



HTML5プロフェッショナル認定試験 レベル1 ポイント解説無料セミナー

2016年9月24日

株式会社富士通ラーニングメディア

結城陽平 抜山雄一



富士通ラーニングメディアのご紹介

1. 会社概要

設立	1977年6月30日
資本金	3億円（全額 富士通株式会社）
売上高	86億円（2015年度）
従業員	427名（2016年3月末現在）
事業内容	人材育成・研修サービス(公開コース 1,250 コース、年間 80,000 名受講) 個人のお客様向けパソコン教室（富士通オープンカレッジ）
関係会社	株式会社富士通ラーニングメディア沖縄（研修サービス・研修サービスサポート） 株式会社富士通ラーニングメディア・スタッフ（人材派遣）
出資会社	株式会社アクト・ブレーン・ベトナム（ソフトウェア開発など）
事業所 関連施設	東京・名古屋・大阪・沖縄に5拠点（約 40 教室、 900 名以上の定員数）



品川本社



品川LC



名古屋事業所/LC



関西事業所/LC



沖縄事業所/LC

2. 富士通ラーニングメディアが目指すこと

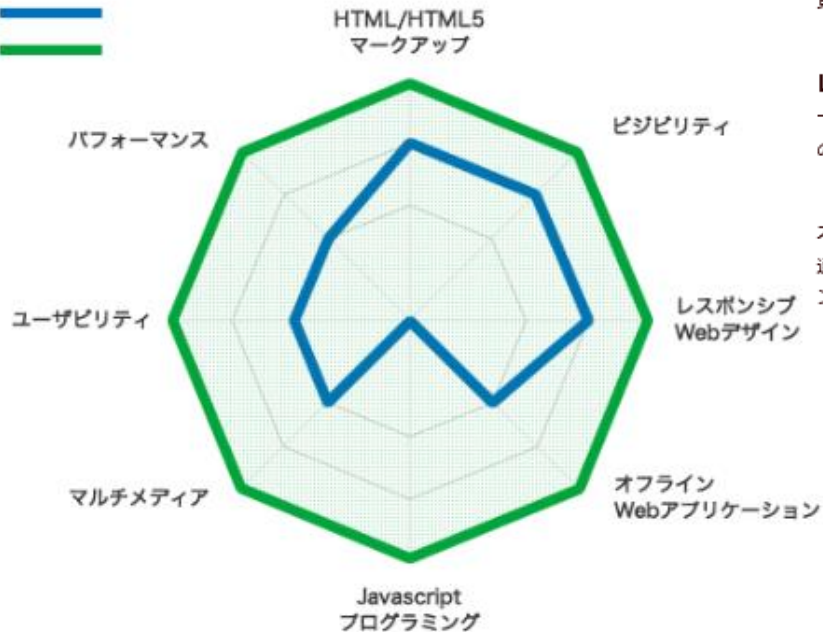
ICT人材を支える人材育成のパートナーでありつづける

- 最高水準の『知』のサービスを提供することにより、お客様の真のパートナーを目指します。
- お客様の「成長のスパイラル」をサポートします。

HTML5 Professional Certification レベル1



Level.1 
Level.2 



HTML/HTML5マークアップ

HTML5に関するタグの用途、構造の組み立て方に関する技術

Javascriptプログラミング

Javascriptを使って、動的なWebコンテンツを作成する技術

ビジビリティ

JavascriptやCSS3などを用いて、デザイン仕様に沿った見やすい表示を行うための技術

マルチメディア

3D・動画・音声ファイルなどのマルチメディアコンテンツの表示・再生に関する技術

レスポンスWebデザイン

一つのソースで、スマートフォンなどの様々なデバイスの画面サイズに対応させるための技術

ユーザビリティ

ナビゲーション、地図表示など操作しやすいコンテンツを作成するための技術

オフラインWebアプリケーション

通信が常時接続状態ではない環境でも、効率的にWebコンテンツを動作させるための技術

パフォーマンス

データベースや、並列処理を使ってコンテンツを効率良く高速に動作させるための技術



UI作成のためのWeb標準技術

■ Webページの文書構造とスタイル

HTMLはWebページの文書構造を決める。CSSはWebページのデザインをする。動作を付けるためには、JavaScriptが必要となる。

■ W3Cによる標準化

HTML5およびCSS3はW3Cによって標準化された仕様である。そのため、ブラウザ間の互換性がある程度保たれている。

■ Webページ作成やデスクトップアプリケーションを作成できる

HTMLとCSSを使用して、ブラウザ上に表示できるWebページを作成可能。また、環境としてElectronを使用すれば、クロスプラットフォームで実行可能なデスクトップアプリケーションを作成することもできる。

■ HTML 4.01

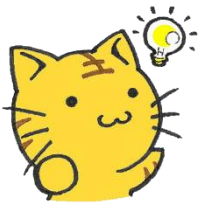
1999年12月勧告。1997年に勧告された HTML 4.0に微調整を加えたバージョン。

■ HTML5

2014年10月勧告。セマンティック要素やマルチメディア対応の追加など、前バージョンから大幅な機能強化が行われたバージョン。

■ HTML5.1

2016年9月勧告予定。picture要素やdetail要素、sortable属性などを追加した小幅なバージョンアップ。



すでにHTML5.2の策定も開始されているよ。
今後は早いサイクルで、機能強化を進める予定なんだ。
新技術は使うときはブラウザの対応状況に注意してね。

■ CSS 2.1

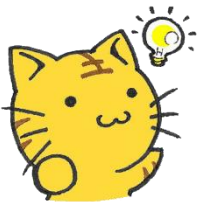
2011年6月勧告。1998年に勧告されたCSS2の曖昧な使用や実装されなかった仕様を削除している。

■ CSS3

アニメーションやセレクタの強化など大幅な機能追加が行われているバージョン。正確には、CSS3というバージョンは存在せず、機能ごとに仕様策定が進められている。それらを総称してCSS3と呼んでいる。

