMLでは、HTMLコードになスペースやなをしてをすることはできません。

ページがされると、ブラウザはなとなをします。

```
<p>This is          another      paragraph, extra spaces  will be  removed  by
browsers</p>
```

オンラインでをむ <https://riptutorial.com/ja/html/topic/7997/>

38:

き

すべてのHTMLタグがじであるわけではありません。ほとんどののはタグ、タグ、およびをとしますが、voidとばれるにはがまれていないので、タグのみがです。このトピックでは、HTMLでvoidをにするについてします

voidはをつことはできませんが、をつことがあります。voidはクローズであるため、じタグをたないようにしてください。

HTML5では、のはです

- area
- base
- br
- col
- embed
- hr
- img
- input
- keygen
- link
- meta
- param
- source
- track
- wbr

Examples

の

HTML 4.01 / XHTML 1.0 Strictには、のvoidがまれています。

- area - クリックなみの
- base - すべてのリンクベースからのベースURLをします。
- br -
- col - テーブルのカラム[]
- hr - ルールライン
- img - イメージ
- ユーザがデータをinput フィールド
- link - リソースをドキュメントにリンクする
- meta - ドキュメントにするをする
- param - プラグインのパラメータをする

HTML 5には、のリストのされていないタグがすべてまれています。

- `command` - ユーザが[obsolete]をびすことができるコマンドをします。
- `keygen` - Webのをにする[]
- `source` - `picture`、`audio`、および`video`のメディアソースをします。

のはvoidをみません

```
<div>
  <a href="http://stackoverflow.com/">
    <h3>Click here to visit <i>Stack Overflow!</i></h3>
  </a>
  <button onclick="alert('Hello!');">Say Hello!</button>
  <p>My favorite language is <b>HTML</b>. Here are my others:</p>
  <ol>
    <li>CSS</li>
    <li>JavaScript</li>
    <li>PHP</li>
  </ol>
</div>
```

すべてのがタグ、タグ、およびタグとタグのテキストまたはそののをどのようにつかにしてください。ただし、のではタグがされています。

```

<br>
<hr>
<input type="number" placeholder="Enter your favorite number">
```

`img`タグをいて、これらのvoidはすべて、タグのみをちます。 `img`タグは、のタグとはなり、タグのよりきいのにじ、があります。スラッシュのにスペースをれるのがベストプラクティスです。

オンラインでをむ <https://riptutorial.com/ja/html/topic/1449/>

39: し

き

HTMLには、なるタグだけでなく、さまざまなサイズとさのしをす6つの々のヘッダータグがされています。し1はし6に、し1はもきくいてテキストをち、し6はレベルまでさく、もい。このトピックでは、これらのタグをしくするについてします。

- `<h1>...</h1>`
- `<h2>...</h2>`
- `<h3>...</h3>`
- `<h4>...</h4>`
- `<h5>...</h5>`
- `<h6>...</h6>`

- `h1 - h6`には、タグとタグのがです。 [1](#)

- `h1 - h6`は、デフォルトではブロックレベルCSSスタイルです `display: block`。 [2](#)

- `h1 - h6`を[section](#)としないでください

- しタグ `h1 - h6` にしていない `head` タグ。

- されたコンテンツ [フレーズのコンテンツ](#)

- しのなるCSSスタイルは、`font-size`と`margin`がなります。のためののCSS-`h1 - h6`は、によって「`」`としてけとしてすることができる[W3C](#)

- エンジンのスパイダーページをエンジンにするコードは、ページのをするためにのいもの `h1`がもく、`h2`がよりなく、`h3`がよりない...などのしににをいます。

Examples

しの

しは、それらがするトピックをするためにされ、`<h1>` `<h6>`タグでされます。しはすべての[グローバル](#)をサポートします。

- `<h1>`はもなしをします。
- `<h6>`は、もでないしをします。

しをする

```
<h1>Heading 1</h1>
<h2>Heading 2</h2>
<h3>Heading 3</h3>
```

```
<h4>Heading 4</h4>
<h5>Heading 5</h5>
<h6>Heading 6</h6>
```

のをする

エンジンやのユーザーエージェントは、しについてページコンテンツをけします。たとえば、をするは、しにしいをすることがです。

に、にはメインタイトルの_{h1}が1つ、_{h2}サブタイトルがですにじてレイヤーをろす。よりいレベルに_{h1}がある、それらはレベルのをするためにされません。

