

Talking to your browser

(doesn't make you crazy if your browser can respond)

Chen Hui Jing / @hj_chen





Surname

First name

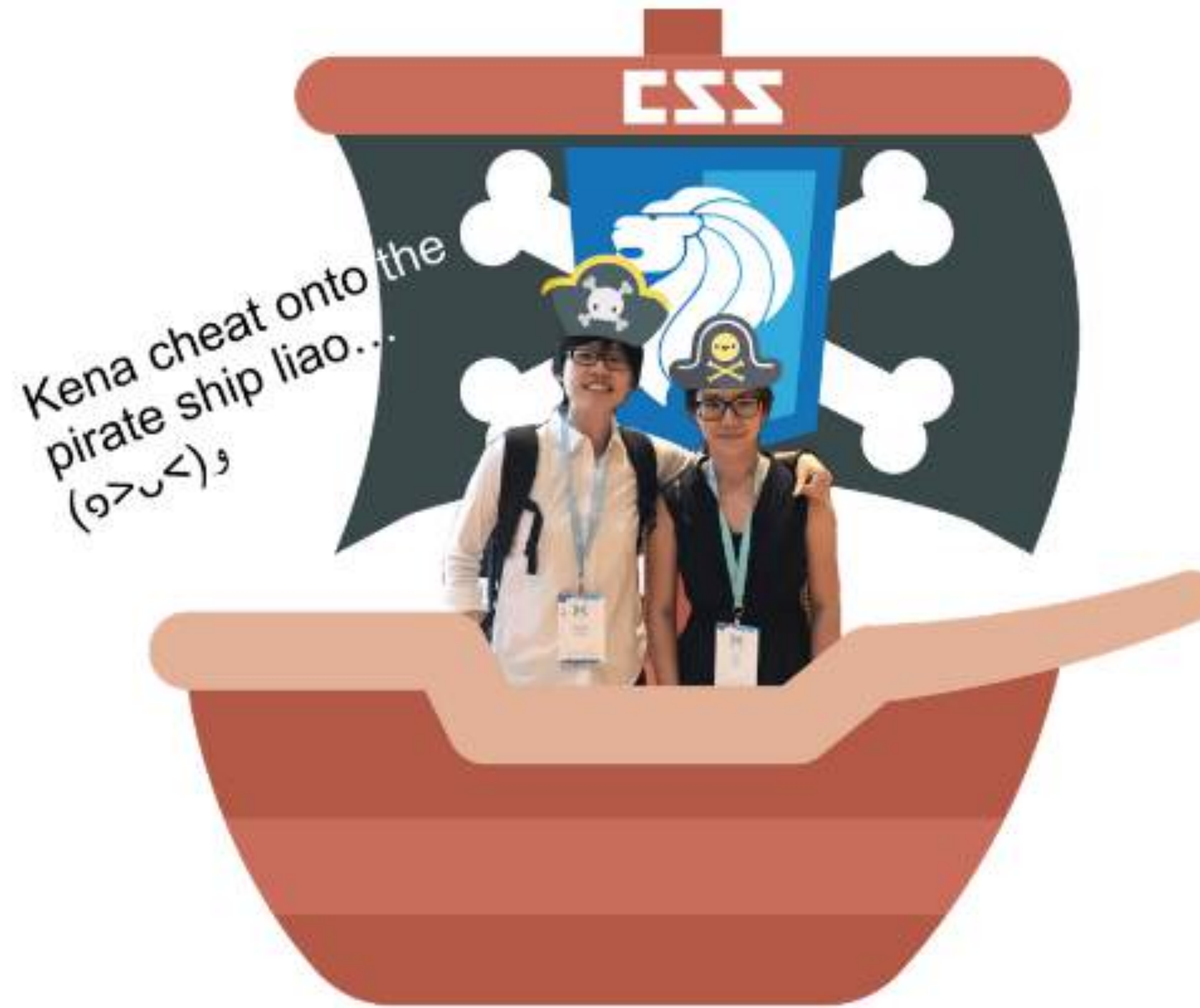
陈	慧	晶
Chen	Hui	Jing



@hj_chen



SingaporeCSS



What is CSS? 🤔

Nah... I'm just kidding. I'm going make a bold assumption that even if you don't write CSS, you at least heard of it before and know what it does.

Hopefully.



Evolution of CSS Specifications

CSS1

Recommendation:
17 Dec 1996

CSS2

Recommendation:
12 May 1998

CSS2.1

Recommendation:
7 Jun 2011

CSS2.2

Working draft:
12 Apr 2016

CSS3

Decision to modularise: 14 Apr 2000
(26 modules)

CSS Snapshot 2018 (99 modules)

Completed

CSS Snapshot 2018
CSS Snapshot 2017
CSS Snapshot 2015
CSS Snapshot 2010
CSS Snapshot 2007
CSS Color Level 3
CSS Namespaces
Selectors Level 3
CSS Level 2 Revision 1
Media Queries
CSS Style Attributes
CSS Fonts Level 3
CSS Basic User
Interface Level 3

Stable

CSS Backgrounds and Borders Level 3
CSS Conditional Rules Level 3
CSS Multi-column Layout Level 1
CSS Values and Units Level 3
CSS Cascading and Inheritance Level 3
CSS Fonts Level 3
CSS Writing Modes Level 3
CSS Counter Styles Level 3
CSS Containment Level 1

Rewriting

CSS Generated Content Level 3

Revising

CSS Paged Media Level 3
CSSOM View
CSS Intrinsic & Extrinsic Sizing Level 3
CSS Ruby Level 1
CSS Overflow Level 3
CSS Box Model Level 3
CSS Pseudo-Elements Level 4
CSS Scrollbars Level 1