■ CSS4

CSS3で確定しなかった仕様などがCSS4で検討される予定。



CSSは仕様策定のスピードアップのため、バージョンはなくなったんだ。特定時点での策定状況は「スナップショット」として公開されているよ。



HTML5仕様策定のコンセプト

HTML5 SEMANTICS

OFFLINE & STORAGE

DEVICE ACCESS

CONNECTIVITY

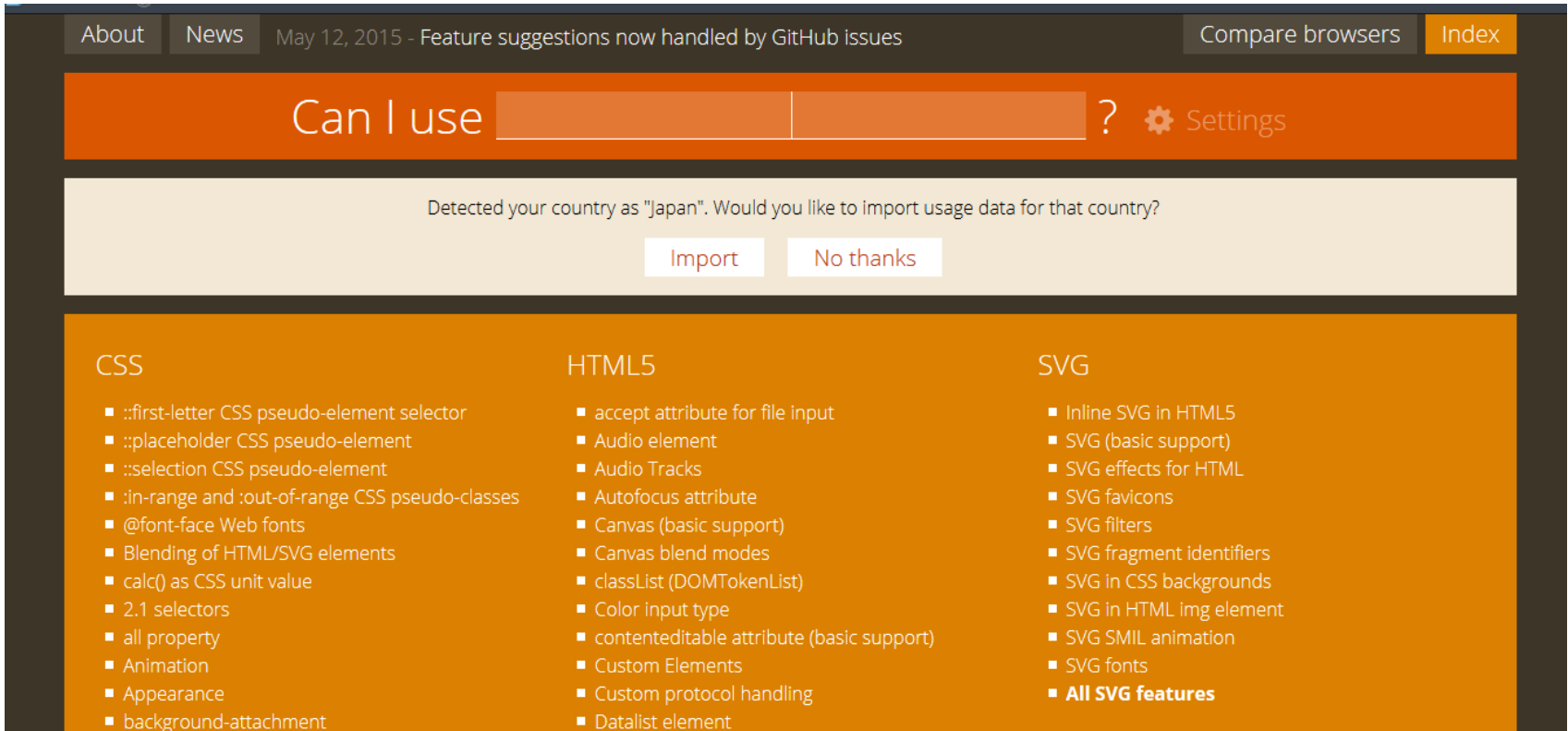
MULTIMEDIA

3D, GRAPHICS, EFFECTS

PERFORMANCE &
INTEGRATION

CSS3 STYLING

【参考】最新技術を使うために 1



The screenshot shows the Can I use website interface. At the top, there are navigation links: About, News, May 12, 2015 - Feature suggestions now handled by GitHub issues, Compare browsers, and Index. Below these is a search bar with the text "Can I use" and a question mark icon, followed by a "Settings" link. A message box states: "Detected your country as 'Japan'. Would you like to import usage data for that country?" with "Import" and "No thanks" buttons. The main content area is divided into three columns: CSS, HTML5, and SVG. Each column contains a list of features and their support status.

CSS	HTML5	SVG
<ul style="list-style-type: none">■ ::first-letter CSS pseudo-element selector■ ::placeholder CSS pseudo-element■ ::selection CSS pseudo-element■ :in-range and :out-of-range CSS pseudo-classes■ @font-face Web fonts■ Blending of HTML/SVG elements■ calc() as CSS unit value■ 2.1 selectors■ all property■ Animation■ Appearance■ background-attachment	<ul style="list-style-type: none">■ accept attribute for file input■ Audio element■ Audio Tracks■ Autofocus attribute■ Canvas (basic support)■ Canvas blend modes■ classList (DOMTokenList)■ Color input type■ contenteditable attribute (basic support)■ Custom Elements■ Custom protocol handling■ Datalist element	<ul style="list-style-type: none">■ Inline SVG in HTML5■ SVG (basic support)■ SVG effects for HTML■ SVG favicons■ SVG filters■ SVG fragment identifiers■ SVG in CSS backgrounds■ SVG in HTML img element■ SVG SMIL animation■ SVG fonts■ All SVG features