をするためのな

```
<h1>Main title</h1>
<p>Introduction</p>

  <h2>Reasons</h2>

    <h3>Reason 1</h3>
    <p>Paragraph</p>

    <h3>Reason 2</h3>
    <p>Paragraph</p>

  <h2>In conclusion</h2>
  <p>Paragraph</p>
```

オンラインでしをむ <https://riptutorial.com/ja/html/topic/226/>し

40:

パラメーター

パラメータ	
	タスクがとするの
	どのくらいのがすでにされているか
ポジション	このは、 <progress>ののをします
ラベル	このは、 <progress>ラベルのリストをしますする

<progress>は、Internet Explorerのバージョンが10ではサポートされていません

<progress>は、タスクではなく、ゲージであるかにされるったです。たとえば、<progress>をしてディスクのをすことはです。わりに、<meter>はこのタイプのユースケースでできます。

Examples

<progress>はHTML5ではしく、タスクのをすためにされます

```
<progress value="22" max="100"></progress>
```

これにより、バーが22

プログレスバーのをする

プログレスバーは、progress[value]セクタでスタイリングすることができます。

このでは、250px、さ20pxプログレスバーがされ20px

```
progress[value] {  
  width: 250px;  
  height: 20px;  
}
```

プログレスバーはにスタイルをするのがしいがあります。

Chrome / Safari / Opera

これらのブラウザでは、-webkit-appearanceセクタをしてタグのスタイルをします。これをにするには、をリセットすることができます。

```
progress[value] {
  -webkit-appearance: none;
  appearance: none;
}
```

さて、コンテナをスタイルすることができます

```
progress[value]::-webkit-progress-bar {
  background-color: "green";
}
```

Firefox

Firefoxでは、プログレスバーのスタイルが異なります。これらのスタイルをするがあります

```
progress[value] {
  -moz-appearance: none;
  appearance: none;
  border: none; /* Firefox also renders a border */
}
```

インターネットエクスプローラ

Internet Explorer 10では、`progress`がサポートされています。ただし、`background-color`プロパティはサポートされていません。わりに`color`プロパティをするがあります。

```
progress[value] {
  -webkit-appearance: none;
  -moz-appearance: none;
  appearance: none;

  border: none; /* Remove border from Firefox */

  width: 250px;
  height: 20px;

  color: blue;
}
```

HTML フォールバック

`progress`をサポートしていないブラウザでは、これをしてできます。

```
<progress max="100" value="20">
  <div class="progress-bar">
    <span style="width: 20%;">Progress: 20%</span>
  </div>
</progress>
```

`progress`タグをサポートしているブラウザは、にれになっているdivをします。タグをできないの

ブラウザでは、わりにdivがレンダリングされます。

オンラインでをむ <https://riptutorial.com/ja/html/topic/5055/>

41: メニューコントロール

- `<select name=""></select>`
- `<datalist id=""></datalist>`
- `<optgroup label="Option Group"></optgroup>`
- `<option value="">Option</option>`

Examples

メニューを

`<select>`は、ユーザーがオプションをできるドロップダウンメニューをします。

```
<select name="">
  <option value="1">One</option>
  <option value="2">Two</option>
  <option value="3">Three</option>
  <option value="4">Four</option>
</select>
```

サイズの

`size`でメニューのサイズをすることができます。サイズが0または1の、のドロップダウンスタイルメニューがされます。サイズが1よりきいは、ドロップダウンをのをするボックスにします。ごとに1つのオプションとスクロールバーがあり、なオプションをスクロールします。

```
<select name="" size="4"></select>
```

のオプションのメニュー

デフォルトでは、ユーザーは1つのオプションのみをできます。 `multiple`をすると、 `multiple`オプションをにし、したすべてのオプションをフォームにできます。 `multiple`をすると、ドロップダウンメニューがにボックスにされ、サイズがされているかのようにになります。これがしたときのデフォルトのサイズは、しているのブラウザによってまります。のをしながらドロップダウンスタイルメニューにすることはできません。

```
<select name="" multiple></select>
```

`multiple`をするは、サイズに0と1をするのにいがありますが、をしないはいはありません。0をすると、ブラウザはデフォルトのするようにプログラムされます。1をすると、のボックスのサイズがに1にされます。

