

CSS Part 2 Exercises

Revision of previous class

Challenge [8 minute]

Create this pricing card mocked below

BASIC

₹1900/mo

Our Basic plan is perfect for small teams and startups looking to manage a few projects.

- ✓ 5 Projects
- ✓ 10GB Storage
- ✗ Advanced Analytics
- ✗ Priority Support
- ✓ Unlimited Users
- ✓ 24/7 Support

GET STARTED

skeleton

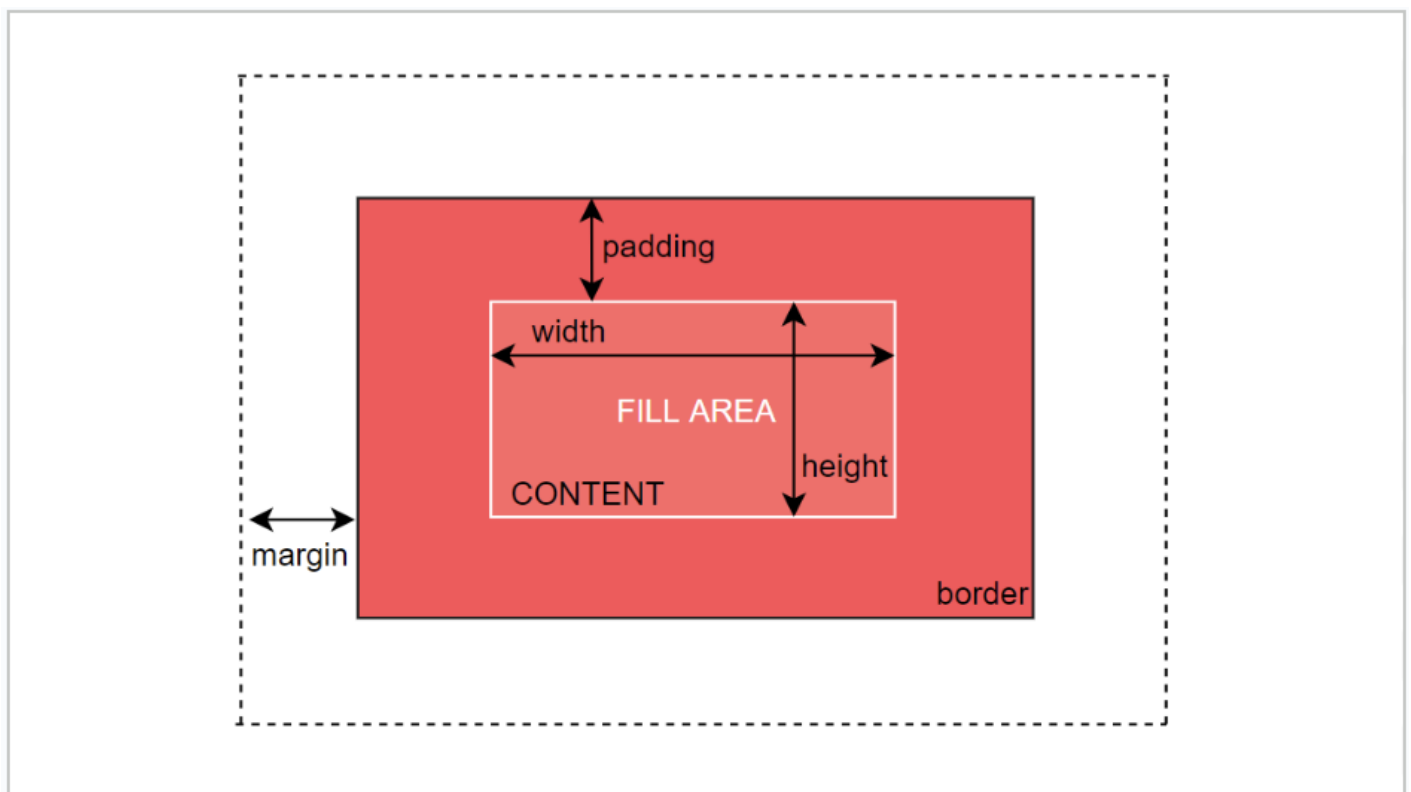
<https://codesandbox.io/s/pricing-card-question-lb0zm9?file=/index.html>

solution

<https://codesandbox.io/s/pricing-card-solution-my98m1?file=/src/styles.css>

Box model

Understanding



Most common and most misunderstood concept

Made up of:

