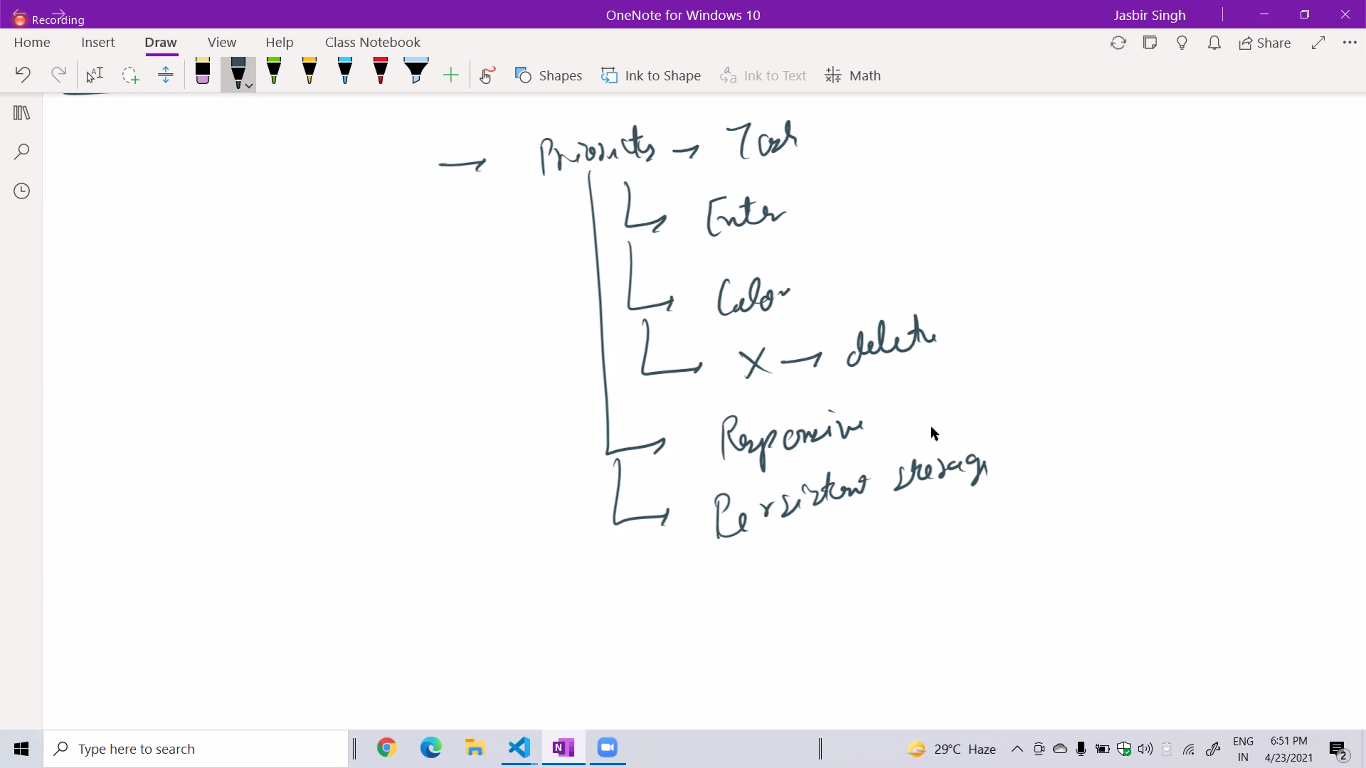
browser gives DOM api, DOM api creates HTML document, in document we have all HTML element (nodes) accessible in a tree form, having relation like: root, child node, sibling node, deccendants node

create: document.createElement(“\_\_tag\_\_”) // document.createElement(“li”)

read: document.getElementById(), document.querySelector()

