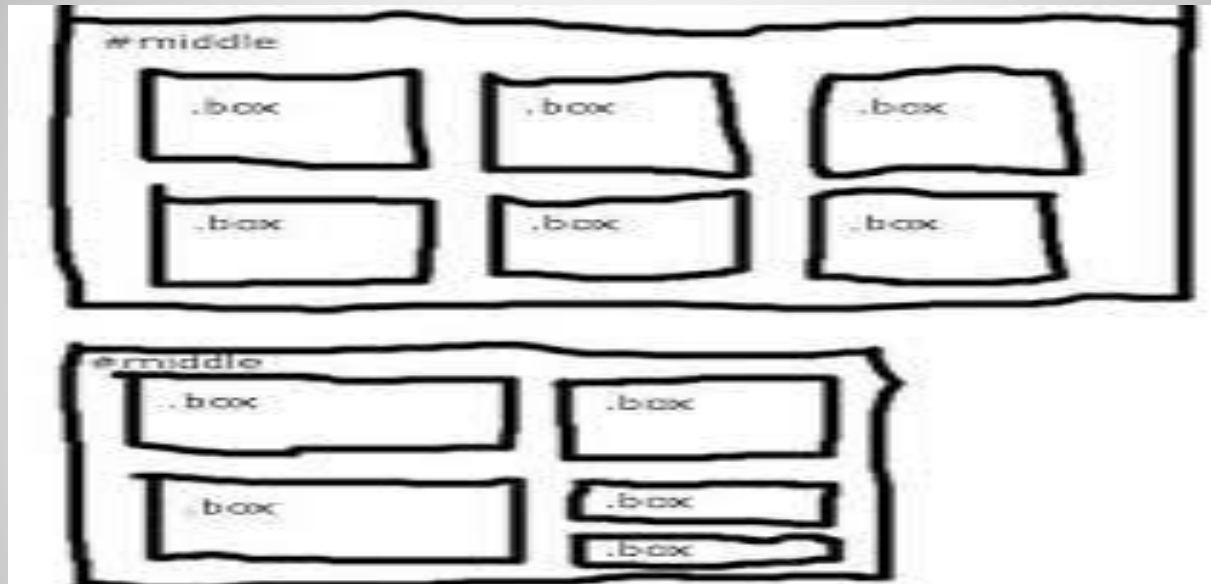


Web Development Fundamentals



Flex Nav REcap
Lecturer: Marie Brennan

The 4 rules for effective use of: Unordered lists for navbars



- The 4 steps for simple (vertical) navigation bars
 1. `` remove bullets and margins and padding
 2. `<a>` display as blocks
 3. `<a>` define `:hover` style
 4. `<a>` define `.current_page` style+ (add some pretty CSS colors/padding/border etc.)
- Two new steps:
 - Make list block `` a flex box container
 - Make the list items `` share space flexibly



RECAP – Steps 1 – 4 for simple nav bars

Nav lists ...



- Features of unordered lists:

- Indentation of whole list
- Bullet points

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>IT Blanchardstown</title>
</head>

<body>

  <nav>
    <ul>
      <li><a href="index.html">home</a></li>
      <li><a href="about.html">about</a></li>
      <li><a href="products.html">products</a></li>
      <li><a href="contact.html">contact</a></li>
    </ul>
  </nav>

</body>
</html>
```

- [home](#)
- [about](#)
- [products](#)
- [contact](#)

- Use CSS to remove spacing and bullet points:

- Note – descendent selector

- So only inside <nav> affected ...