- Content Area
- Padding
- Fill area (Content area + padding)
- Border

- Margin

Examples :

- A building
 - carpet area == content area
 - walls == padding
 - (walls and the carpet area) / built up area == fill area
 - paint == border
 - free space between two building blocks == margin

Check the box model for each of the elements in the developer console

Background applies to the fill area

Box-sizing

- By default browser applies the width to the content-area and default box-sizing value is content-box
- We can change this to take the width of the entire box including border and padding by setting `box-sizing` to `border-box`
- [Illustration code sandbox](#)
- you can set this as default in your reset

```
*,
*::before,
*::after {
  box-sizing: border-box;
}
```

COPY

- pseudo elements are not considered in the universal selector (0,0,0,0)

Challenge

- Create an article with some text
- Parent section has width of 480px
- article should width should be 50% of the parent
- article should have a 6px border
- horizontal padding and margin of 16px and 24px respectively for both section and article

- Solution - <https://codesandbox.io/s/box-sizing-practice-solution-bgouxi>

Outlines are outside border generally used for a11y.

- Outlines don't effect the layout much like box-shadow.
- They are purely cosmetic
- `outline-offset` - gap between box and the outline

Borders:

- `width`[optional - 1 px default] `style`[mandatory] `color`[optional - defaults to current font color]
- explicitly use `currentColor` in css for the same behaviour
- play with different border styles and mixed styles - <https://codesandbox.io/s/border-styles-fnu7en?file=/src/styles.css>

Margins

- margins are the outer space of the box / space between boxes
- Margins could be negative
 - reduces gap between boxes
 - effects the layout and hence effects all the siblings
 - Illustration - <https://codesandbox.io/s/position-1-0o8u0x?file=/src/styles.css>

Challenge

- Question link : <https://codesandbox.io/s/negative-margin-question-pig8m3?file=/src/styles.css>

Header

Lorem ipsum dolor sit amet consectetur adipisicing elit. Optio, dolore eligendi repellendus, delectus animi rem ipsum fuga corporis beatae dicta iure explicabo impedit suscipit quidem autem laudantium, quis tempore tenetur?

- Solution link - <https://codesandbox.io/s/negative-margin-solution-r15rx5?file=/src/styles.css>

Margins have auto property

- Auto for top and bottom is 0
- Only works when width is provided
- <https://codesandbox.io/s/margin-auto-lbeczd?file=/src/styles.css>

Margins can collapse

- <https://codesandbox.io/s/margin-auto-lbeczd?file=/src/styles.css>

Flow layout

Understanding

- Default layout system of the browsers
- A plain HTML document with no author CSS applied uses the flow layout
- Each HTML element has the default flow type
- browsers were meant to emulate digital copies of printed documents
- flow layout is similar to a document layout
- Toggle element with `display`

```
a {  
  display: block;  
}
```
- Flow layout will never try to overlap pixels

COPY

Block elements

- `div`, `article`, `section`, `header`, `h1`, `h2`, `h3`
- The chunks of content that make up a page.
- They don't share space, greedily expand to take the entire space available horizontally.
- by default the block level elements have width `auto` which makes them greedily take the available space.