オプショングループ

`<optgroup>`をすると、オプションのいリストでよりされたレイアウトをするために、メニューでオプションをきれいにまとめることができます。

はになものです。に `label` のをしてグループのタイトルをし、そのグループにめるがあるゼロのオプションをんでいます。

```
<select name="">
  <option value="milk">Milk</option>
  <optgroup label="Fruits">
    <option value="banana">Bananas</option>
    <option value="strawberry">Strawberries</option>
  </optgroup>
  <optgroup label="Vegetables" disabled>
    <option value="carrot">Carrots</option>
    <option value="zucchini">Zucchini</option>
  </optgroup>
</select>
```

オプショングループをする、すべてのオプションをグループにめるはありません。に、オプショングループをにすると、グループのすべてのオプションがになり、なグループの1つのオプションをでびにすることはできません。

オプション

メニューのオプションは、ユーザーがするものです。オプションののはのとおります。

```
<option>Some Option</option>
```

しかし、`<option>`ののテキストは必ずしもされるわけではなく、されていないのデフォルトとなることにすることがです。

オプションののおよびをするは、`value`および`label`。ラベルは、ドロップダウンメニューにされるテキストあなたがしているものとクリックしてするものをします。は、フォームのとともにされるテキストをします。これらののいずれかがされた、わりにのテキストがとしてされます。したがって、にしたは、これに "" することができます

```
<option label="Some Option" value="Some Option">
```

にメニューにオプションをするはないテキストとタグのにしてください。それらがまれている、のがにされていてテキストがでないため、のテキストはされます。しかし、あなたはこうにくのがいていとはわないでしょう。もなは、サーバーにされると、にはラベルになるテキストです。

```
<option value="option1">Some Option</option>
```

デフォルトでオプションをする

また、`selected` をメニューにすることで、メニューにするのオプションをデフォルトですることもできます。デフォルトでは、メニューでされているオプションがされていない、レンダリングにメニューのオプションがされます。のオプションに`selected`アトリビュートがアタッチされている、そのアトリビュートをつメニューにあるのオプションは、デフォルトでされたものになります。

```
<option value="option1" selected>Some option</option>
```

オプションのメニューでをしている、をつすべてのオプションがデフォルトでされ、をつオプションがないはされません。

```
<select multiple>
  <option value="option1" selected>Some option</option>
  <option value="option2" selected>Some option</option>
</select>
```

データリスト

`<datalist>` タグのされたオプションのリストをする`<input>`。 `<input>`に "autocomplete"をします。ユーザーは、オプションのドロップダウンリストをします。

```
<input list="Languages">

<datalist id="Languages">
  <option value="PHP">
  <option value="Perl">
  <option value="Python">
  <option value="Ruby">
  <option value="C+">
</datalist>
```

ブラウザのサポート

クロム	エッジ	Mozilla	サファリ	オペラ
20.0	10.0	4.0	サポートされていません	9.0

オンラインでメニューコントロールをむ <https://riptutorial.com/ja/html/topic/722/>メニューコントロール

クレジット

S. No		Contributors
1	HTMLをいめる	4444 , Abhishek Pandey , aea2002 , ahmednawazbutt , Alexander Wigmore , Alexandre N. , Amanda Ahn , amflare , Amitay Stern , animuson , Anthony Pham , Boris , bwegs , Callan Heard , ChrisD , CocoaBean , Community , Dave Everitt , Dinidu , dippas , dtyler , duskwuff , Eric Dobbs , Firix , FlyingPiMonster , geek1011 , George Bailey , Gerold Broser , H Mirza , H. Pauwelyn , Harish Gyanani , Hemant Kumar , hillary.fraley , Hudson Taylor , ihavemorealts , intboolstring , Isak Combrinck , Jeffrey Lin , JHS , jmmarco , joe_young , John Slegers , Jon Chan , JonasCz , JPB , kelvinelove , Krii , Kurniawantaari , Lahiru Ashan , Lambda Ninja , Léo Martin , Leonidas Menendez , Malcolm , Matt , Matt , MC93 , Michael Moriarty , mnoronha , Muntasir , Nishchay , Ortomala Lokni , Persijn , Prateek , Pyloid , Ryan Hilbert , Shannon Young , sideshowbarker , stark , Stelian Matei , Sunny R Gupta , the12 , tmg , unor , user3130333 , Valor Naram , Willi , Wolfgang , Zaz , zygimantus , Zze
2	CSSでのHTMLの	animuson , bdkopen , Christian Ternus , Community , Euan Williams , feeela , Jones Joseph , Michael Moriarty , Thomas Gerot , thouusten
3	Div	animuson , Araknid , Black Mamba , Chris , Content Solutions , D M , Faust , feeela , Gal Ratzkin , JHS , MySpeed , S.L. Barth , SuperStormer , Thomas Gerot
4	Doctypes	Akshay Anand , Al.G. , Angelos Chalaris , Ani Menon , animuson , Chris , Content Solutions , heerfk , mnoronha , pinjasaur , Right leg , Sumner Evans , Thomas Gerot , tmg , unor
5	HTML 5キャッシュ	Farhad , TricksfortheWeb , Valor Naram
6	HTMLイベント	Paresh Maghodiya
7	HTMLにJavaScriptコードをめる	Alexandre N. , andreaem , animuson , Anselm Urban , Charles , Marjorie Pickard , MervS , Roko C. Buljan , Sildoreth , SuperStormer
8	IFrame	Adjit , Alexandre N. , animuson , ChrisD , dorukayhan , Duh-Wayne-101 , Emanuel Vintilă , J F , Ojen , Wojciech Kazior
9	SVG	andreas , Black Mamba , ChrisD , HerrSerker , Patrickdev ,

