CSS for internationalisation

By Chen Hui Jing/@huijing@tech.lgbt

Numeronyms

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 Internationalisation

1 2 3 4 5 6 7 8 9 10 Ocalisation

Internationalisation is the design and development of a product, application or document that enables easy localisation for target audiences that vary in culture, region, or language.

What is CSS?

A vocabulary for communicating with the browser.

To inform the browser how we want elements to be styled and laid out.

CSS is a domain-specific, declarative programming language.

-Lara Schenck



Why lang matters?

- Language-specific behaviour in browsers
- Default font selection
- Search engine optimisation
- Spelling and grammar checks
- Translation
- Non-text readers
- Parsers and scripts

Source: Why use the language attribute?

Basics of using lang

Set the default language of the document:

<html lang="zh">

If you have mixed language content on your page:

The fourth animal in the Chinese Zodiac is Rabbit (兔子)

The:lang() pseudo-class

A selector that represents an element based on its language

We use *italics* to emphasise words in English, 但是中文 ···· 则是用着重号.

```
We use <em>italics</em> to emphasise words in English, <span lang="zh">但是中文则是用<em>着重号</em></span>
```

```
.lang-pc em:lang(zh) {
  font-style: normal;
  text-emphasis: dot;
}
```

Attribute selectors

A selector that matches elements based on the presence or value of a given selector

```
[lang="zh"]
/* will match only zh */
```

```
[lang^="zh"]
/* will match zh, zh-HK, zh-Hans, zhong, zh123...
 * basically anything with zh as the first 2 characters */
```

```
[lang|="zh"]
/* will match zh, zh-HK, zh-Hans, zh-amazing, zh-123 */
```

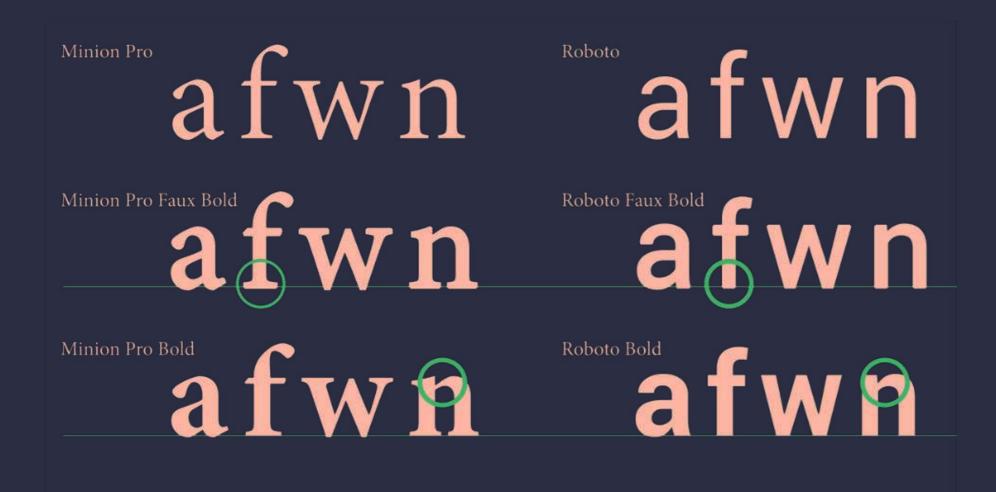
Some things to note...

The : lang() pseudo-class does take care of partial tag matching

In terms of selector matching, the |= operator performs a comparison against a given **attribute** on the element, while the :lang() pseudo-class uses the UAs knowledge of the document's **semantics** for the comparison

You can always use normal **classes** or **ids**, but you would no longer be making use of the convenience of what already exists

The faux fonts problem



Spotting faux bold

afeunyi

Afeunyi

Afeunyi

Minion Pro Faux Italic

Afeunyi

Minion Pro Italic

Afeunyi

Spotting faux italic

Image source: Cai

font-synthesis

Controls if the browser should synthesize a missing typeface or not

We use *italics* to emphasise words in English, 但是中文则是用着重号.

```
faux em:lang(zh) {
  font-synthesis: none;
}
```

Glyph differences



lang="zh-hans"



lang="zh-hant"

font-variant-east-asian

Allows control of glyph substitution and sizing in East Asian text

一個簡體字可能對應多個繁體字,如簡體字「發」,其相應的繁體字可能為「發」或「髮」;一個繁體漢字對應多個簡體漢字的情況與前者相比數量極少但仍需註意,如繁體字「乾」可能對應簡體字「幹」或「乾」。

```
.fvea :lang(zh) {
  font-variant-east-asian:
traditional;
}
```

font-language-override

To control the use of language-specific glyph substitutions and positioning

```
<!-- Macedonian lang code -->
<body lang="mk">
        <h4>Член 9</h4>
        Никој човек нема да биде подложен на произволно апсење, притвор или прогонув</body>
```

```
body {
  /* Serbian OpenType language tag */
  font-language-override: "SRB";
}
```

Example lifted from CSS Fonts Module Level 4

text-transform

Transforms text for styling purposes

If I want [flowers], I'm going to send them to myself.