```
nav ul {
  margin: 0;
  padding: 0;
  list-style-type: none;
}
```

[home](#)
[about](#)
[products](#)
[contact](#)

step 1

remove bullets/spacing from list

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>IT Blanchardstown</title>
</head>

<body>

<nav>
  <ul>
    <li><a href="index.html">home</a></li>
    <li><a href="about.html">about</a></li>
    <li><a href="products.html">products</a></li>
    <li><a href="contact.html">contact</a></li>
  </ul>
</nav>

</body>
</html>
```

```
nav ul {
  margin: 0;
  padding: 0;
  list-style-type: none;
}
```

[home](#)
[about](#)
[products](#)
[contact](#)

step 2

display <a> as block



step 1

```
nav ul {  
    margin: 0;  
    padding: 0;  
    list-style-type: none;  
}
```

```
nav ul li a {  
    display: block;  
    background-color: yellow;  
    padding: 1em;  
    border: 1px solid black;  
}
```

step 2

[home](#)

[about](#)

[products](#)

[contact](#)

step 3

style a: hover for link when mouse over

step 1



step 2

```
nav ul {  
    margin: 0;  
    padding: 0;  
    list-style-type: none;  
}
```

```
nav ul li a {  
    display: block;  
    background-color: yellow;  
    padding: 1em;  
    border: 1px solid black;  
}
```

```
/* step 3 - :hover style for links */
```

```
nav ul li a: hover {  
    background-color: black;  
    color: yellow;  
    text-decoration: none;  
}
```

home



[about](#)

[products](#)

[contact](#)

step 4

style current page differently

```
nav ul {  
    margin: 0;  
    padding: 0;  
    list-style-type: none;  
}
```

step 1

```
nav ul li a {  
    display: block;  
    background-color: yellow;  
    padding: 1em;  
    border: 1px solid black;  
}
```

step 2

```
/* step 3 - :hover style for links */
```

```
nav ul li a:hover {  
    background-color: black;  
    color: yellow;  
    text-decoration: none;  
}
```

step 3

```
<li>  
  <a href="about"  
    class="current_page">about</a>  
</li>
```

step 4

```
nav ul li a.current_page {  
    background-color: pink;  
}
```

step 4

[home](#)

[about](#)



Make `` block a FLEX container



- At the core of horizontal navbars is a list that is a flex box:

```
nav ul {  
    display: flex;  
}
```

- Note – at present the ITB Chrome version is a little old, so you also need to write the `–webkit-` version of the CSS property..try it without the kit!

```
nav ul {  
    display: flex;  
    display: -webkit-flex;  
}
```



**Make `` items expand
(flex) to use up extra space**



- If all list items have flex ratio of '1', they will all take an equal share of remaining horizontal space
- So the list items will expand to 'spread' out

```
nav ul li {  
    flex: 1;  
}
```

/ can shorten, if only ever one list in your
'nav' blocks */*

```
nav li {  
    flex: 1;  
}
```



- For older versions of chrome you need the webkits...

```
nav ul li {  
    flex: 1;  
    -webkit-flex: 1;  
}
```

a basic horizontal navigation bar
using a FLEX 'ul' containing block (so 'li' items can stack up horizontally on the same line) and each 'li' element stretches to use up spare spacing