Refining

CSS Animations Level 1
Web Animations
CSS Text Level 3
CSS Transforms Level 1
CSS Transitions
CSS Box Alignment Level 3
Selectors Level 4
CSS Lists Level 3
Motion Path Level 1
Preview of CSS Level 2
CSS Fonts Level 4
CSS Easing Functions Level 1
CSS Logical Properties and Values
Level 1

Testing

CSS Images Level 3
CSS Speech
CSS Text Decoration Level 3
CSS Shapes Level 1
CSS Masking Level 1
CSS Fragmentation Level 3
CSS Cascading Variables
Compositing and Blending Level 1
CSS Syntax Level 3
CSS Grid Layout Level 1
CSS Display Level 3
CSS Will Change Level 1
Media Queries Level 4
Geometry Interfaces Level 1
CSS Cascading and Inheritance Level 4
CSS Scroll Snap Level 1
CSS Painting API Level 1
CSS Writing Modes Level 4

Exploring

CSS Backgrounds and Borders Level 4
CSS Device Adaptation
CSS Exclusions
Filter Effects Level 1
CSS Generated Content for Paged Media
CSS Page Floats
CSS Template Layout
CSS Line Grid
CSS Positioned Layout Level 3
CSS Regions
CSS Table Level 3
CSS Object Model
CSS Font Loading
CSS Scoping Level 1
CSS Inline Layout Level 3
CSS Round Display Level 1
CSS Basic User Interface Level 4
CSS Text Level 4
CSS Properties and Values API Level 1

CSS Typed OM Level 1
Worklets Level 1
CSS Color Level 4
CSS Rhythmic Sizing Level 1
CSS Image Values and Replaced
Content Level 4
CSS Fill and Stroke Level 3
CSS Overflow Level 4
CSS Grid Layout Level 2
CSS Text Decoration Level 4
CSS Layout API Level 1
CSS Values and Units Level 4
CSS Shadow Parts
CSS Fragmentation Level 4
CSS Spatial Navigation Level 1
CSS Color Adjustment Level 1
CSS Overscroll Behavior Level 1
CSS Animation Worklet API
CSS Containment Level 2



CSS Snapshot 2018

(99 modules)

Completed

CSS Snapshot 2018
CSS Snapshot 2017
CSS Snapshot 2015
CSS Snapshot 2010
CSS Snapshot 2007
CSS Color Level 3
CSS Namespaces
Selectors Level 3
CSS Level 2 Revision 1
Media Queries
CSS Style Attributes
CSS Fonts Level 3
CSS Basic User Interface
Level 3

Revising

CSS Paged Media Level 3
CSSOM View
CSS Intrinsic & Extrinsic
Sizing Level 3
CSS Ruby Level 1
CSS Overflow Level 3
CSS Box Model Level 3
CSS Pseudo-Elements Level 4
CSS Scrollbars Level 1

Stable

CSS Backgrounds and Borders Level 3
CSS Conditional Rules Level 3
CSS Multi-column Layout Level 1
CSS Values and Units Level 3
CSS Cascading and Inheritance Level 3
CSS Fonts Level 3
CSS Writing Modes Level 3
CSS Counter Styles Level 3
CSS Containment Level 1

Rewriting

CSS Generated Content Level 3

Refining

CSS Animations Level 1
Web Animations
CSS Text Level 3
CSS Transforms Level 1
CSS Transitions
CSS Box Alignment Level 3
Selectors Level 4
CSS Lists Level 3
Motion Path Level 1
Preview of CSS Level 2
CSS Fonts Level 4
CSS Easing Functions Level 1
CSS Logical Properties and Values
Level 1

Exploring

CSS Backgrounds and Borders Level 4
CSS Device Adaptation
CSS Exclusions
Filter Effects Level 1
CSS Generated Content for Paged Media
CSS Page Floats
CSS Template Layout
CSS Line Grid
CSS Positioned Layout Level 3
CSS Regions
CSS Table Level 3
CSS Object Model
CSS Font Loading
CSS Scoping Level 1
CSS Inline Layout Level 3
CSS Round Display Level 1
CSS Basic User Interface Level 4
CSS Text Level 4
CSS Properties and Values API Level 1

Testing

CSS Images Level 3
CSS Speech
CSS Text Decoration Level 3
CSS Shapes Level 1
CSS Masking Level 1
CSS Fragmentation Level 3
CSS Cascading Variables
Compositing and Blending Level 1
CSS Syntax Level 3

CSS Typed OM Level 1

Worklets Level 1

CSS Color Level 4

CSS Rhythmic Sizing Level 1
CSS Image Values and Replaced
Content Level 4
CSS Fill and Stroke Level 3
CSS Overflow Level 4
CSS Grid Layout Level 2
CSS Text Decoration Level 4
CSS Layout API Level 1
CSS Values and Units Level 4
CSS Shadow Parts
CSS Fragmentation Level 4
CSS Spatial Navigation Level 1
CSS Color Adjustment Level 1
CSS Overscroll Behavior Level 1
CSS Animation Worklet API
CSS Containment Level 2

CSS Grid Layout Level 1
CSS Display Level 3
CSS Will Change Level 1
Media Queries Level 4
Geometry Interfaces Level 1
CSS Cascading and Inheritance Level 4
CSS Scroll Snap Level 1
CSS Painting API Level 1
CSS Writing Modes Level 4



TABLE OF CONTENTS

1	Introduction
2	Foreground Color: the 'color' property
3	Representing Colors: the <color> type
3.1	Accessibility and Conveying Information By Color
3.2	Color Spaces of Untagged Colors
3.3	Resolving <color> Values
4	sRGB Colors
4.1	The RGB functions: 'rgb()' and 'rgba()'
4.2	The RGB hexadecimal notations: '#RRGGBB'
5	Color Keywords
5.1	Named Colors
5.2	System Colors
5.3	The 'transparent' keyword
5.4	The 'currentcolor' keyword
6	HSL Colors: 'hsl()' and 'hsla()' functions
6.1	Converting HSL colors to sRGB colors
6.2	Examples of HSL colors
7	HWB Colors: 'hwb()' function
7.1	Converting HWB colors to sRGB colors
7.2	Examples of HWB Colors
8	Device-independent Colors: Lab and LCH