SÜSSE SOSSEN-KLÖSSE GENIESSEN MASSGEBLICH GEFRÄSSIGE PREUSSISCHE NUTZNIESSER.

ΟΥΔΈΝ ΚΑΚΌΝ ΑΜΙΓΈΣ ΚΑΛΟΎ.

アイウエオカクケ

```
.txt-trans p:lang(en) {
  text-transform: normal;
}

.txt-trans p:lang(de),
.txt-trans p:lang(gr) {
  text-transform: uppercase;
}

.txt-trans p:lang(jp) {
  text-transform: full-size-kana;
}
```

Line breaks for inline boxes

If an element generates

zero boxes, was it really

there at all?

If an element generates zero boxes

```
.linebox p.line-container {
  font-size: 200%;
}
.linebox .line-container em {
  background-color: mediumpurple;
  padding: 0.25em;
}
.linebox .line-container strong {
  background-color: cornflowerblue;
  padding: 0.25em;
  mix-blend-mode: color-burn;
}
```

CSS for controlling line breaks

line-break	allows choosing various levels of "strictness" for line breaking restrictions
word-break	controls what types of letters are glommed together to form unbreakable "words", causing CJK characters to behave like non-CJK text or vice versa
hyphens	controls whether automatic hyphenation is allowed to break words in scripts that hyphenate
overflow-wrap	allows the UA to take a break anywhere in otherwise-unbreakable strings that would otherwise overflow

Line breaking by Florian Rivoal @ dotCSS

Hyphenation

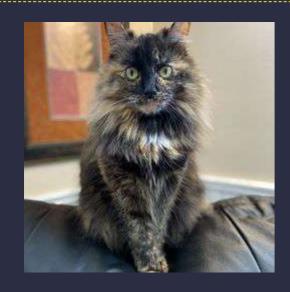
If you don't give a lang attribute, you don't get automatic hyphenation.

-Florian Rivoal

Browsers use language-specific dictionaries to figure out where the hyphenation points should be.

The hyphens property

The manual route is to use ­ but are you really going to manually do that?



konservatorio konservatorio



põllumajandusministeeriumvervoerdersaansprakelijkheidsverzekering

```
result hyphen-lang {
  hyphens: auto;
}
```

writing-mode

שלים שלים שלי השיליות השליל שמצם שילי אבית שלים שליש שליש בליים ניצייוניתית שלים נעשת המליט בליים ניצייוניתית שלים איזיית שילים אל שלבית איזיית המני שיליבית

Ort Jorkang Jurgitta

writing-mode: vertical-lr

すべての人間は、生まれなおりにして自由であり、 万町しとを授けられており、 互いにである。人間は、理性と良いである。人間は、理性と良いなければならない。 生まれなみ

writing-mode: vertical-rl

text-orientation & text-combine-upright

National Basketball
National Basketball
Association ,縮寫:

是北美的男子職業籃球腦Association,縮寫:點Association,縮寫:點國家籃球協會(英語:

text-orientation: upright

text-combine-upright: all

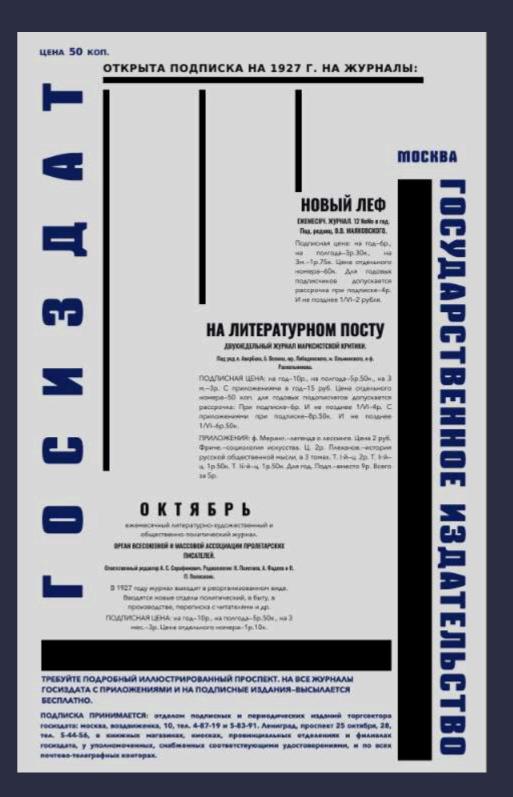
Right-to-left languages

CSS should **not** be used for bi-directional styling

Because directionality is an integral part of the document structure, markup should be used to set the directionality for a document or chunk of information, or to identify places in the text where the Unicode bidirectional algorithm alone is insufficient to achieve desired directionality.