「Can I use」

<<<http://caniuse.com/>>>

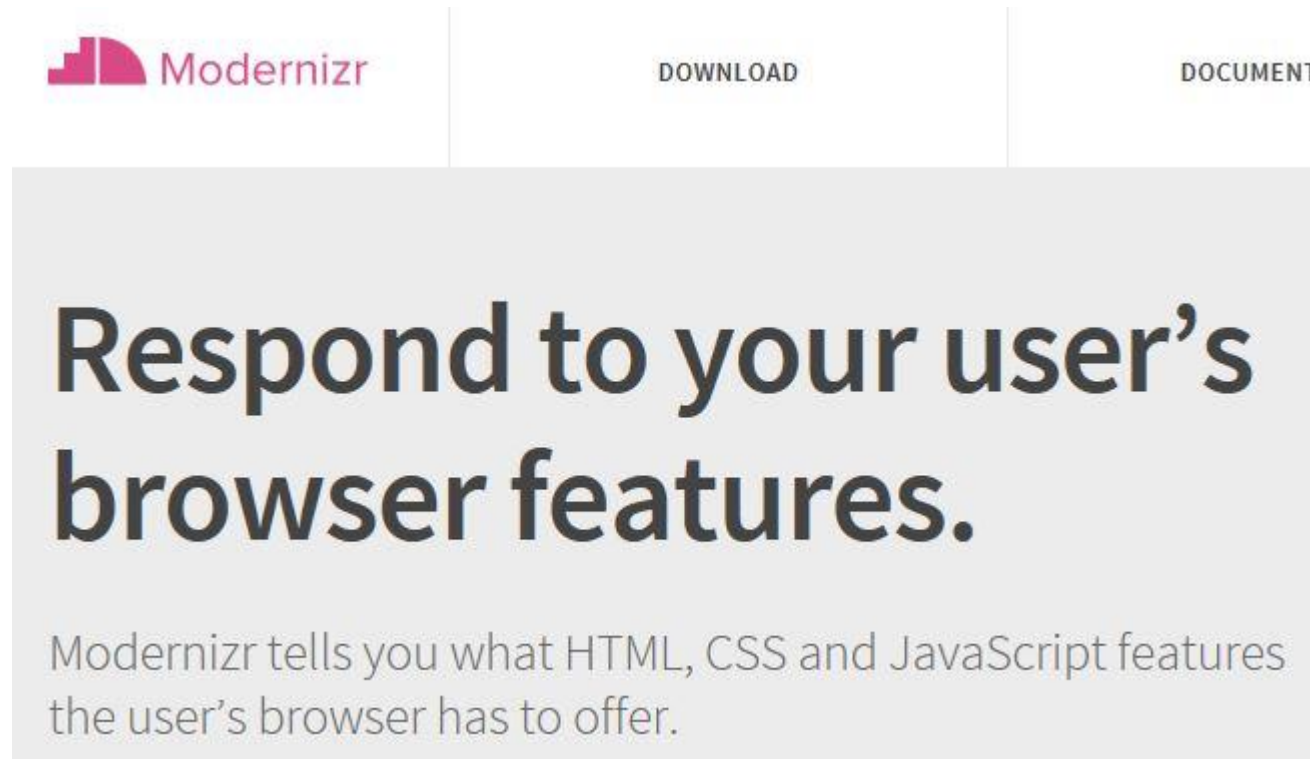
ブラウザごとの対応状況を公開しているサイト

【参考】最新技術を使うために 2



「Mozilla Developer Network」
<<<https://developer.mozilla.org/ja/>>>
Web標準技術に関する情報が充実している。

【参考】最新技術を使うために 3



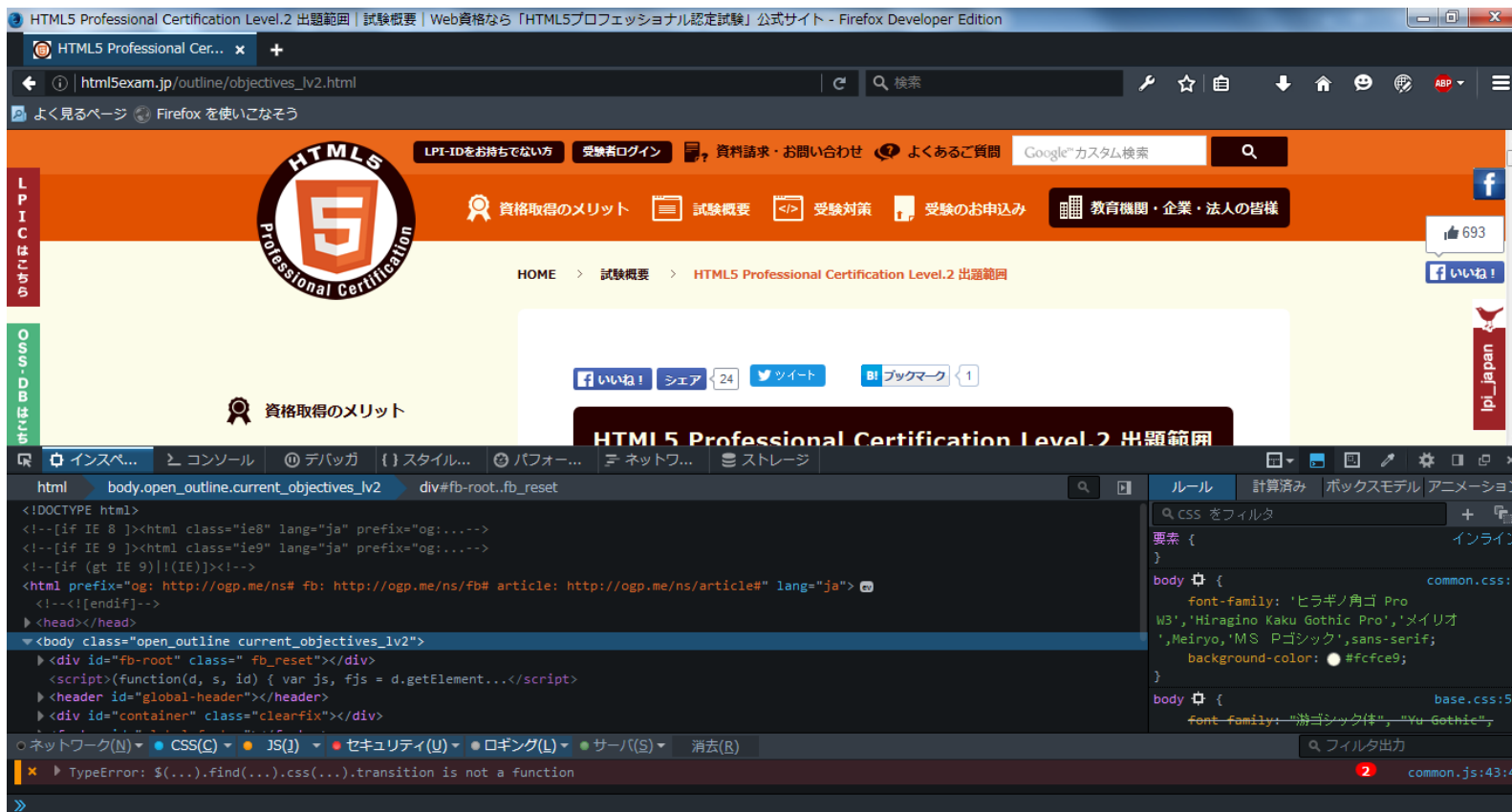
「Modernizr」

<<<https://modernizr.com/>>>

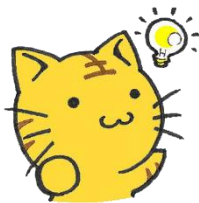
ユーザーが使用しているブラウザのHTML/CSS/JavaScriptの対応状況を調査する。
調査結果に基づいてポリフィルを記述できる。