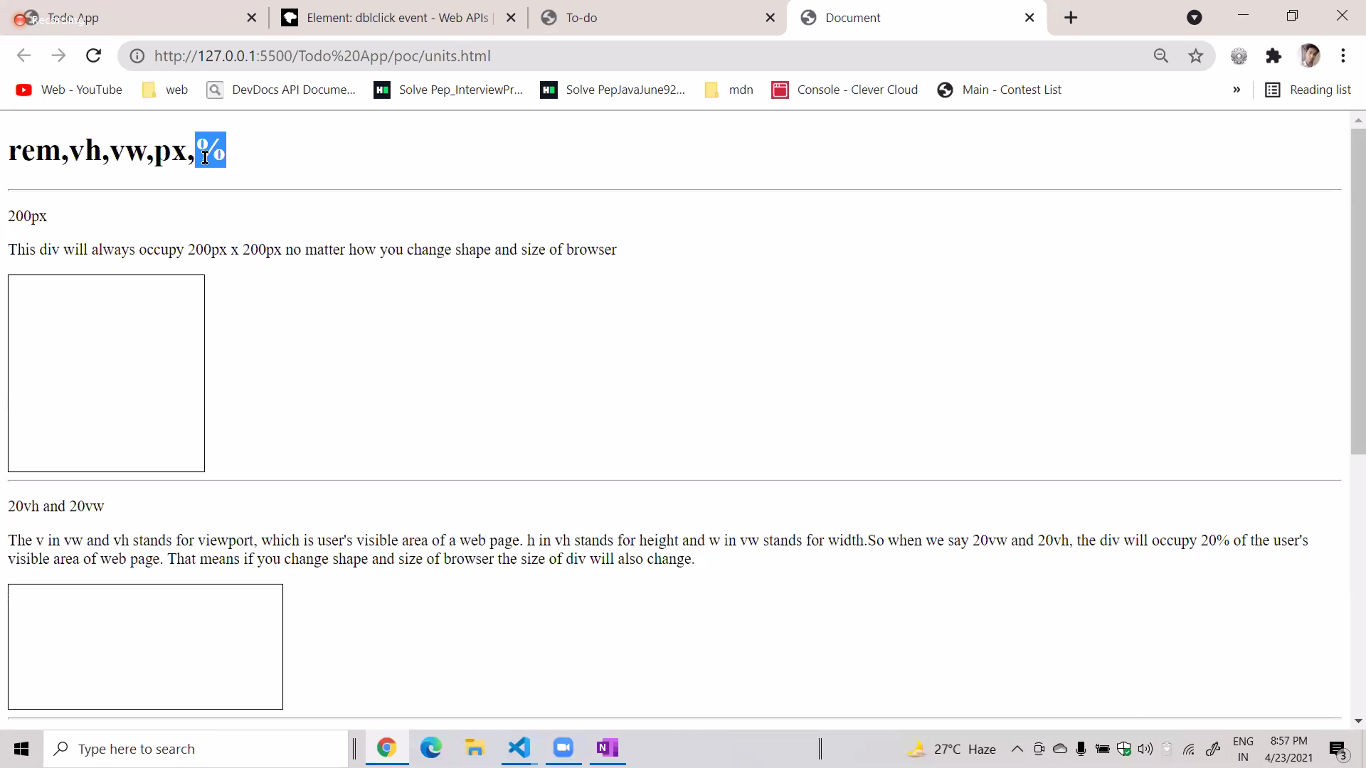
update: change prop/attribute/value of ele: like innerText, innerHTML OR use document.appendChild(“\_\_node created\_\_”), ele.setAttribute(“class”,”task”);

delete: \_\_element\_\_.remove();