Home

Contact Us

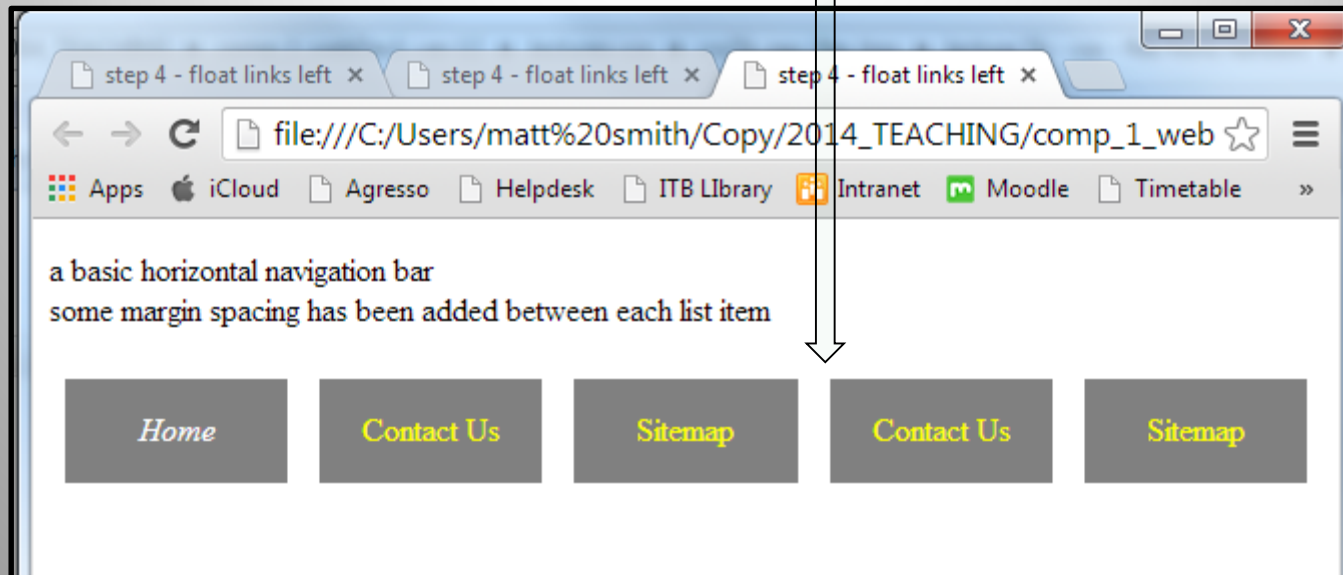
Sitemap

Contact Us

Sitemap

- Above – basic horizontal navbar
- Below – some margin spacing added between each item

```
nav li {  
  margin: 0.5em;  
}
```





Summary and Conclusions

steps horiz-1 & 2

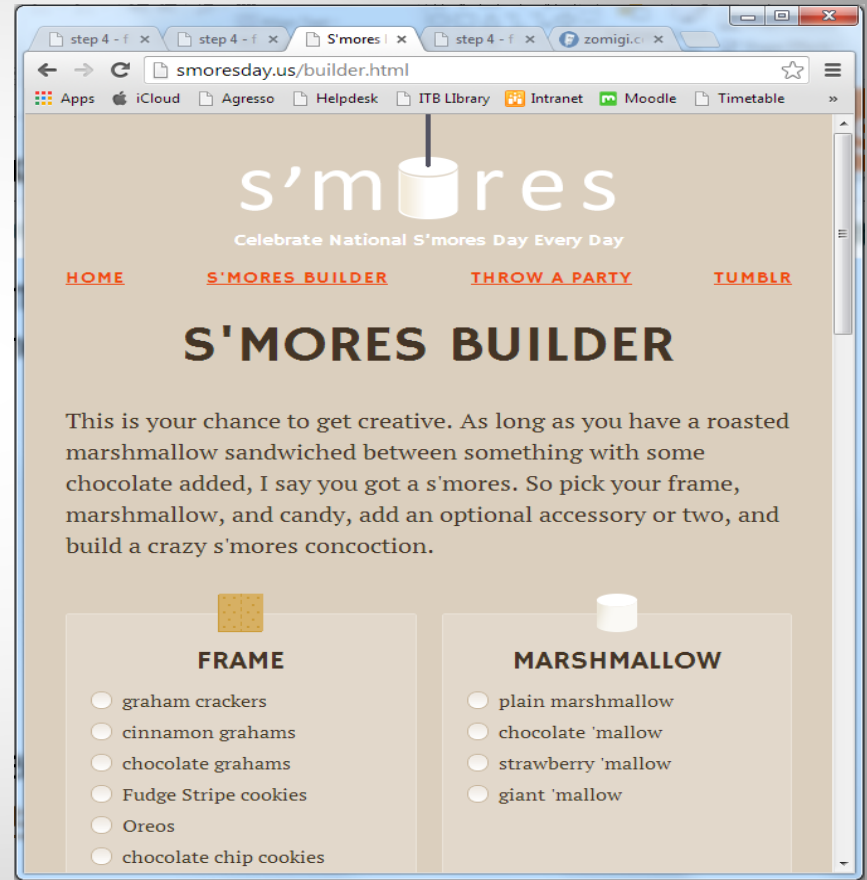
- 2 simple rules

- one for 'parent' , and one for 'child'

```
nav ul {  
    display: flex;  
}  
  
nav li {  
    flex: 1;  
}
```

- To learn more see:

- <http://smoresday.us/builder.html>



To cope with older Chrome version add the **-webkit-** version of each CSS property



```
nav ul {  
    display: flex;  
    display: -webkit-flex;  
}
```

```
nav li {  
    flex: 1;  
    -webkit-flex: 1;  
}
```



Wrapping of items for narrow windows ...

