MAKING CSS FROM GOOD TO GREAT
THE SINGLISH EDITION

Hui Jing / @hj_chen
SINGLISH-O-METER

ANG MOH CATCH NO BALL

QUEEN’S ENGLISH
Singlish lesson: Tok Kong

Hokkien term meaning “superb” or “top of the line”. In Mandarin, it's possible to write it as “督公 (dū gōng)”

“Have you used CSS grid to build layout? It's super tok kong!”

Source: The Oxford Singlish Dictionary
WEB LAYOUTS OVER THE YEARS

Singlish lesson: Swee
Hokkien for “pretty” or “beautiful”. Also used to describe something as properly or correctly done. “This layout do until very swee leh.”

Source: The Coxford Singlish Dictionary
WEB LAYOUTS OVER THE YEARS
WEB LAYOUTS OVER THE YEARS

Singlish lesson: Jia lat
Hokkien adjective literally meaning "to eat strength". Means that a task is onerous and consumes one's energy. Can sometimes be used to denote being in trouble. “Have you inspected Twitter lately? The markup nest until very jia lat leh.”
Source: The Coxford Singlish Dictionary
WEB LAYOUTS OVER THE YEARS
WEB LAYOUTS OVER THE YEARS

Singlish lesson: Senget
Malay for “slanted” or “crooked”.
“How come the last item in the row senget wan?”

Source: The Coxford Singlish Dictionary
WEB LAYOUTS OVER THE YEARS
which one of the two possible websites are you currently designing?
**BLOCK LAYOUT**
designed for laying out documents

**INLINE LAYOUT**
designed for laying out text

**TABLE LAYOUT**
designed for laying out 2D data in a tabular format

**POSITIONED LAYOUT**
designed for very explicit positioning without much regard for other elements in the document
FLEXIBLE BOX LAYOUT describes a CSS box model optimised for user interface design and designed for laying out more complex applications and webpages.
GRID LAYOUT

Defines a two-dimensional grid-based layout system, optimised for user interface design.
WHY IS CSS GRID SPECIAL?

“Grid works from the container in, other layout methods start with the item”
—Rachel Andrew
# LAYOUT TECHNIQUE: inline-block

<table>
<thead>
<tr>
<th>Item A</th>
<th>Item B</th>
<th>Item C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item D</td>
<td>Item E</td>
<td>Item F</td>
</tr>
</tbody>
</table>

```
.inlineblock__item {
  display: inline-block;
  width: calc(100% / 3);
}
```
**LAYOUT TECHNIQUE: float**

<table>
<thead>
<tr>
<th>Item A</th>
<th>Item B</th>
<th>Item C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item D</td>
<td>Item E</td>
<td>Item F</td>
</tr>
</tbody>
</table>

```css
.float__item {
    float: left;
    width: calc(100% / 3);
}
```
## Layout Technique: flex

<table>
<thead>
<tr>
<th>Item A</th>
<th>Item B</th>
<th>Item C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item D</td>
<td>Item E</td>
<td>Item F</td>
</tr>
</tbody>
</table>

```css
.flexbox {
    display: flex;
    flex-wrap: wrap;
}
.flexbox__item {
    flex: 0 0 calc(100% / 3);
}
```
Grid is the only layout technique that establishes a relationship between rows and columns of grid items.
"Grid is the relational database's way of handling a large number of items."

"...fuses a relational database with a grid..."
BASICS OF CSS GRID

Define your grid.

Place items in the grid.
GRID TERMINOLOGY

- Grid line
- Grid track
- Grid gap
- Grid area
- Grid cell
<table>
<thead>
<tr>
<th>Sun</th>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**March 2017**

<table>
<thead>
<tr>
<th>Sun</th>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
<th>Sat</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>20</td>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>28</td>
<td>29</td>
<td>30</td>
<td>31</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**October 2017**
Singlish lesson: Si bei

Originated from Teochew, means “very” or “extremely”
Literally translates to “死父 (sī fù)” or “dead father”
“The CSS grid roll-out in 2017 was sibei successful.”
# Browser Support for Grid (Level 1)

CSS Grid Layout (level 1) - Method of using a grid concept to lay out content, providing a mechanism for authors to divide available space for layout into columns and rows using a set of predictable sizing behaviors. Includes support for all "grid-*" properties and the "fr" unit.