		Timothy Miller , w5m
10	アリア	Bhavya Singh , Paul Sweatte , Shannon Young , Travis , unor , user30796
11	アンカーとハイパーリンク	Al.G. , animuson , Anselm Urban , Anthony Pham , ban17 , DawnPaladin , Emil , FlyingPiMonster , insertusernamehere , J F , JHS , joe_young , Jojodmo , Jones Joseph , Lambda Ninja , Matas Vaitkevicius , Nathan Tuggy , Pranav , Prateek , Raystafarian , Robert Columbia , Squidward , Steyn van Esveld , Thomas Gerot , unor , Wolfgang
12	イメージ	Alex , Alexandre N. , andreaem , animuson , Boysenb3rry , Caleb Kleveter , Gabriel Chi Hong Lee , Infuzed guy , ivn , Kake_Fisk , Maximillian Laumeister , Mohd Samir Khan , Mr Lister , Nhan , Shivangi Chaurasia , Stephen Leppik , Wolfgang
13	イメージマップ	animuson , Lambda Ninja , RamenChef
14	キャンバス	cone56 , Richard Hamilton , Roko C. Buljan , tonethar , Trevor Clarke , user4040648
15	クラスとID	Angelos Chalaris , animuson , brandaemon , Caleb Kleveter , Community , Duh-Wayne-101 , Emil , Epodax , Evan , GoatsWearHats , Ingrid Stevens , jhnance , JHS , John Slegers , lexith , Luca Putzu , Michael_B , Natalie , Nhan , Richard Hamilton , Simone Carletti , Thomas Gerot , Timothy , Tyler Zika , unor , Wolfgang , xims
16	グローバルアトリビュート	animuson , Becca , unor , Zange-chan
17	コメント	Ani Menon , animuson , Ashwin Ramaswami , bdkopen , ChrisD , Epodax , JHS , jkdev , RamenChef , Robert Grant , Soaring Code , Squazz , Thomas Gerot , Ulrich Schwarz , Wolfgang
18	コンテンツ	animuson , FelipeAls , Gerold Broser , Isak Combrinck , Muntasir , Shannon Young , unor
19	コンピュータコードをマークアップする	4444 , Naveen Gogineni , Shannon Young , SuperStormer , Tot Zam , unor
20	セクショニング	Andrew Brooke , Anil , animuson , Hanif Formoly , nalply , Shannon Young , SuperBiasedMan , SuperStormer , Yossi Aharon
21	タブインデックス	Content Solutions , Psaniko
22	データ	animuson , Community , Faegy , Infuzed guy , James Donnelly , Manish , Nathan Tuggy , Nhan , Racil Hilan , rajarshig , swatchai ,