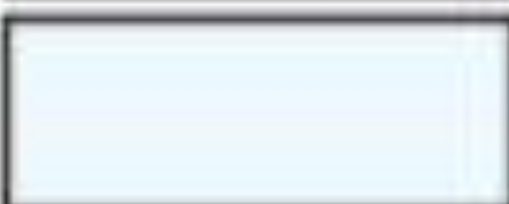
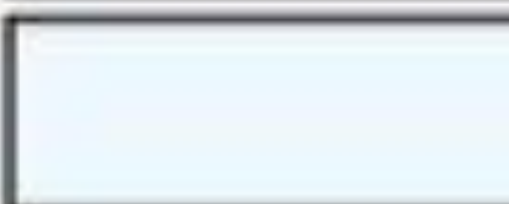












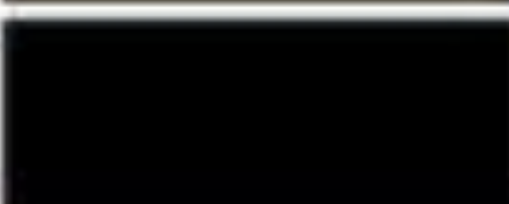
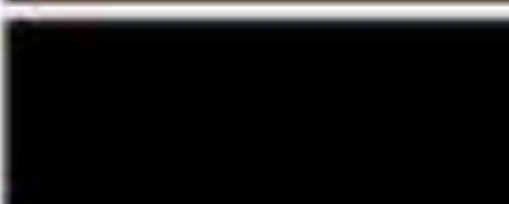






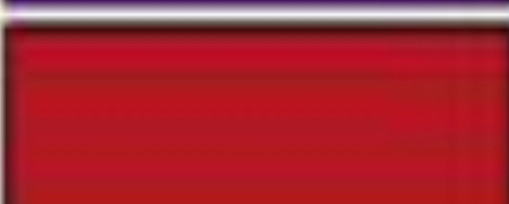
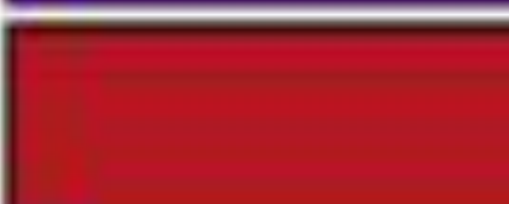






§ 5.1. Named Colors

CSS defines a large set of **named colors**, so that common colors can be written and read more easily. A **'<named-color>'** is written as an **<ident>**, accepted anywhere a **<color>** is. As usual for CSS-defined **<ident>**s, all of these keywords are case-insensitive.

The names resolve to colors in sRGB.

16 of CSS's named colors come from HTML originally: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white, and yellow. Most of the rest come from one version of the X11 color system, used in Unix-derived systems to specify colors for the console. (Two special color values, **'transparent'** and **'currentcolor'**, are specially defined in their own sections.)

The following table defines all of the opaque named colors, by giving equivalent numeric specifications in the other color syntaxes.

Named	Numeric	Color name	Hex rgb	Decimal
		'aliceblue'	#F0F8FF	240 248 255
		'antiquewhite'	#FAEBD7	250 235 215
		'aqua'	#00FFFF	0 255 255
		'aquamarine'	#7FFFD4	127 255 212
		'azure'	#F0FFFF	240 255 255
		'beige'	#F5F5DC	245 245 220
		'bisque'	#FFE4C4	255 228 196
		'black'	#000000	0 0 0
		'blanchedalmond'	#FFEBCD	255 235 205
		'blue'	#0000FF	0 0 255
		'blueviolet'	#8A2BE2	138 43 226
		'brown'	#A52A2A	165 42 42
		'burlywood'	#DEB887	222 184 135
		'cadetblue'	#5F9EA0	95 158 160
		'chartreuse'	#7FFF00	127 255 0

If you also like colours...



insert random caption here

Where did CSS named colours come from?