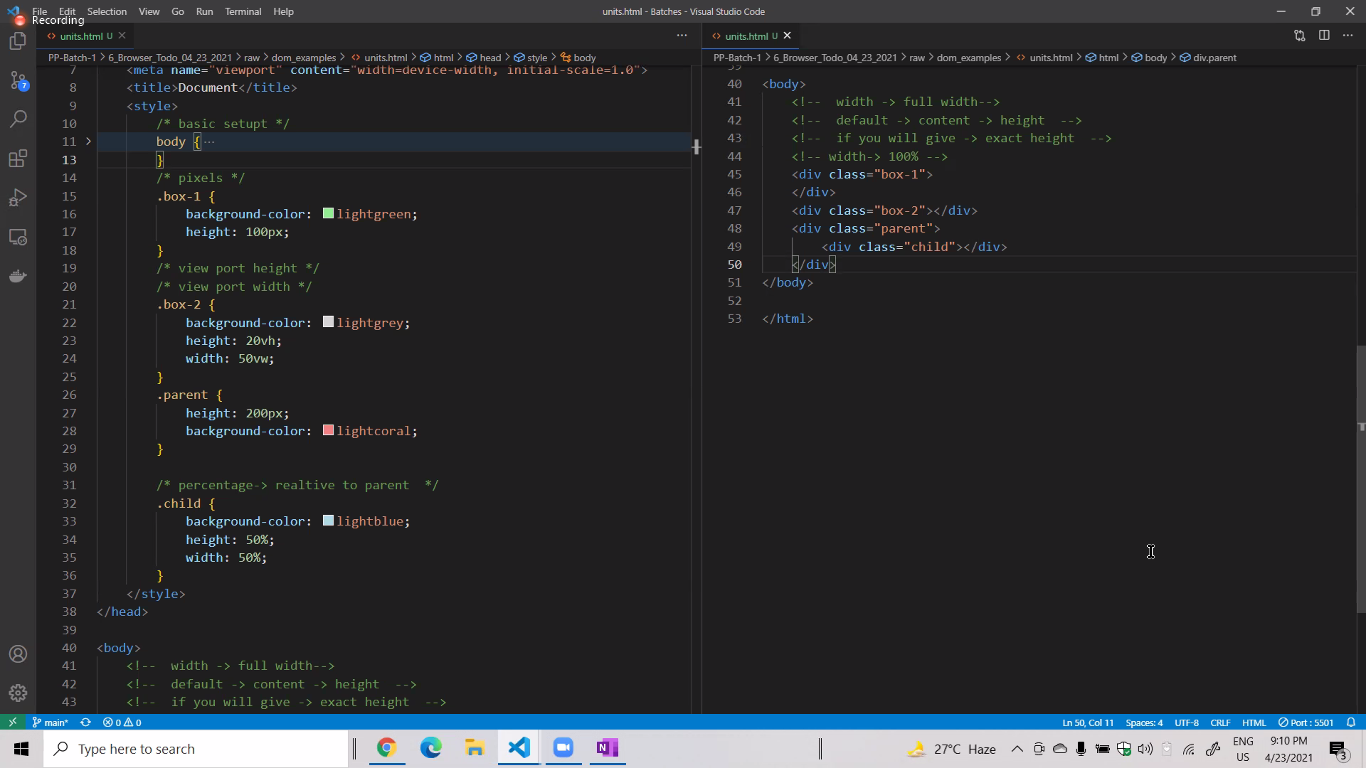


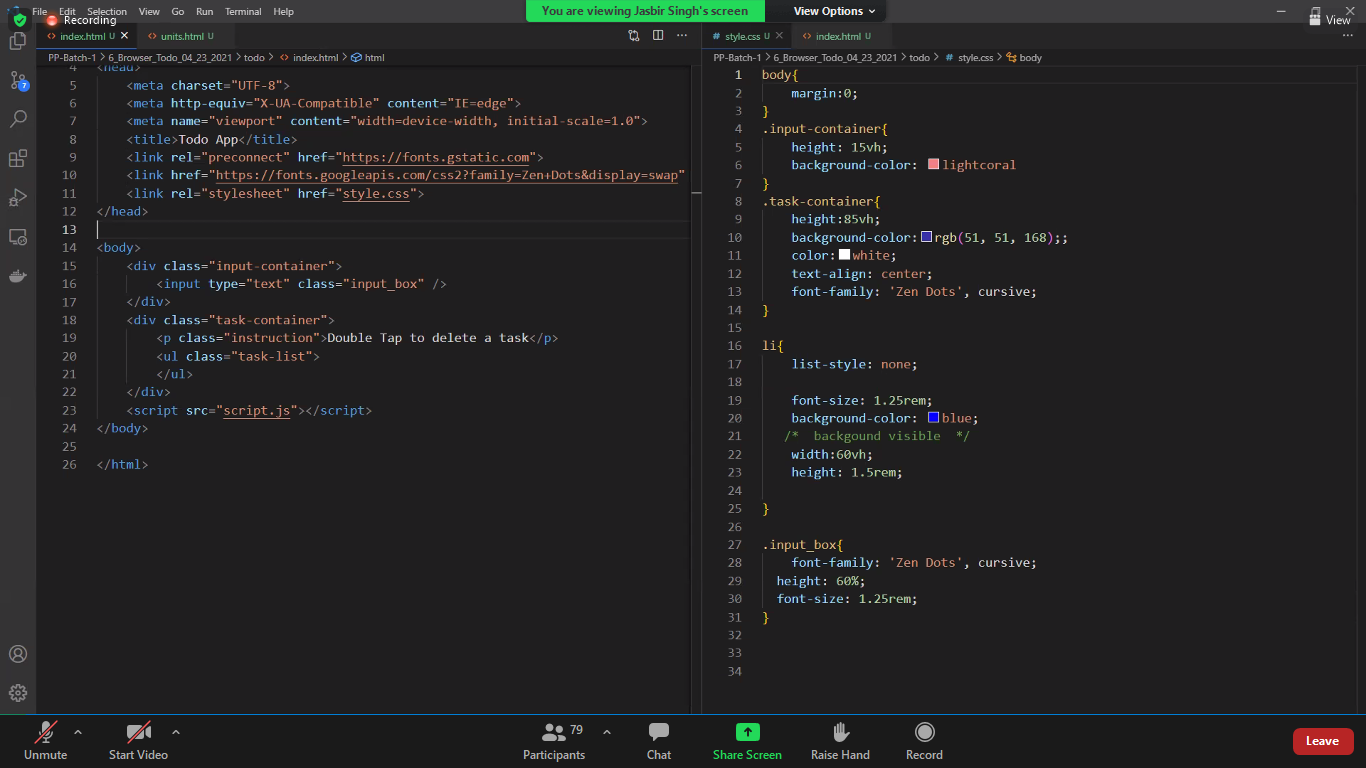
Make clean UI

and responsive webapp



div tag takes full width





rem for height setting or font size setting

box size calculation, height=ele ht + paddin ht (top&bottom) + border ht (top&bottom)

but this can be overridden if we use

\*{ //\* means for all elements

box-sizing: border-box; //now padding and border are all deducted from the element ht and width