Not just for East Asian text





Confusing physical directions

Physical directions for writing modes other than horizontal top-to-bottom may be confusing

```
Госиздат
```

```
.vert .title {
 writing-mode: vertical-rl;
 margin-inline-start: 1em;
 margin-block-end: 1em;
```

Logical properties

writing-mode / direction

		horizontal-tb		vertical-rl		vertical-lr	
		ltr	rtl	ltr	rtl	ltr	rtl
Edge	top	inset-block- start	inset-block- start	inset-inline- start	inset-inline- end	inset-inline- start	inset-inline- end
	right	inset-inline- end	inset-inline- start	inset-block- start	inset-block- start	inset-block- end	inset-block- end
	bottom	inset-block- end	inset-block- end	inset-inline- end	inset-inline- start	inset-inline- end	inset-inline- start
	left	inset-inline- start	inset-inline- end	inset-block- end	inset-block- end	inset-block- start	inset-block- start

Result

A comparison of physical and logical directions for borders

Given the requirement is to have a box with a run of text within it with the following characteristics:

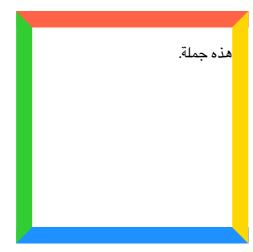
- 1. The border colour at the top edge **of the run of text** should be red.
- 2. The border colour at the right edge of the run of text should be green.
- 3. The border colour at the bottom edge of the run of text should be blue.
- 4. The border colour at the left edge of the run of text should be yellow.

Using physical directions requires a modification every time the writing direction changes, whereas using logical properties allows the same properties and values for all six use cases.

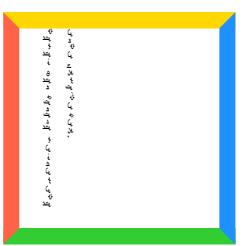
Physical directions

This is a sentence.

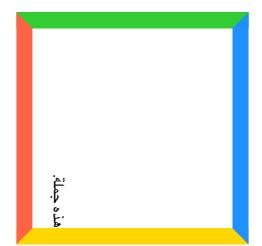
border-top-color: tomato; border-right-color: limegreen; border-bottom-color: dodgerblue; border-left-color: gold;



border-top-color: tomato;
border-left-color: limegreen;
border-bottom-color: dodgerblue;
border-right-color: gold;



border-left-color: tomato; border-bottom-color: limegreen; border-right-color: dodgerblue; border-top-color: gold;



border-left-color: tomato; border-top-color: limegreen; border-right-color: dodgerblue; border-bottom-color: gold;



border-right-color: tomato;
border-bottom-color: limegreen;
border-left-color: dodgerblue;
border-top-color: gold;



border-right-color: tomato;
border-top-color: limegreen;
border-left-color: dodgerblue;
border-bottom-color: gold;

Resources

1× 0.5× 0.25×

Keri

Lists and counters

Allows us to display ordered lists with international numbering systems

```
甲、
                                         .counters ol {
乙、
                                           list-style-type: cjk-heavenly-
丙、
                                         stem;
                                           text-align: center;
                                           font-size: 85%;
戊、
庚、
辛、
壬、
```

Full list of list-style-type options at MDN

CSS Fizzbuzz

```
1.
2.
Fizz
4.
Buzz
Fizz
7.
8.
Fizz
Buzz
11.
```

```
.fizzbuzz ol li:nth-of-
type(3n+3)::marker {
   content: "Fizz"
}
.fizzbuzz ol li:nth-of-
type(5n+5)::marker {
   content: "Buzz"
}
.fizzbuzz ol li:nth-of-
type(3n+3):nth-of-type(5n+5)::marker
{
   content: "FizzBuzz"
}
```

Further reading and references

- CSS for internationalisation
- CSS is a Declarative, Domain-Specific Programming Language
- Why use the language attribute?
- Using the HTML lang attribute
- On Use of the Lang Attribute
- Declaring language in HTML
- Robust Vertical Text Layout
- Unicode Bidirectional Algorithm































Thank you

- https://chenhuijing.com
- () @huijing
- @huijing@tech.lgbt
- @huijing

Font is Redaction by Titus Kaphar and Reginald Dwayne Betts.

https://huijing.github.io/slides/108-dsw-2024 @huijing@tech.lgbt