maroon #800000 rgba(128,0,0,1)	darkred #8B0000 rgba(139,0,0,1)	brown #A52A2A rgba(165,42,42,1)	firebrick #B22222 rgba(178,34,34,1)	rosybrown #BC8F8F rgba(188,143,143,1)	indianred #CD5C5C rgba(205,92,92,1)	lightcoral #F08080 rgba(240,128,128,1)	red #FF0000 rgba(255,0,0,1)	snow #FFFAFA rgba(255,250,250,1)	mistyrose #FFE4E1 rgba(255,228,225,1)	salmon #FA8072 rgba(250,128,114,1)	tomato #FF6347 rgba(255,99,71,1)	darksalmon #E9967A rgba(233,150,122,1)	coral #FF7F50 rgba(255,127,80,1)	orangered #FF4500 rgba(255,69,0,1)						
lightsalmon #FFA07A rgba(255,160,122,1)	sienna #A0522D rgba(160,82,45,1)	seashell #FFF5EE rgba(255,245,238,1)	saddlebrown #8B4513 rgba(139,69,19,1)	chocolate #D2691E rgba(210,105,30,1)	sandybrown #F4A460 rgba(244,164,96,1)	peachpuff #FFDAB9 rgba(255,218,185,1)	peru #CD853F rgba(205,133,63,1)	linen #FAF0E6 rgba(250,240,230,1)	bisque #FFE4C4 rgba(255,228,196,1)	darkorange #FF8C00 rgba(255,140,0,1)	burlywood #DEB887 rgba(222,184,135,1)	tan #D2B48C rgba(210,180,140,1)	antiquewhite #FAEBD7 rgba(250,235,215,1)	navajowhite #FFDEAD rgba(255,222,173,1)						
blanchedalmond #FFEBCD rgba(255,235,205,1)	papayawhip #FFEFD5 rgba(255,239,213,1)	moccasin #FFE4B5 rgba(255,228,181,1)	orange #FFA500 rgba(255,165,0,1)	wheat #F5DEB3 rgba(245,222,179,1)	oldlace #FDF5E6 rgba(253,245,230,1)	floralwhite #FFFAF0 rgba(255,250,240,1)	darkgoldenrod #8B800B rgba(139,128,11,1)	goldenrod #DAA520 rgba(218,165,32,1)	cornsilk #FFF8DC rgba(255,248,220,1)	gold #FFD700 rgba(255,215,0,1)	khaki #F0E68C rgba(240,230,140,1)	lemonchiffon #FFFFACD rgba(255,250,205,1)	palegoldenrod #EEE8AA rgba(238,232,170,1)	darkkhaki #8DB600 rgba(189,183,107,1)						
olive #808000 rgba(128,128,0,1)	beige #F5F5DC rgba(245,245,220,1)	lightgoldenrodyellow #FAFAD2 rgba(250,250,210,1)	yellow #FFFF00 rgba(255,255,0,1)	lightyellow #FFFFE0 rgba(255,255,224,1)	ivory #FFFFF0 rgba(255,255,240,1)	olivedrab #6B8E23 rgba(107,142,35,1)	yellowgreen #9ACD32 rgba(154,205,50,1)	darkolivegreen #556B2F rgba(85,107,47,1)	greenyellow #ADFF2F rgba(173,255,47,1)	chartreuse #7FFF00 rgba(127,255,0,1)	lawngreen #7CFC00 rgba(124,252,0,1)	darkseagreen #8FBC8F rgba(143,188,143,1)	darkgreen #006400 rgba(0,100,0,1)	green #008000 rgba(0,128,0,1)						
lime #00FF00 rgba(0,255,0,1)	forestgreen #228B22 rgba(34,139,34,1)	limegreen #32CD32 rgba(50,205,50,1)	lightgreen #90EE90 rgba(144,238,144,1)	<div><div></div><div>English!</div><div></div></div>										mediumaquamarine #66CDAA rgba(102,205,170,1)	aquamarine #7FFFD4 rgba(127,255,212,1)	turquoise #40E0D0 rgba(64,224,208,1)	lightseagreen #20B2AA rgba(32,178,170,1)			
mediumturquoise #48D1CC rgba(72,209,204,1)	teal #008080 rgba(0,128,128,1)	darkcyan #008B8B rgba(0,139,139,1)	aqua #00FFFF rgba(0,255,255,1)	lightcyan #ADD8E6 rgba(173,216,216,1)	skyblue #87CEEB rgba(135,206,235,1)	lightskyblue #87CEFA rgba(135,206,250,1)	steelblue #4682B4 rgba(70,130,180,1)	aliceblue #F0F8FF rgba(240,248,255,1)	dodgerblue #1E90FF rgba(30,144,255,1)	slategray #708090 rgba(112,128,144,1)	slategrey #708090 rgba(112,128,144,1)	lightslategray #778899 rgba(119,136,153,1)	lightslategrey #778899 rgba(119,136,153,1)	lightsteelblue #B0C4DE rgba(176,196,222,1)	cornflowerblue #6495ED rgba(100,149,237,1)	royalblue #4169E1 rgba(65,105,225,1)	navy #000080 rgba(0,0,128,1)	darkblue #00008B rgba(0,0,139,1)	mediumblue #0000CD rgba(0,0,205,1)	
darkblue #00008B rgba(0,0,139,1)	midnightblue #191970 rgba(25,25,112,1)	lavender #E6E6FA rgba(230,230,250,1)	ghostwhite #F8F8FF rgba(248,248,255,1)	slateblue #6A5ACD rgba(106,90,205,1)	darkslateblue #483D8B rgba(72,61,139,1)	mediumslateblue #7B68EE rgba(123,104,238,1)	mediumpurple #9370DB rgba(147,112,219,1)	rebeccapurple #800080 rgba(128,0,128,1)	blueviolet #8A2BE2 rgba(138,43,226,1)	indigo #4B0082 rgba(75,0,130,1)	darkorchid #9932CC rgba(153,50,204,1)	darkviolet #8A0080 rgba(138,0,128,1)	mediumorchid #BA55D3 rgba(186,85,211,1)	purple #800080 rgba(128,0,128,1)						
darkmagenta #8B008B rgba(139,0,139,1)	thistle #D8BFD8 rgba(216,191,216,1)	plum #DDA0DD rgba(221,160,221,1)	violet #EE82EE rgba(238,130,238,1)	fuchsia #FF00FF rgba(255,0,255,1)	magenta #FF00FF rgba(255,0,255,1)	orchid #DA70D6 rgba(218,112,214,1)	mediumvioletred #C71585 rgba(199,21,133,1)	deeppink #FF1493 rgba(255,20,147,1)	hotpink #FF69B4 rgba(255,105,180,1)	lavenderblush #FFF0F5 rgba(255,240,245,1)	palevioletred #DB7093 rgba(219,112,147,1)	crimson #DC143C rgba(220,20,60,1)	pink #FFC0CB rgba(255,192,203,1)	lightpink #FFB6C1 rgba(255,182,193,1)						
				black #000000 rgba(0,0,0,1)	dimgray #696969 rgba(105,105,105,1)	dimgrey #696969 rgba(105,105,105,1)	gray #808080 rgba(128,128,128,1)	grey #808080 rgba(128,128,128,1)	darkgray #A9A9A9 rgba(169,169,169,1)	darkgrey #A9A9A9 rgba(169,169,169,1)	silver #C0C0C0 rgba(192,192,192,1)	lightgray #D3D3D3 rgba(211,211,211,1)	lightgrey #D3D3D3 rgba(211,211,211,1)	gainsboro #DCDCDC rgba(220,220,220,1)	whitesmoke #F5F5F5 rgba(245,245,245,1)	white #FFFFFF rgba(255,255,255,1)				



Your browser can speak English

For the most part.



