POSITIONING CONTENT

POSITIONING ELEMENTS FOR LAYOUTS

POSITIONING WITH FLOATS

- `float` removes an element from the normal flow of a page
- It positions an element to the **left** or **right** of its parent element - *it being the floated element*
- **All other elements** on the page will then flow around the floated element
- http://codepen.io/ianshea/pen/OMZggB?editors=1100

POSITIONING WITH FLOATS

- When floating an element, it'll default to the width of the content.
- We can adjust this by adding a `width` property.
- We can adjust white space between elements using the `margin` property
- A combination of `width` and `margin` is needed to position our floated layout items correctly.
- http://codepen.io/ianshea/pen/OMZggB?editors=1100

CLEARING FLOATS

- Content will flow down the side of a floated object.
- **Pitfalls of float:**
  - Proper styles will not render on other elements that are next to floated elements.
- Sometimes unwanted content begins to wrap around a floated element
- To prevent content from wrapping around floated elements, we need to clear, or contain, those floats and return the page to its normal flow.
- By adding a `clear` property, an element will move down past the float.
- *Lets go back and fix our example…*

CLEAR PROPERTY

- Has 3 values
  - `left`
    Clears `left` floats
  - `right`
    Clears `right` floats
  - `both`
    Clears both `left` and `right` and is most often used.

CONTAINING FLOATS - CLEARFIX

- Our reading calls this containing floats, it’s most often referred to as “clear fix” or “clearfix”
- The parent element of our floated elements acts as a container
- We add in some CSS which clears any floated elements and returns the flow of the document back to normal
- http://codepen.io/ianshea/pen/EPLpdW?editors=1100

WHAT CLEARFIX IS DOING

- Clearing any floated elements, within the element with the `.clearfix` class and returning the flow back to normal.
- The :pseudo selectors dynamically generate elements above and below the section with the property `display: table;`
- The element with `.clearfix` on it clears any floated elements above it.
- All to return us to the normal document flow

LAYOUT HELPERS

- `clearfix` is a layout helper.
- Grids are layout helpers as well.
  - Popular grids:
    - Bootstrap http://getbootstrap.com/css/#grid-example-basic
    - Foundation http://foundation.zurb.com/sites/docs/v/5.5.3/components/grid.html
- We can bring in these additional CSS rules/files to utilize these in our layouts.
- We do not have to do all the heavy lifting for every project.
- CSS should be modular

POSITION PROPERTY

- Sometimes we want to be **precise** in positioning elements.
- `position:`

  ```
  static (default)
  relative
  absolute
  fixed
  inherit
  ```

POSITION PROPERTY

- `position: static;`
- Static is the default value for elements.
- You use `static` for simple, single-column layouts where each element sits on top of the next one.
- http://codepen.io/ianshea/pen/OMZozP?editors=1100

POSITION PROPERTY

- `position: relative;`
- `relative` works very similar to `static` except…
- We can use offset properties like;
  `top`, `right`, `left`
- http://codepen.io/ianshea/pen/OMZoZV?editors=1100

POSITION PROPERTY

- `position: relative;`
- `relative` items create a “coordinate system” for child elements
- This “coordinate system” is a reference point for offset properties
- http://codepen.io/ianshea/pen/OMZoZV?editors=1100

POSITION PROPERTY

- `position: absolute;`
- `absolute` also items create a “coordinate system” for child elements
- Removes the element from the normal document flow.
- http://codepen.io/ianshea/pen/KVRGvR?editors=1100

POSITION PROPERTY

- `position: fixed;`
- Shares same concepts as an `absolute` element
- Removes the element from the normal document flow.
- Fixes an element to a position on the page.
- http://codepen.io/ianshea/pen/mVLzgw?editors=1100

POSITION PROPERTY

- `position: inherit;`
- Tells child properties to inherit the position property of it’s parent element.

POSITION PROPERTY

- `z-index: number;`
- Controls the stacking order of positioned elements.
- http://codepen.io/ianshea/pen/YwLRwQ?editors=1100

REFERENCE

- http://alistapart.com/article/css-positioning-101
- http://learn.shayhowe.com/html-css/positioning-content/