開発ツール



各種ブラウザで「F12」キーを押すと、開発ツールが開くよ。
HTML/CSSの調整に役立つよ。



1.1.1 HTTP, HTTPSプロトコル

1.1.1 HTTP, HTTPSプロトコル	
重要度	★★★★★★ 7
出題種別	<ul style="list-style-type: none">知識問題記述問題
説明（望まれるスキル）	<p>HTTPのコンテンツ作成や、サイト全体の設計を行うために必要なHTTPおよびHTTPSプロトコルに関する知識を有している。</p> <p>また、ブラウザでアクセスした時に返ってくるエラーコードの意味を理解できて、問題を解決するヒントとする事ができる。</p>

HTTP1.1



ブラウザ

HTMLをリクエスト

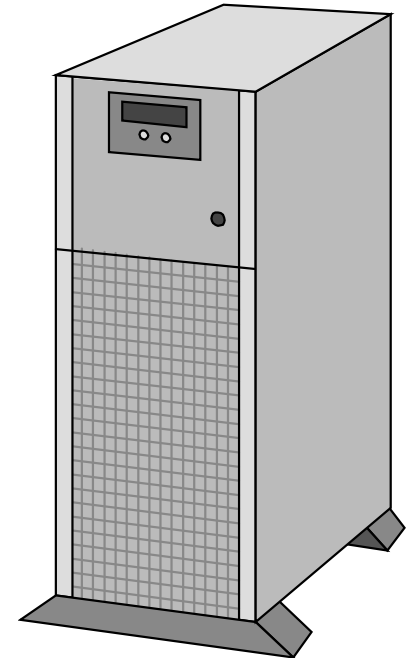
HTMLをレスポンス

CSSをリクエスト

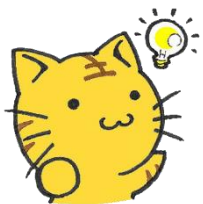
CSSをレスポンス

画像をリクエスト

画像をレスポンス



Webサーバ



Webページに埋め込まれたファイルごとにリクエスト/レスポンスを繰り返しているんだ!!そのせいでファイル数が多いと低速になっちゃう…。(モダンブラウザならパイプライン処理で6多重程度はできるけどね。)

■ 画像ファイルをCSSスプライトで減らす

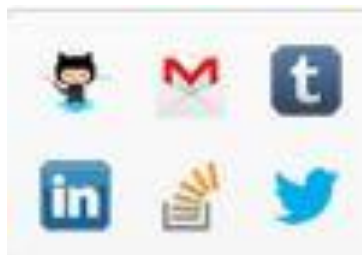
CSSスプライトとは、複数の画像ファイルをまとめることで、画像ファイル数を減らす手法である。HTTPによるリクエスト数を減らすことができる。まとめた画像はCSSで分割する。

■ CSSファイルやJSファイルを結合&圧縮する

CSSファイルやJSファイルをまとめてファイル数を減らす。また、JSファイル内の改行などをなくすことでファイルサイズを小さくする（圧縮）。

■ gzipでファイルサイズを減らす

gzipはWebサーバの圧縮技術でファイルサイズを削減できる。



CSSスプライト用アイコン

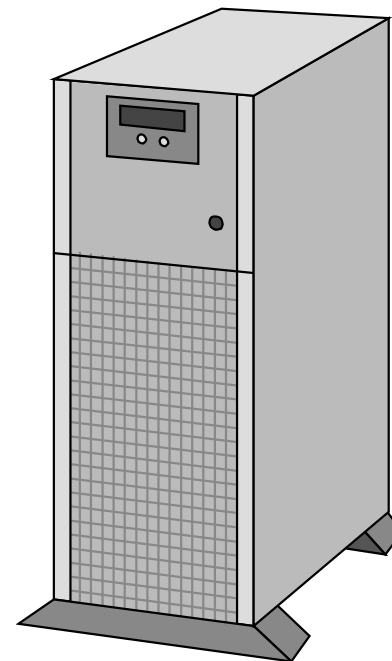
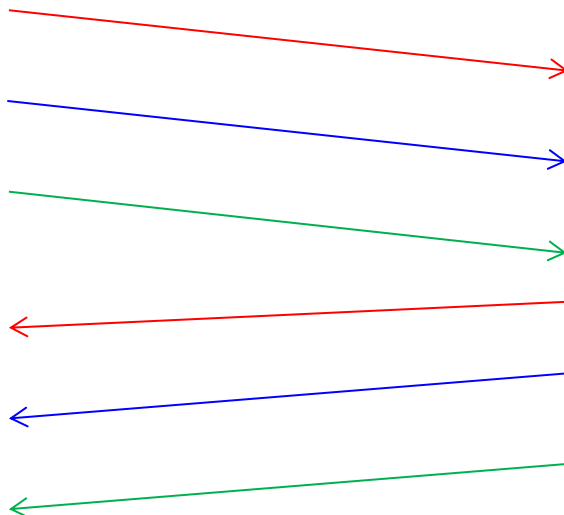
```
1 /*! jQuery v1.7.1 jquery.com | jquery.org/license */
2 (function(a,b){function cy(a){return f.isWindow(a)?a:a.nodeType
{var b=c.body,d=f("<"+a+">").appendTo(b),e=d.css("display");d
cl.frameBorder=cl.width=cl.height=0},b.appendChild(cl);if(!cm
```

JSファイルの圧縮

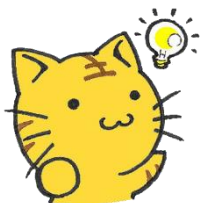
HTTP2



ブラウザ



Webサーバ



結果を待たずにリクエスト/レスポンスできるから高速!!!!!!
多重に通信するのでCSSスプライトなどの必要性が落ちるよ。
ブラウザとWebサーバ両方がHTTP2に対応してないとだめだよ。
仕様には含まれてないけど、ブラウザベンダの方針でHTTPSも必須だよ。



1.3.2 HTML5で新しく加わった要素および属性

1.3.2 HTML5で新しく加わった要素および属性	
重要度	★★★★★★★★★★ 10
出題種別	<ul style="list-style-type: none">• 知識問題• コードリーディング問題• 記述問題
説明（望まれるスキル）	HTML5で新しく加わった要素や属性に関して、仕様に沿ったHTMLコードを正しく記述する事ができ、要件を受けてどのようなコードを記述するのが適切か判断する事ができる。

コンピュータがWebページの意味を分かるようになるマークアップ

HTML 4.01

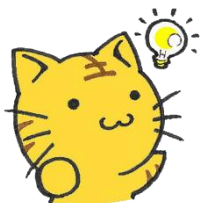
1. `<div id="header"></div>`
2. `<div id="main"></div>`
3. `<div id="footer"></div>`



新しいタグで意味が分かるように

HTML5

1. `<header id="header"></header>`
2. `<main id="main"></main>`
3. `<footer id="footer"></footer>`



スクリーンリーダーの精度向上や
検索エンジンのSEO対策にも役立つんだ!!