Cue bo liao idea...

Singlish lesson #1: Bo liao

Hokkien for “nothing better to do” . Dangerously idle.

In Mandarin, it's “无聊 (wú liáo)”

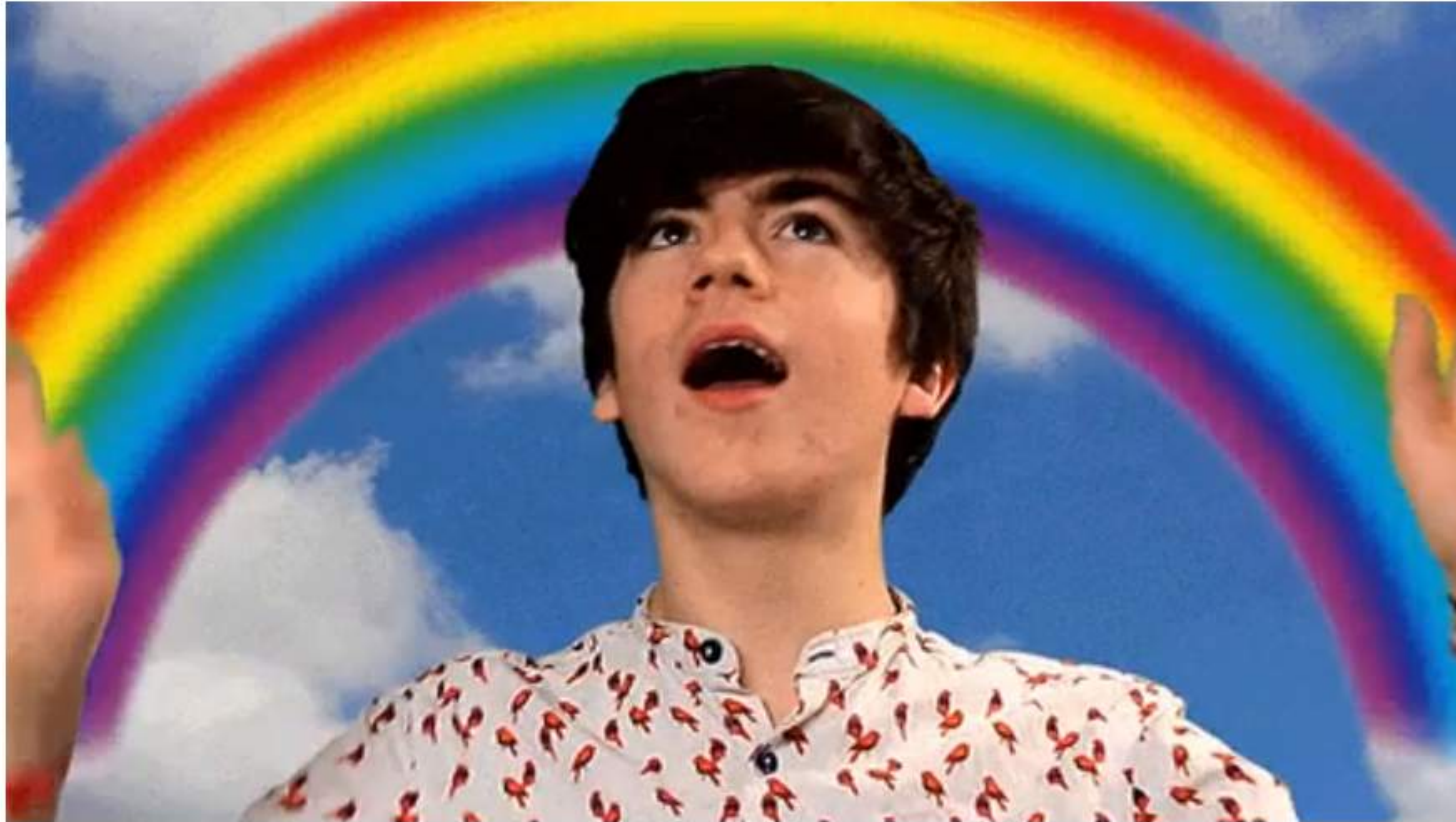
“What for he go and do that sort of thing? Must be damn bo liao.”

Source: [The Coxford Singlish Dictionary](#)



1.00

Change background colour with your voice!



Ah-maz-ing



Web Speech API

Speech Recognition

provides the ability to recognise voice context from an audio input and respond appropriately

- `SpeechRecognition`
- `SpeechGrammar`

Speech Synthesis

a text-to-speech component that allows programs to read out their text content

- `SpeechSynthesis`
- `SpeechSynthesisVoice`
- `SpeechSynthesisUtterance`

Make your browser listen to you 🦻

Bo liao also must plan

- 甲、Have web page
- 乙、Set background colour with CSS custom property
- 丙、Have button to trigger microphone
- 丁、Capture voice and process with SpeechRecognition
- 戊、Use result to update background colour



Have web page

```
<!doctype html>
<html class="no-js" lang="en">
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <title>Let's talk CSS colours</title>
    <meta name="description" content="Playing around with the WebSpeech API, CSS custom properties and CSS na
    <meta name="author" content="Chen Hui Jing">

    <link rel="stylesheet" href="styles.css">

    <!--[if lt IE 9]>
    <script src="http://html5shiv.googlecode.com/svn/trunk/html5.js"></script>
    <![endif]-->
  </head>

  <body>
    <main></main>
    <script src="scripts.js"></script>
  </body>
</html>
```