Enable 'wrapping' of nav items if a narrow browser window ...

horiz-wrap



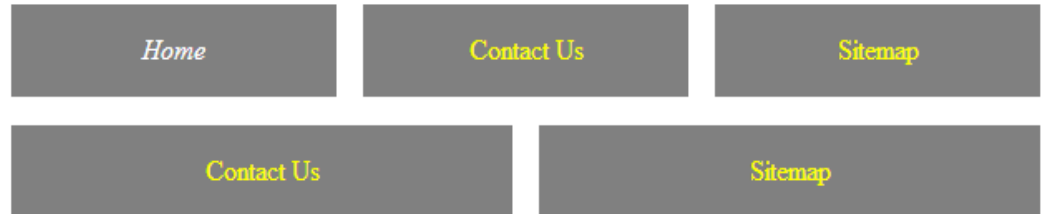
- 2 simple rules
 - one for 'parent' : wrap items if they cannot all fit
 - and one for 'child' : set minimum width for items

```
nav ul {  
  display: flex;  
  flex-wrap: wrap;  
}
```

```
nav ul li {  
  flex: 1 0 12em;  
} /* could replace '12em' with '200px' etc. */
```

allow WRAPPING (onto next line) of the flex'd items inside 'ul'
NOTE: need to set MIN width greater than default 0px, to allow WRAP of nav bar items when browser window is narrow

try changing the WIDTH of this browser window
and see the nav bar items 'wrap' onto the next line
(and automatically stretch to use up all the width :-)



And here is **-webkit-** version ...



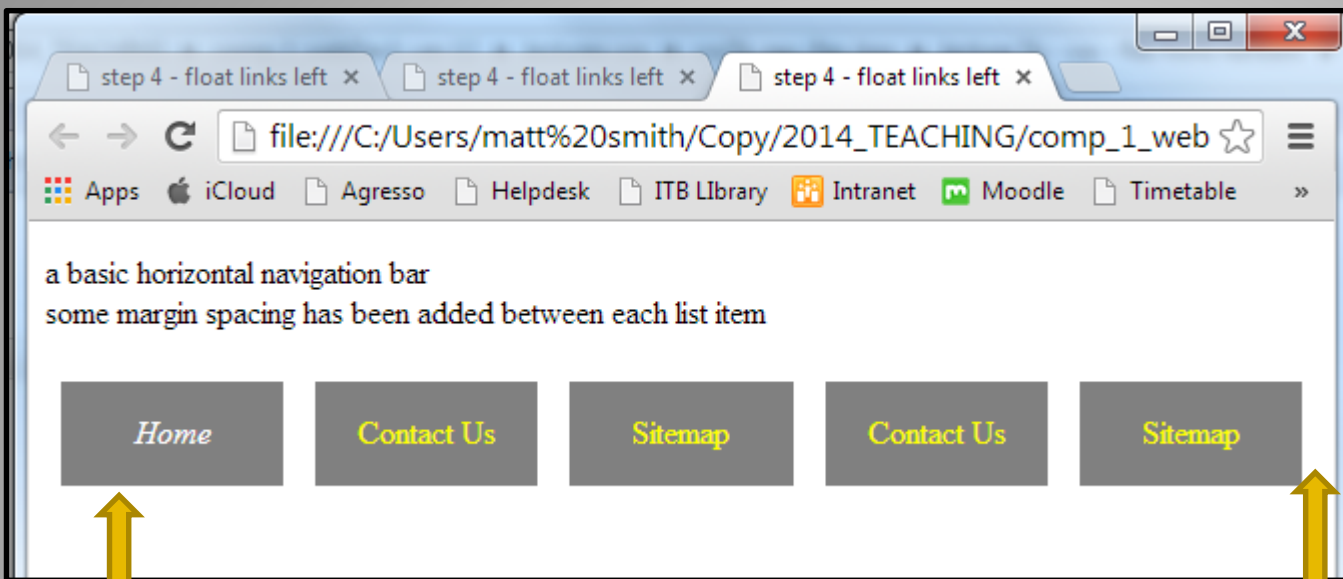
```
nav ul {  
  display: flex;  
  display: -webkit-flex;  
  flex-wrap: wrap;  
  -webkit-flex-wrap: wrap;  
}  
  
nav ul li {  
  flex: 1 0 12em;  
  -webkit-flex: 1 0 12em;  
}
```