タグ	説明
<section>	文章のセクションを表す。子要素にh1-6要素を含める。
<article>	記事を表す。子要素にh1-6要素を含める。
<aside>	メインコンテンツと関連があまりないことを表す。サイドバーや広告を含める。
<hgroup>	h要素をまとめる。HTML5の仕様から外されたが各種ブラウザで動作する。
<header>	ヘッダーを表す。
<footer>	フッターを表す。
<nav>	リンク集を表す。
<figure> <figcaption>	メインコンテンツと関連の薄い図像を表す。

■ セマンティックとして最適な要素を に記述しなさい。

1. <section>
2. <h2>News一覧</h2>
3.
4. <h3>HTML5.1の勧告間近!?!</h3>
5. <p>W3Cの発表によると…</p>
6.
7. </section>

- セマンティックとして最適な要素を に記述しなさい。

1. `<section>`
2. `<h2>News一覧</h2>`
3. `<article>`
4. `<h3>HTML5.1の勧告間近!?!</h3>`
5. `<p>W3Cの発表によると…</p>`
6. `</article>`
7. `</section>`

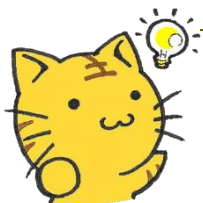
1つ1つが記事にあたるので、`<article>`が最適だよ。
`<section>`要素に入れ子することももちろんできるよ。



アドオン（Flash）を使用せずに動画再生できる

1. `<video loop poster="movies/brandBanner.png">`
2. `<source src="movies/top.mp4">`
3. `<source src="movies/top.webm">`
4. `<source src="movies/top.ogv">`
5. `<track kind="subtitles" src="subtitles.vtt" srclang="ja">`
6. `</video>`

ブラウザごとにサポートする動画フォーマットが異なるので、
動画の指定にはsource要素を複数配置しているよ!!
track要素を使えば、字幕も出せる!!



属性	説明
autoplay	自動再生する。
controls	再生や停止ができるコントロールを表示する。
loop	動画再生をループする。
muted	既定値が消音になる。
preload	動画の事前読み込み設定。autoplay属性を設定していると無視される。
poster	動画再生前の画像を指定する。
src	再生する動画を指定する。source要素を使用している場合は、省略可能。

サンプル問題 2

- 以下のソースコードの実行結果の説明として、正しいものを選びなさい。

```
1. <video preload="none" controls loop>
2.   <source src="movies/first.mp4">
3.   <source src="movies/second.webm">
4.   <source src="movies/third.ogv">
5.   <track kind="subtitles" src="subtitles.vtt" srclang="ja">
6. </video>
```

- A) firstファイルの再生終了後、secondファイルの再生が始まる。
- B) preload属性がnoneのため、firstファイルはロードされない。
- C) track要素を指定すると、再生時間が表示される。
- D) 既定値を音消しにする場合、silence属性を追加する。
- E) ブラウザが対応するフォーマットの動画が1つのみ再生される。

- 以下のソースコードの実行結果の説明として、正しいものを選びなさい。

```
1. <video preload="none" controls loop>
2.   <source src="movies/first.mp4">
3.   <source src="movies/second.webm">
4.   <source src="movies/third.ogv">
5.   <track kind="subtitles" src="subtitles.vtt" srclang="ja">
6. </video>
```

A) firstフォーマットの再生終了後、secondフォーマットの再生が始まる。

source要素は上から順に確認されて、最初に見つかったブラウザがサポートしているフォーマットのものだけが再生されるよ。

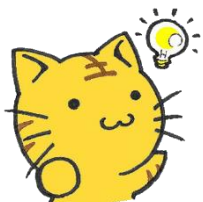
D) 既定値を音消しにする場合、silence属性を追加する。

☒ E) ブラウザが対応するフォーマットの動画が1つのみ再生;



input要素の種類

type属性値	説明
color	色を入力する。
date	日付を入力する。
datetime-local	タイムゾーンがない日付を入力する。
email	メールアドレスを入力する。
month	付きを入力する。
number	数字を入力する。
range	一定範囲から値を入力する。
search	メインコンテンツと関連の薄い図像を表す。
tel	電話番号を入力する。
time	時刻を入力する。
url	URLを入力する。



見栄えや動作が変わるものと、セマンティクスが付与されるだけのものがあるよ!!未対応ブラウザだとtext扱いになるんだ。

input要素で使用する便利な属性

type属性値	説明
autofocus	フォーカスを合わせる。
form	submitボタンのフォームを指定する。
formaction	フォームの送信先を指定する。
formmethod	フォームのメソッドを指定する。
max	最大値を指定する。
min	最小値を指定する。
pattern	入力形式を指定する。
placeholder	プレースホルダーを指定する。

```
1. <div>
2.   <label for="color">color</label>
3.   <input type="color" id="color">
4. </div>
5. <div>
6.   <label for="range">range</label>
7.   <input type="range" id="range" min="1" max="10">
8. </div>
9. <div>
10.  <label for="search">search</label>
11.  <input type="search" id="search" required>
12.</div>
```

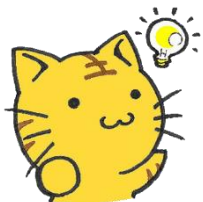


colorなどはブラウザ対応状況にばらつきがあるから要注意だ!