Set background colour with CSS custom property

```
:root {  
  --bg-colour: transparent;  
}  
  
main {  
  /* Of course got other styles la... */  
  /* You think magic meh... */  
  background-color: var(--bg-colour);  
}  
  
/* Moar styles not shown here */
```



What is a CSS custom property? 🤔

Defined in **CSS Custom Properties for Cascading Variables Module Level 1**

Introduces cascading variables as a new primitive value type that is accepted by all CSS properties, and custom properties for defining them

```
var( <custom-property-name> , <declaration-value>? )
```



Have button to trigger microphone

```
<body>
  <main>
    <h1>CSS Colours</h1>
    <p>How well do you know CSS named colours? Test both your knowledge as
well as your browser's ability to recognise your accent when you speak
English <span class="kaomoji">~\_(\_)\_/~</span></p>

    <button type="button" id="activateMic" class="btn-speak">Speak</button>

    <pre><code id="consoleLog">Click the button then say a colour...</code>
</pre>
  </main>

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



Capture voice and process with SpeechRecognition

But first...














Feature detection.



Browser support for SpeechRecognition 🙄

Speech Recognition API [🔗](#)

Method to provide speech input in a web browser.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	16	68 	76 	12	12.1			4.4	8.2 
10	17	69 	77 	12.1	12.4			4.4.4	9.2 
11	18	70 	78 	13	13.2	all	78 	76	10.1 
	76 	71 	79 	TP					

✓ ✗ Partial Support Prefixed  Behind a Flag

Global: 0% + 69.76% = 69.76%

Data from caniuse.com | Embed from caniuse.bitsofco.de

[Enable accessible colours](#)




```

((window, undefined) => {
  const document = window.document;
  const docElement = document.documentElement;

  const speechRecognition = window.webkitSpeechRecognition || window.mozSpeechRecognition || window.msSpeechRecognition;
  const speechGrammarList = window.webkitSpeechGrammarList || window.mozSpeechGrammarList || window.msSpeechGrammarList;

  function addClass(className) {
    docElement.className = `${docElement.className} ${className}`;
  }

  docElement.className = docElement.className.replace(/(^|\s)no-js(\s|$)/, '$1js$2');
  if (speechRecognition !== undefined) {
    addClass('speech');
  } else {
    addClass('no-speech');
  }
})(window);

```

Feature detection by Cătălin Mariș



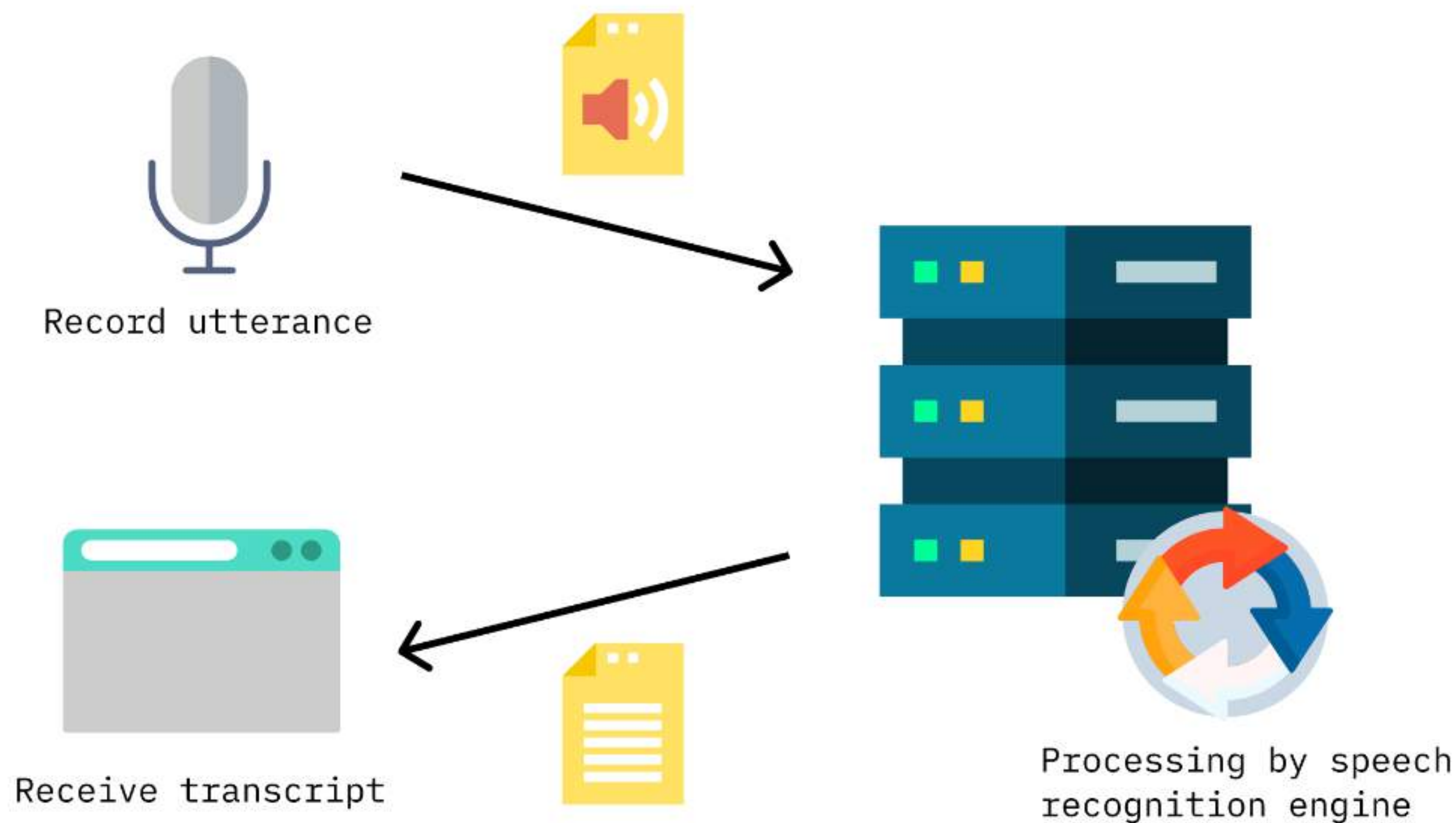
To activate in Firefox

Must be newer than 72.0a1 (2019-10-22)

- type `about:config` in your address bar
- search for the `media.webspeech.recognition.enable` and `media.webspeech.recognition.force_enable` preferences
- set them to `true`



How come browser can speak English?



Who's doing the processing?

Google.

Google **Cloud Speech-to-Text**, with speech recognition in 120 languages.

Mozilla is currently developing their own service called **Deep Speech**, hopefully can be validated in 2020 as a replacement for Google, at least in English.



Data and grammar stuff...