<table>
<thead>
<tr>
<th></th>
<th>IE 9</th>
<th>Edge</th>
<th>Firefox</th>
<th>Chrome</th>
<th>Safari</th>
<th>iOS Safari</th>
<th>Opera Mini</th>
<th>Chrome for Android</th>
<th>Android Browser</th>
<th>Samsung Internet</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>18</td>
<td>70</td>
<td>78</td>
<td>13</td>
<td>13.2</td>
<td>all</td>
<td>79</td>
<td>76</td>
<td>76</td>
<td>16.1</td>
</tr>
<tr>
<td>10</td>
<td>17</td>
<td>69</td>
<td>77</td>
<td>12.1</td>
<td>13.1</td>
<td></td>
<td>69</td>
<td></td>
<td>4.4</td>
<td>5.2</td>
</tr>
<tr>
<td>9</td>
<td>16</td>
<td>68</td>
<td>76</td>
<td>12</td>
<td>12.4</td>
<td></td>
<td>76</td>
<td></td>
<td>4.4</td>
<td>4.4</td>
</tr>
</tbody>
</table>

Data from caniuse.com | Embedded from caniuse.com
USING DEVTOOLS TO LEARN GRID

Live demo
WHAT'S IN GRID LEVEL 2?

Level 2 expands Grid by:

- adding “subgrid” capabilities for nested grids to participate in the sizing of their parent grids
- aspect-ratio–controlled gutters
WHY DO WE NEED SUBGRID?
THE SUBGRID SYNTAX

Same properties, more values

```css
.grid-template-columns: subgrid <line-name-list>?
.grid-template-rows: subgrid <line-name-list>?
```

The `subgrid` keyword indicates to the browser the nested grid will use the same sizing as its parent along the relevant axis.

```
.subgrid-container {
  grid-columns: 2 / 5; /* placement for the subgrid container itself */
  display: grid; /* you must still apply a display: grid to the subgrid */
  grid-template-columns: subgrid;
  grid-template-rows: subgrid;
}
```
USING DEVTOOLS TO LEARN SUBGRID

Live demo
BROWSER SUPPORT FOR SUBGRID

Feature of the CSS Grid Layout Module Level 2 that allows a grid-item with its own grid to align in one or both dimensions with its parent grid.

<table>
<thead>
<tr>
<th></th>
<th>IE</th>
<th>Edge</th>
<th>Firefox</th>
<th>Chrome</th>
<th>Safari</th>
<th>iOS Safari</th>
<th>Opera Mini</th>
<th>Chrome for Android</th>
<th>Android Browser</th>
<th>Samsung Internet</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>9</td>
<td>16</td>
<td>68</td>
<td>76</td>
<td>12</td>
<td>12.4</td>
<td></td>
<td></td>
<td>4.4</td>
<td>8.2</td>
</tr>
<tr>
<td>10</td>
<td>10</td>
<td>17</td>
<td>69</td>
<td>77</td>
<td>12.1</td>
<td>13.1</td>
<td></td>
<td></td>
<td>4.4,3</td>
<td>9.2</td>
</tr>
<tr>
<td>11</td>
<td>11</td>
<td>18</td>
<td>70</td>
<td>78</td>
<td>13</td>
<td>13.2</td>
<td>all</td>
<td></td>
<td>78</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>16.1</td>
</tr>
</tbody>
</table>

76 | 71 | 79 | TP | 13.3 |
REFERENCES

- CSS Flexible Box Layout Module Level 1
- CSS Grid Layout Module Level 1
- CSS Grid Layout Module Level 2
- Codrops reference: Subgrid
- Subgrid on MDN
- Grid Level 2 and Subgrid
- Why display: contents is not CSS Grid Layout subgrid
- CSS Grid Level 2: Here Comes Subgrid
- Grid by Example
- Researching a Property in the CSS Specifications
THANKS SIA 🤗

https://www.chenhuijing.com

@hj_chen

@huijing

Header font is Ostrich Sans by Tyler Finck.
Body font is Libre Baskerville by Pablo Impallari.