		unor , Yasir T
23	テーブル	albert , Alexandre N. , animuson , Cedric Zoppolo , Eduardo Molteni , Grant Palin , J F , j08691 , JHS , joe_young , Lambda Ninja , Mottie , Mr Lister , Nijin22 , Prateek , PrAtik Lochawala , Praveen Kumar , Sildoreth , svarog , Ted Goas , tehciolo , Thomas Landauer , zer00ne
24	テキストフォーマット	animuson , Ben Rhys-Lewis , Emil , gustavohenke , J F , Matas Vaitkevicius , Peter L. , Raystafarian , Stephen Leppik , Thomas Gerot , unor , Wolfgang
25	ナビゲーションバー	Community
26	フォーム	Ali Almoullim , Ani Menon , animuson , Aown Muhammad , Chris Rutherford , Cullub , Gabriel Chi Hong Lee , Greg T , j08691 , Kimmmax , Luca Iangella , Niek Brouwer , Thomas Gerot
27	マークアップの	Content Solutions , mnoronha , unor
28	メタ	Abhishek Pandey , Akshit Soota , Alexander Wigmore , Angelos Chalaris , Ani Menon , animuson , Anselm Urban , Bálint , bdkopen , Bookeater , Boris , coliff , Domenic , geek1011 , Habel Philip , Hafidz Ilham Aji Permana , Himanshu Vaghela , insertusernamehere , jhoanna , JHS , kelvinelove , m_callens , Matt S , Michael Moriarty , Mr. Alien , Nishchay , Ortomala Lokni , Peter O. , Safoor Safdar , Senjuti Mahapatra , Shannon Young , Stas Christiansen , Stephen Leppik , Ted Goas , Thomas Gerot , timmyRS , tmg , unor , VatsalSura , xims
29	メディア	feeela , Isak Combrinck , LisaMM , Shiva , Yossi Aharon
30	ラベル	Anthony Pham , fredden , Josiah Keller , m_callens , Roko C. Buljan
31	リスト	animuson , BiscuitBaker , Daniel Käfer , Grace Note , H. Pauwelyn , Jon Ericson , kcpike , Marvin , Matas Vaitkevicius , platy11 , Prateek , Pseudonym Patel , Richard Hamilton , Right leg , Sayakiss , Stewartside , Thomas Gerot , tmg , Tom Johnson , unor , Zack
32	リソースのリンク	AA2992 , animuson , Anselm Urban , Aravind Suresh , Callan Heard , Chris Rutherford , cone56 , DawnPaladin , Domenic , feeela , Henrique Barcelos , Infuzed guy , JHS , Lambda Ninja , Matas Vaitkevicius , Nhan , Thomas Gerot , unor , V4karian , vladdobra
33		Abhishek Pandey , Al.G. , Alohci , amflare , Amitay Stern , Angelos Chalaris , Ani Menon , animuson , bhansa , Bob , Charlie H ,

		Christophe Strobbe , CN , Community , cone56 , Daniel , DawnPaladin , Dipen Shah , Domenic , Druzion , Edvin Tenovimas , Epodax , Franck Dernoncourt , gabe3886 , geeksal , H. Pauwelyn , Henrique Barcelos , Huy Nguyen , J F , John Slegers , Kashyap Jha , Lahiru Ashan , Lankymart , Magisch , Marvin , Matas Vaitkevicius , Matt , Maximillian Laumeister , Mike McCaughan , morewry , Mosh Feu , Nathan Arthur , Nil Llisterri , Nishchay , niyasc , NoobCoder , Optimiser , Ortomala Lokni , Pi Programs , Pimgd , Prateek , Prav , Praveen Kumar , Psaniko , QoP , RamenChef , Ranjit Singh , Richard Hamilton , Robert Columbia , Roko C. Buljan , SeinopSys , Sharavnan Kv , Shivangi Chaurasia , SJDS , Squazz , Stephen Leppik , Stewartside , Sunny R Gupta , sv3k , the12 , think123 , Thomas Gerot , Timon , tmg , Tot Zam , trungk18 , Undo , vladdobra , zzzzBov
34		J F , Stephen Leppik , zer00ne
35	めみ	Alexandre N.
36	エンティティ	animuson , MervS , stack-learner
37		Abrar Jahin , Thomas Gerot , Valor Naram
38		4444 , ChrisD , Thomas Gerot , unor
39	し	Ani Menon , animuson , dippas , Evan , Infuzed guy , joe_young , MervS , Nathan Arthur , Pseudonym Patel , sasha , Thomas Gerot , unor , V-Kopio
40		animuson , Content Solutions , Richard Hamilton
41	メニューコントロール	Ali Almoullim , amflare , animuson , GentlePurpleRain , Ilyas karim , Mosh Feu , Tot Zam