【参考】HTML5仕様から漏れた要素や属性

要素/属性	説明
command要素	廃止。
hgroup要素	廃止。WHAT-WGでは継続
keygen要素	廃止。
details要素	HTML5.1に先送り。
contextmenu 属性	HTML5.2に先送り。
datetime-local 属性	WHAT-WGで廃止。HTML Specでは継続中。

勧告されていない仕様は変更の可能性があるよ。
新技術を追いかけるときは注意してね。

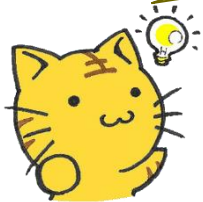


【参考】 picture要素

画面サイズなどに応じて画像を切り替えられる要素

1. `<picture>`
2. `<source srcset="images/koban.jpg" media="(min-width: 600px)">`
3. ``
4. `</picture>`

画面サイズが600px以上なら、koban画像が、それ以下ならsnowman画像が表示されるよ。デスクトップ向けには、高解像度画像を、スマホ向けには低解像度画像を送るといった使い分けができるよ。



サンプル問題 3

- 以下のWebページを実行した際の説明のうち、正しいものを**すべて**選択しなさい。

```
1. <form id="fm" method="get">
2.   <label for="date">date</label>
3.   <input type="date" id="date" name="d">
4. </form>
5. <input type="submit" value="submit" form="fm"
   formaction="b.html" formmethod="post">
```

- A) 日付のパラメータ名は"d"である。
- B) submitボタンをクリックするとpostメソッドで送信される。
- C) ブラウザがtype="date"に対応していない場合、非表示になる。
- D) form要素の外にinput要素があるため、データを送信できない。
- E) form要素のaction属性は送信方法を指定する。

- 以下のWebページを実行した際の説明のうち、正しいものを**すべて**選択しなさい。

```
1. <form id="fm" method="get">
2.   <label for="date">date</label>
3.   <input type="date" id="date" name="d">
4. </form>
5. <input type="submit" value="submit" form="fm"
   formaction="b.html" formmethod="post">
```

- ☒ A) 日付のパラメータ名は"d"である。
- ☒ B) submitボタンをクリックするとpostメソッドで送信される。
- ☐ C) ブラウザで表示されない場合、非表示
- ☐ D) form要素でデータを送信で
- ☐ E) form要素で送信する。

form属性を指定するとform要素外にもボタンを配置できるから、柔軟性が高い!!
formaction属性やformmethod属性を指定するとformの設定を上書きするよ。



1.3.3 HTML5で廃止されたタグおよび属性

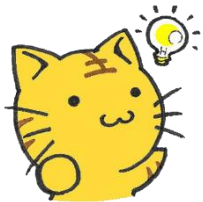
1.3.3 HTML5で廃止されたタグおよび属性	
重要度	★★★★★ 5
出題種別	<ul style="list-style-type: none">• 知識問題• コードリーディング問題• 記述問題
説明（望まれるスキル）	HTML5で廃止されたタグや属性を知っており、HTML5の仕様に沿った場合代わりにどのような記述をすればいいのかを理解している。

廃止された要素/属性

要素/属性	説明
acronym要素	略語であることを示す。
big要素	文字の大きさを変更する。
center要素	要素を中央寄せする。
font要素	フォントを変更する。
frame要素	Webページにインラインフレームを埋める。
div要素などのalign属性	要素内の文字列を整列する。
hr要素などのwidth属性	要素の幅を指定する。

廃止された要素は以下の観点で踏まえて覚えるといいよ。

- あまり使われなかったもの（acronym要素など）
- 見栄え設定にかかわるもの（center要素など）
- セキュリティ上の問題がもの（frame要素など）



1.2.1 スタイルシートの基本

1.2.1 スタイルシートの基本	
重要度	★★★★★★ 6
出題種別	<ul style="list-style-type: none">• 知識問題• コードリーディング問題• 記述問題
説明（望まれるスキル）	大規模なサイトやコンテンツにおいても、見やすく効率的なコードを記述し、複数のページで共有させるために必要な記述を適切に行うことができる。

要素の特定の状態を表す

擬似クラス	説明
:first-child	自分の親要素の最初の子要素に適用する。
:last-child	自分の親要素の最後の子要素に適用する。
:nth-child(an + b)	an+b-1 個の兄弟要素を持ち、親要素を持つ要素にマッチする。
:nth-of-type(an + b)	同じ要素名の an+b-1 個の兄弟要素を持ち、親要素を持つ要素とマッチする。
:read-only	読み取り専用の要素にのみ適用する。
:in-range	inputなどの入力値がmin-maxの範囲内なら適用する。
:out-of-range	inputなどの入力値がmin-maxの範囲外なら適用する。

```
1. li:first-child, li:last-child {  
2.     list-style-type: square;  
3. }  
4. input[type="number"]:in-range {  
5.     background-color: rgba(0, 255, 0, 0.25);  
6. }  
7. input[type="number"]:out-of-range {  
8.     background-color: rgba(255, 0, 0, 0.25);  
9.     border: 2px solid red;  
10.}
```



要素名での指定は要素セレクタと言って、同じ要素すべてにCSSを適用するセレクタなんだ。属性の値で設定したいときは、[] を使うよ。
また、カンマ (,) を使えば同じ値を複数のセレクタにかけられる。

サンプル問題 4

- 以下のWeb ページにCSSを適用した場合、span2の背景色は何色になるか記入しなさい。なお、既定の背景色は白とします。

```
1. <p id="pseudo">  
2.   <span>span1 </span><em>em </em>  
3.   <span>span2 </span><span> span3</span>  
4. </p>
```

- A) #pseudo span:nth-of-type(2n + 1) {
B) background-color: red;
C) color: white;
D) }

解答：

- 以下のWeb ページにCSSを適用した場合、span2の背景色は何色になるか記入しなさい。なお、既定の背景色は白とします。

```
1. <p id="pseudo">  
2.   <span>span1 </span><em>em </em>  
3.   <span>span2 </span><span> span3</span>  
4. </p>
```

- A) #pseudo span:nth-of-type(2n + 1) {
B) background-color: red;
C) color: white;
D) }

解答 : 白



1.2.2 CSSデザイン

1.2.2 CSSデザイン	
重要度	★★★★★★★★★ 9
出題種別	<ul style="list-style-type: none">• 知識問題• コードリーディング問題• 記述問題
説明（望まれるスキル）	要件に沿ったデザインをCSSを利用して実現する際に、どのような実現方法が適切か判断でき、仕様に沿った正しいコードを記述することができる。

【参考】ベンダプレフィックス

ブラウザベンダの独自機能、または実験実装であることを表す接頭辞

ベンダプレフィックス	ブラウザ
-moz-	Firefox
-ms-	Internet Explorer / Edge
-o-	Opera
-webkit-	Chrome / Safari