}

display: 3 types : display: block, inline-block, inline

**margin: 0 auto** means margin on top & rt is 0 whereas margin on left and right is same, works for block element

**To make website responsive use media query**

media query is added to style.css to make it responsive from all screen size

@media

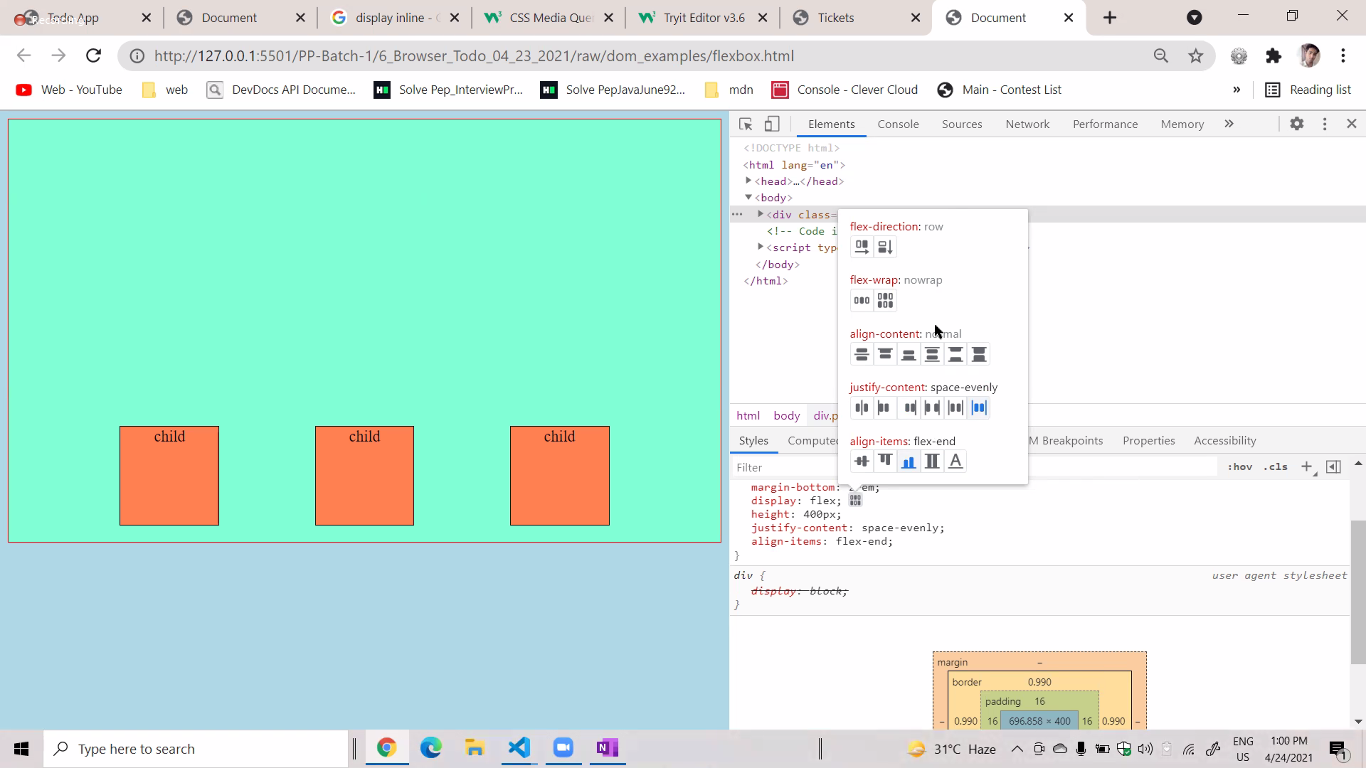
**FLEXBOX**

**PARENT APPLYING TO ITS CHILDREN**

parent style🡪 display: flex

to align children horizontally use : justify content: \_\_value

to align



[**https://flexboxfroggy.com/**](https://flexboxfroggy.com/)**, <https://css-tricks.com/snippets/css/complete-guide-grid/>, https://css-tricks.com/snippets/css/a-guide-to-flexbox/**

**Positioning:**

**position: relative** we can move the element relative to its original position, **no other ele can take the position of the moved element;**

Example: .box{

position:relative;

top: 50px; **//moves to bottom**

left: 50px;

} //**position relative is used when ele’s position can’t be taken by other elements**

**position: absolute:**

**MAKES A NEW LAYER, IF PARENT’S POSITION IS NOT DEFAULT THIS LAYER IS AS PER PARENT’S POSITION, POSITION.ABSOLUTE WHEN USED THE ELEMENT GIVES UP ITS ORIGINAL POSITION UNLIKE POSITION.RELATIVE**

.box2{

position:absolute //make new layer, other element can take its pos, as now box2 is in new layer, this layer is starting at (0,0) of page, if parent ele has different position, children will get abs position accordingly,

top: 50px; //move ele 50px to the bottom of new layer

left: 50px; //move ele 50px to the left of new layer

}