```
const colours = ['maroon', 'darkred', 'brown', 'firebrick',  
'rosybrown', 'indianred', 'lightcoral', 'red', 'snow',  
'mistyrose'.../* the rest of the 148 named CSS colours */];  
const grammar = '#JSGF V1.0; grammar colours; public <colour> = '  
+ colours.join(' | ') + ' ;';
```

JSpeech Grammar Format (JSGF)

#JSGF V1.0; states the format and version used. Must be included first.

grammar colours; public <colour> indicates the type of term we want recognised, followed by list of items separated by pipe character.




```
/* Define speech recognition instance */  
const recognition = new speechRecognition();  
/* Create new speech grammar list */  
const speechRecognitionList = new speechGrammarList();  
/* Add grammar to the list */  
speechRecognitionList.addFromString(grammar, 1);  
/* Add speech grammar list to speech recognition instance */  
recognition.grammars = speechRecognitionList;  
/* Set language of the recognition */  
recognition.lang = 'en-US';  
/* Can choose to return interim results or final results */  
recognition.interimResults = false;  
/* Set number of alternative potential matches */  
recognition.maxAlternatives = 1;
```




```
const micBtn = document.getElementById('activateMic')
const consoleLog = document.getElementById('consoleLog')

micBtn.addEventListener('click', function() {
  recognition.start(); /* Start speech recognition service */
  consoleLog.innerHTML = 'Ready to receive a colour command.'
}, false)
```




```
recognition.onresult = function(event) {  
  const last = event.results.length - 1;  
  const colour = event.results[last][0].transcript;  
  const sanitiseColour = colour.replace(/\s/g, '');  
  consoleLog.innerHTML = 'You probably said: ' + sanitiseColour +  
    docBody.style.setProperty('--bg-colour', sanitiseColour);  
}
```

Returns SpeechRecognitionResultList object with SpeechRecognitionResult objects, which can be accessed like an array

[last] returns the SpeechRecognitionResult at the last position




```
▼ SpeechRecognitionResultList(1)
  ▼ 0: SpeechRecognitionResult(1)
    ▼ 0: SpeechRecognitionAlternative
      confidence: 0.7521789073944092
      transcript: "green"
      ► <prototype>: SpeechRecognitionAlternativePrototype {
        transcript: Getter, confidence: Getter, ... }
      isFinal: true
      length: 1
      ► <prototype>: SpeechRecognitionResultPrototype { item:
        item(), length: Getter, isFinal: Getter, ... }
      length: 1
      ► <prototype>: SpeechRecognitionResultListPrototype { item:
        item(), length: Getter, ... }
```



Bonus CSS thing

Don't need an extra HTML element for the warning message

```
.no-speech body::before {  
  content: 'Tragically, your browser does not support the Speech I';  
  font-family: sans-serif;  
  line-height: 1.3;  
  font-size: 85%;  
  padding: 0.5em;  
  background-color: #ab3c3c;  
  color: white;  
  text-align: center;  
}
```



Make your browser talk back 🗨️

Browser support for SpeechSynthesis



Speech Synthesis API [🔗](#)

A web API for controlling a text-to-speech output.

IE	Edge	Firefox	Chrome	Safari	iOS Safari	Opera Mini	Chrome for Android	Android Browser	Samsung Internet
9	16	68	76	12	12.1			4.4	8.2
10	17	69	77	12.1	12.4			4.4.4	9.2
11	18	70	78	13	13.2	all	78	76	10.1
	76	71	79	TP					

✓ ✗ Partial Support

Global: 90.65% + 0% = 90.65%

Data from caniuse.com | Embed from caniuse.bitsofco.de

[Enable accessible colours](#)



Moar planning...

- 子、 Add select dropdown and play button
- 丑、 Populate select with device voice options
- 寅、 Change voice based on selected option
- 卯、 Play response when button clicked



Add requisite elements

```
<form id="hearResponse" class="response">  
  <select id="pickVoice"></select>  
  <button id="playResponse" class="btn-response">Hear response</button>  
</form>
```



Populate the select list

```
const select = document.getElementById('pickVoice');
voices = speechSynthesis.getVoices();
voices.forEach(function(voice) {
  const option = document.createElement('option');
  option.textContent = voice.name + ' (' + voice.lang + ')';
  if(voice.default) {
    option.textContent += ' -- DEFAULT';
  }
  option.setAttribute('data-lang', voice.lang);
  option.setAttribute('data-name', voice.name);
  select.appendChild(option);
});
```



Firefox voice list