1. `background:-moz-linear-gradient(...);`
2. `background:-webkit-linear-gradient(...);`
3. `background:linear-gradient(...);`

複数ブラウザ対応と保守性を考えるとベンダプレフィックスありとなし、両方の記述をした方がよい。
ブラウザベンダはベンダプレフィックス付きの実験実装をやめる方針だから、将来的には不要になるはずだよ。



複数列による段組みレイアウトを設定する

プロパティ	説明
column-count	列数を指定する。
column-fill	コンテンツをどのように列に分けるかを決める。 autoやbalanceを指定できる。
column-gap	列間隔を指定する。
column-rule	column-rule-width、column-rule-style および column-rule-colorをまとめて設定できる。
column-rule-color	列間の色を指定する。
column-rule-style	列間の罫線の形式を指定する。値はborder- styleと同じ。
column-rule-width	列間の罫線の幅を指定する。
column-span	値がallのとき、その要素がすべての列にまたが るように表示される。
column-width	列幅を指定する。

```
1. #multi {  
2.     column-count: 3;  
3.     column-fill: balance;  
4.     column-gap: 3px;  
5.     column-rule: 1px dotted red;  
6. }
```

multi-column

maltimaltimaltimaltimaltimaltimaltimaltimaltimalti
maltimaltimaltimaltimaltimaltimaltimaltimaltimalti

maltimaltimaltimaltimaltimaltimaltimaltimaltimalti
maltimaltimaltimaltimaltimaltimaltimaltimaltimalti

maltimaltimaltimaltimaltimaltimaltimaltimaltimalti
maltimaltimaltimaltimaltimaltimaltimaltimaltimalti

ボックスモデルを柔軟に扱う

プロパティ	説明
display	flexを指定することで可変ボックスレイアウトを使用できる。
flex-direction	レイアウトの方向を指定する。row（文字方向と同一）やrow-reverse（文字方向と逆）などを指定できる。
flex-wrap	可変ボックス内のレイアウトを1行に収めるか（nowrap）折り返しても良いか（wrap）を指定する。
order	可変ボックス内の順序を指定する。
justify-content	可変ボックス内のアイテムの配分を指定する。始端から詰める（flex-start）や終端から詰める（flex-end）、中央から詰める（center）などの指定ができる。

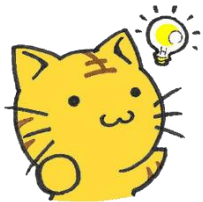
```
1. #flex {  
2.     display: flex;  
3.     flex-direction: row;  
4.     justify-content: space-around;  
5. }
```

flexible box

item1 item1 item1 item1 item1 item1 item1 item1 item1 item1 item1 item1 item1 item1 item1

item2 item2 item2 item2 item2 item2 item2 item2 item2 item2 item2 item2 item2 item2 item2

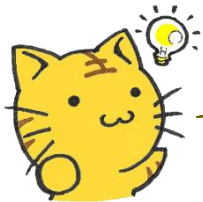
従来のfloatを使った回り込みより簡単に設定できるんだ!
ちなみに、space-aroundは均等割り付けだよ。



要素を変形させる

値	説明
translate(x,y)	要素をx軸、y軸に指定した分だけ移動させる。単位はpxを使用する。
scale(n)	指定した分だけ要素を拡大/縮小する。
skew(ax)	指定した分だけ要素をゆがめる。単位はdegを使用する。
rotate(angle)	指定した分だけ、要素を回転させる。単位はdegを使用する。

これらの値のことをCSS関数と呼ぶよ。関数は()に渡した値で何らかの処理をするんだ。~()という書き方をしているから関数の判別は簡単だよ。





transformの使用例

```
1. #tran {  
2.     height: 100px;  
3.     width: 100%;  
4.     font-size: 2em;  
5.     color: white;  
6.     background-color: black;  
7.     transform: skew(1deg, 1deg) rotate(1deg);  
8. }
```

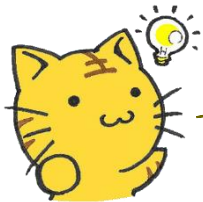
transform

transform

プロパティの変化速度を設定する

プロパティ	説明
transition	以下、4つのプロパティをまとめて指定する。
transition-delay	変更が起こるまでの遅延時間を指定する。
transition-duration	変化時間を指定する。
transition-property	トランジションを適用するプロパティ名を指定する。
transition-timing-function	トランジションの変化曲線を指定する。値としてeaseやease-inなどが指定できる。

トランジションを使うには、擬似セレクタ（:hoverなど）やJavaScriptで対象のプロパティの値が変化させる必要があるよ。





マルチカラムレイアウトの使用例

```
1. #transition: hover {  
2.     height: 100px;  
3.     width: 100%;  
4.     color: blue;  
5.     background-color: black;  
6.     transform: skew(30deg, 10deg) rotate(15deg);  
7.     transition-property: transform, background-color;  
8.     transition-duration: 5s;  
9.     transition-timing-function: ease-in-out;  
10.    transition-delay: 1s;  
11. }
```


CSSによるアニメーションを設定する

プロパティ	説明
@keyframes	アニメーションのキーフレーム（通過点）を指定する。
animation	以下、6つのプロパティをまとめて指定する。
animation-delay	アニメーションの遅延時間を指定する。
animation-direction	アニメーションの方向を指定する。alternate（正逆どちらも再生）やreverse（逆再生）などを指定できる。
animation-duration	アニメーションの実行時間を指定する。
animation-fill-mode	アニメーション実行後のCSSプロパティの状態を指定する。backward（アニメーション前に戻す）やforward（アニメーションで適用したCSSを適用したままにする）などを指定できる。
animation-iteration-count	アニメーションの実行回数を指定する。infinityを指定するとループする。
animation-name	適用するアニメーションのキーフレーム名を指定する。
animation-play-state	アニメーションがpaused（停止）かrunning（実行中）かを示す。
animation-timing-function	アニメーションの変化曲線を指定する。値としてeaseやease-inなどが指定できる。

マルチカラムレイアウトの使用例

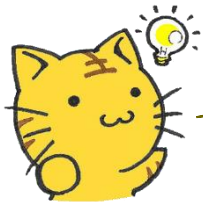
```
1. @keyframes sample {  
2.   0% {transform: rotate(0deg);}  
3.   50% {transform: rotate(180deg);}  
4.   100% {transform: rotate(360deg) translate(300px);}  
5. }  
6. #animation {  
7.   width: 100px;  
8.   height: 100px;  
9.   background-color: red;  
10.  border-radius: 3px;  
11.  animation-name: sample;  
12.  animation-delay: 3s;  
13.  animation-direction: alternate;  
14.  animation-duration: 3s;  
15.  animation-iteration-count: infinite;  
16.  animation-timing-function: linear;  
17. }
```

