Measurments on the web with css and javascript

How do we measure height and width for your layout and

Javascript

clientWidth/Height

Measures the inner width/height of an element in pixels. It includes padding but not the vertical/Horizontal scrollbar (if present, if rendered), border or margin.

```javascript
let clientX = document.documentElement.clientWidth;
let clientY = document.documentElement.clientHeight;
```

InnerWidth/Height

Width/Height (in pixels) of the browser window viewport including, if rendered, the vertical/Horizontal scrollbar if presented and rendered.

```javascript
let windowX = window.innerWidth;
let windowY = window.innerHeight;
```

Safe bet the bigest of ClientWidth/Height and innerWidth/Height

The safest way to measure viewport dimensions is to use the largest of clientWidth / innerWidth and clientHeight / innerHeight, something like this

```javascript
let windowW = Math.max(clientX, windowX);
let windowH = Math.max(clientY, windowY);

console.log(`There biggest values are ${windowW} x ${windowH}`);
```

Making this into a function

Rather than make the calculation eveerytime we need it and run it again when the screen changes either because we resize the window in desktop (which has no effect in mobile) or change the display orientation (useful in mobile and some desktop monitors that can turn from vertical to horizontal).

We first initialize variables to false or 0 in the global scope.

We then calculate the values for the different elements inside the `calcScreen` function.

We run the function.

The most important parts of this version are the event listeners for `resize` and or `ientationchange`. When either of those events trigger we want to rerun the function to get updated values.

```
let screenW = screenH = windowW = windowH = false;

var calcScreen = function() {
  screenW = screen.width;
  screenH = screen.height;
  windowW = Math.max(document.documentElement.clientWidth, window.innerWid
  windowH = Math.max(document.documentElement.clientHeight, window.innerHe
};

calcScreen();
window.addEventListener("resize", calcScreen());
window.addEventListener("orientationchange", calcScreen());
```

CSS

Viewport Units

- http://thenewcode.com/660/Using-vw-and-vh-Measurements-In-Modern-Site-Design
- http://thenewcode.com/1137/MinMaxing-Understanding-vMin-and-vMax-

[in-CSS](#)