```
▼ (48) [...]
  ▶ 0: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.Alex", name:
    "Alex", lang: "en-US", ... }
  ▶ 1: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.alice", name:
    "Alice", lang: "it-IT", ... }
  ▶ 2: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.alva", name:
    "Alva", lang: "sv-SE", ... }
  ▶ 3: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.amelie", name:
    "Amelie", lang: "fr-CA", ... }
  ▶ 4: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.anna", name:
    "Anna", lang: "de-DE", ... }
  ▶ 5: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.carmit", name:
    "Carmit", lang: "he-IL", ... }
  ▶ 6: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.damayanti",
    name: "Damayanti", lang: "id-ID", ... }
  ▶ 7: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.daniel", name:
    "Daniel", lang: "en-GB", ... }
  ▶ 8: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.diego", name:
    "Diego", lang: "es-AR", ... }
  ▶ 9: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.ellen", name:
    "Ellen", lang: "nl-BE", ... }
  ▶ 10: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.fiona", name:
    "Fiona", lang: "en-scotland", ... }
  ▶ 11: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.Fred", name:
    "Fred", lang: "en-US", ... }
  ▶ 12: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.ioana", name:
    "Ioana", lang: "ro-RO", ... }
  ▶ 13: SpeechSynthesisVoice { voiceURI: "urn:moz-tts:osx:com.apple.speech.synthesis.voice.joana", name:
    "Joana", lang: "pt-PT", ... }
```



Chrome voice list

▼ Array(67) ⓘ

- ▶ 0: SpeechSynthesisVoice {voiceURI: "Alex", name: "Alex", lang: "en-US", localService: true, default: true}
- ▶ 1: SpeechSynthesisVoice {voiceURI: "Alice", name: "Alice", lang: "it-IT", localService: true, default: false}
- ▶ 2: SpeechSynthesisVoice {voiceURI: "Alva", name: "Alva", lang: "sv-SE", localService: true, default: false}
- ▶ 3: SpeechSynthesisVoice {voiceURI: "Amelie", name: "Amelie", lang: "fr-CA", localService: true, default: false}
- ▶ 4: SpeechSynthesisVoice {voiceURI: "Anna", name: "Anna", lang: "de-DE", localService: true, default: false}
- ▶ 5: SpeechSynthesisVoice {voiceURI: "Carmit", name: "Carmit", lang: "he-IL", localService: true, default: false}
- ▶ 6: SpeechSynthesisVoice {voiceURI: "Damayanti", name: "Damayanti", lang: "id-ID", localService: true, default: false}
- ▶ 7: SpeechSynthesisVoice {voiceURI: "Daniel", name: "Daniel", lang: "en-GB", localService: true, default: false}
- ▶ 8: SpeechSynthesisVoice {voiceURI: "Diego", name: "Diego", lang: "es-AR", localService: true, default: false}
- ▶ 9: SpeechSynthesisVoice {voiceURI: "Ellen", name: "Ellen", lang: "nl-BE", localService: true, default: false}
- ▶ 10: SpeechSynthesisVoice {voiceURI: "Fiona", name: "Fiona", lang: "en", localService: true, default: false}
- ▶ 11: SpeechSynthesisVoice {voiceURI: "Fred", name: "Fred", lang: "en-US", localService: true, default: false}
- ▶ 12: SpeechSynthesisVoice {voiceURI: "Ioana", name: "Ioana", lang: "ro-RO", localService: true, default: false}
- ▶ 13: SpeechSynthesisVoice {voiceURI: "Joana", name: "Joana", lang: "pt-PT", localService: true, default: false}
- ▶ 14: SpeechSynthesisVoice {voiceURI: "Jorge", name: "Jorge", lang: "es-ES", localService: true, default: false}
- ▶ 15: SpeechSynthesisVoice {voiceURI: "Juan", name: "Juan", lang: "es-MX", localService: true, default: false}
- ▶ 16: SpeechSynthesisVoice {voiceURI: "Kanya", name: "Kanya", lang: "th-TH", localService: true, default: false}
- ▶ 17: SpeechSynthesisVoice {voiceURI: "Karen", name: "Karen", lang: "en-AU", localService: true, default: false}
- ▶ 18: SpeechSynthesisVoice {voiceURI: "Kyoko", name: "Kyoko", lang: "ja-JP", localService: true, default: false}
- ▶ 19: SpeechSynthesisVoice {voiceURI: "Laura", name: "Laura", lang: "sk-SK", localService: true, default: false}
- ▶ 20: SpeechSynthesisVoice {voiceURI: "Lekha", name: "Lekha", lang: "hi-IN", localService: true, default: false}




```
const responseForm = document.getElementById('hearResponse')
responseForm.addEventListener('submit', function(event) {
  event.preventDefault();
  const select = document.getElementById('pickVoice');
  speechSynthesis.cancel(); /* Needed to clear the previous result */
  /* create a new SpeechSynthesisUtterance() instance */
  const utterStuff = new SpeechSynthesisUtterance(result);
  const selectedVoice = select.selectedOptions[0].getAttribute('data-voice');
  voices.forEach(function(voice) {
    if(voice.name === selectedVoice) {
      utterStuff.voice = voice;
    }
  });
  speechSynthesis.speak(utterStuff); /* Start the utterance being
}, false)
```



Do the live demo thingy



Links and stuff

- [Web Speech API \(Draft Community Group Report\)](#)
- [Using the Web Speech API](#)
- [Web Speech API - Speech Recognition](#)
- [Web Speech API Demonstration](#) by Google
- [Let's talk CSS Colours](#) (the bo liao app)
- [Source code for the bo liao app](#)

Thank you!



<https://www.chenhuijing.com>



@hj_chen



@hj_chen



@huijing

Header font is **Vera Cruz BT** by **Ray Cruz**
Body font is **Morandi** by **Jovica Veljović**