【参考】CSS変数

プロパティの値を格納し、様々なところで使用できる

```
1. :root {  
2.   --brand-color: #FF0000;  
3. }  
4. #variable {  
5.   color: var(--brand-color);  
6. }
```

変数名は--*の形式でなければいけないよ。CSS変数は何度でも使い廻せるから、CSSファイルのメンテナンス性や可読性を向上できるよ。

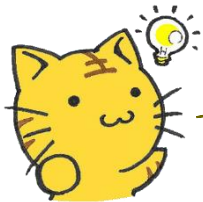


【参考】縦書きレイアウト

文字表記を縦に変更できる

```
1. #writing{  
2.   writing-mode: vertical-rl;  
3.   text-orientation: upright;  
4.   line-height: 1.2em;  
5. }
```

writing-modeをvertical-rlに設定すると縦書きになるよ。text-orientationをuprightにすると、アルファベットの向きも変更できる!!



サンプル問題 5

■ 以下のHTMLを表示した場合の説明として、正しいものを選びなさい。

1. `<div style="color:red;color:yellow;transition-property:color">`
2. `char`
3. `</div>`

- A) 文字が黒で表示される。
- B) 文字が赤で表示される。
- C) 文字が黄で表示される。
- D) 文字が黒→赤→黄の順で変化して表示される。
- E) 文字が赤→黄の順で変化して表示される。

■ 以下のHTMLを表示した場合の説明として、正しいものを選びなさい。

1. `<div style="color:red;color:yellow;transition-property:color">`
2. `char`
3. `</div>`

- A) 文字が黒で表示される。
- B) 文字が赤で表示される。
- ☒ C) 文字が黄で表示される。
- D) 文字が黒→赤→黄の順で変化して表示される。
- E) 文字が赤→黄の順で変化して表示される。

CSSで同じ値を指定した場合、優先順位が高い方が適用されるんだ。今回だと後から書いたものが後順位になるよ。
transitionは何らかの状態で値が変化しないと適用されないよ。



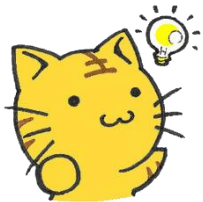
1.4.1 マルチデバイス対応ページの作成

1.4.1 マルチデバイス対応ページの作成	
重要度	★★★★ 4
出題種別	<ul style="list-style-type: none">• 知識問題• コードリーディング問題• 記述問題
説明（望まれるスキル）	要件に沿ったページをデザイン・設計する際に、どのような画面サイズであってもデザインが仕様どおりになるようなページの実現方法を理解しており、マルチデバイス対応のページを作る事ができる。

ブラウザの仮想サイズの指定する

1. `<head>`
2. `<meta name="viewport" content="width=device-width,initial-scale=1.0">`
3. `</head>`

スマートフォンのブラウザは、デバイス幅より広いブラウザサイズがあると認識しているんだ。
viewportのwidthをdevice-widthに指定することで、デバイス分しか幅がないことを指定できるんだ。



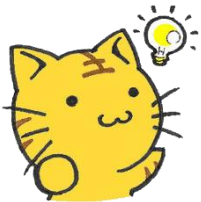
1.4.2 メディアクエリ

1.4.2 メディアクエリ	
重要度	★★★★ 4
出題種別	<ul style="list-style-type: none">• 知識問題• コードリーディング問題• 記述問題
説明（望まれるスキル）	メディアクエリを利用して、画面サイズなどの様々な環境に合わせて表示を変えるページを作成することができる。

メディアタイプやブラウザ幅に応じて適用するCSSを切り替える

```
1. @media only screen and (max-width: 600px) {  
2.     #media {  
3.         color: red;  
4.     }  
5. }
```

メディアタイプでCSSを適用するメディアを指定できる。ブラウザのメディアタイプはscreenだよ。そのほか、すべて（all）やプリンタ（print）などがあるんだ。
max-widthは、このCSSが適用される最大幅になるよ。最小幅で指定したいなら、min-widthも使えるよ。



サンプル問題 6

■ 以下のメディアクエリのうち、正しく動作するものを**すべて**選びなさい。

- A) `@media only screen and (max-width: 600px) {}`
- B) `@media only screen and (min-width: 600px) {}`
- C) `@media only screen or (max-width: 600px) {}`
- D) `@media only print {}`
- E) `@media only screen and (orientation:landscape) {}`

■ 以下のメディアクエリのうち、正しく動作するものを**すべて**選びなさい。

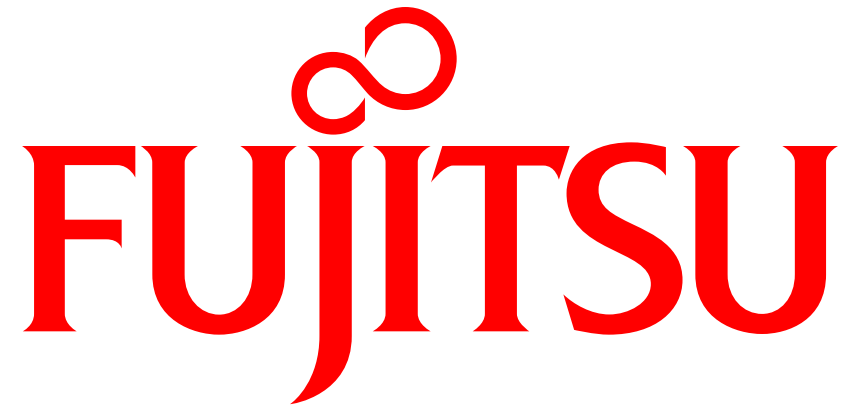
- (A) @media only screen and (max-width: 600px) {}
- (B) @media only screen and (min-width: 600px) {}
- (C) @media only screen or (max-width: 600px) {}
- (D) @media only print {}
- (E) @media only screen and (orientation:landscape) {}

orientationを使うと、デバイスの向きによって適用の可否を設定できるよ。or条件の指定はないからね！



LPI-JAPAN HTML5 Professional Certification

Open the Future with **HTML5**.



shaping tomorrow with you