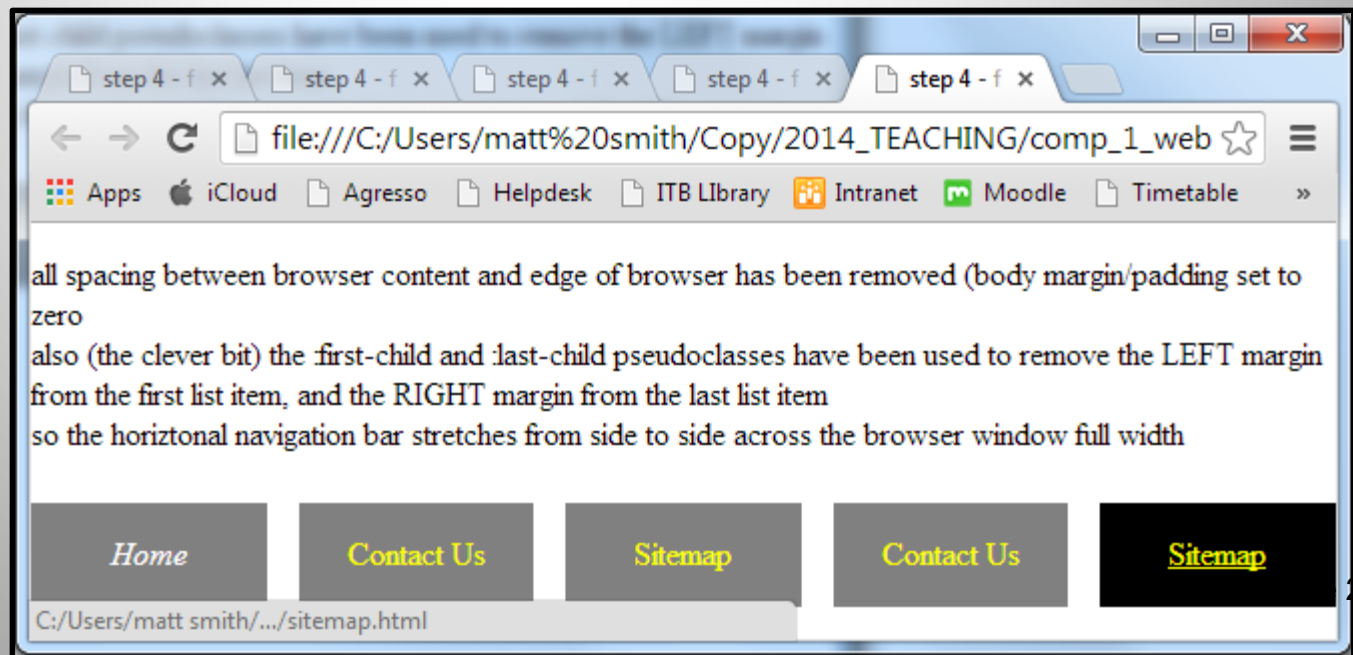
Something a bit clever ...

**Removing left and right margins
with pseudoclasses**

So navbar FULL WIDTH available



- We can remove LEFT margin for FIRST list item
- And remove RIGHT margin for LAST list item



The solution ...



```
/* remove padding/margin around all of 'body'  
   for that 'edge to edge' browser effect    */
```

```
body {  
    margin: 0;  
    padding: 0;  
}  
  
nav li:first-child {  
    margin-left: 0;  
}  
  
nav li:last-child {  
    margin-right: 0;  
}
```