- the default width is not 100%. its context aware of any margins

Inline elements

- a , b , button, img, input
- Generally, inline elements are meant to highlight bits of text, or elements within a block container.
- Don't effect block layout
- Bunch of css properties do not work, as they are not supposed to.
 - Height width margin-top margin-bottom
- Inline elements can line-wrap
- As the box shape can be weird , because they can line wrap
 - add border to see the actual shape
 - <https://codesandbox.io/s/inline-knowbd?file=/index.html>
- can't give width height . Always adheres to container
- Certain inline are exception. width and height could be set
 - these are called replaced elements
 - img, video, canvas
- sometimes adds extra space as browser considers this as typography

Inline - block elements

- css properties work as it behaves like a block but is placed in an inline way
- Illustration <https://codesandbox.io/s/inline-block-kuhlmu>
- Doesn't line wrap
- Illustration <https://codesandbox.io/s/inline-block-wrap-gfsuob>

Challenge 1

- Use the pricing card to create a stack of 3 cards aligned horizontally
- Question - <https://codesandbox.io/s/inline-cards-question-fmit27>
- Solution - <https://codesandbox.io/s/inline-cards-solution-tt4vpy?file=/index.html>

Positioned layout

Items can overlap in this layout

Types of positions

- default / static, relative, absolute, fixed, sticky

Default

- static is default
- means its not using any positioning system or some other layout
- opt out of the positioned layout

```
.box {  
  /* Revert to its default value, which is 'static' */  
  position: initial;  
}
```

COPY

Relative positioning

Understanding

```
.some-box {  
  position: relative;  
}
```

COPY

- Constraints certain children
 - positioned children will be constrained inside a relative parent
- Enables additional CSS properties to be used.
 - top, left, right, bottom
 - relative to the natural position or in flow position
- The big difference is that position doesn't impact layout and other siblings
 - <https://codesandbox.io/s/position-1-0o8u0x>

Challenge

- use relative positioning to move the card(from previous challenge) top bottom

The image shows three pricing cards for different service plans. Each card has a title, a price, a description, a list of features with checkmarks or crosses, and a 'GET STARTED' button.

Plan	Price	Description	5 Projects	10GB Storage	Advanced Analytics	Priority Support	Unlimited Users	24/7 Support
BASIC	₹1900/mo	Our Basic plan is perfect for small teams and startups looking to manage a few projects.	✓	✓	✗	✗	✓	✓
PREMIUM	₹3900/mo	Our Premium plan is designed for larger organizations with complex project management needs.	✓	✓	✓	✓	✓	✓
PRO	₹2900/mo	Our Pro plan is ideal for growing teams who need more advanced features and support.	✓	✓	✓	✗	✓	✓

- <https://codesandbox.io/s/inline-cards-enhanced-edq8rn?file=/src/styles.css>
- Create the below UI
- skeleton - <https://codesandbox.io/s/relative-practice-skeleton-h58yme?file=/src/styles.css>
- solution - <https://codesandbox.io/s/relative-practice-10r9o7?file=/src/styles.css>

The image shows a promotional banner with the text "Free shipping for you." in a large, bold, purple font. The word "only" is written in a smaller, red, cursive font above the word "shipping". A red arrow points upwards from the word "shipping" to the word "only".

Absolute positioning #

Understanding

- Elements that need to float above others
- here is no effect of display properties
- Also enables top, left, right, bottom properties but relative to containing block and not the in flow position
- Containing block of the absolute element is the first parent with positioning (any of them) turned on, this essentially means that the containing block of an absolutely positioned element need not be its direct parent.
- Will be placed by default as per flow order
- Parent collapses
 - parent becomes empty as the element is pulled out of the flow order
 - Illustration <https://codesandbox.io/s/absolute-gkej70?file=/src/styles.css>

Challenge

- Exercise modal / centring
 - skeleton - <https://codesandbox.io/s/exercise-modal-dialog-skeleton-58jonj?file=/src/styles.css>
 - solution - <https://codesandbox.io/s/exercise-modal-dialog-solution-d0gq3c?file=/src/styles.css>

Fixed positioning

Understanding

- It's absolutely positioned with respect to the viewport.
- Immune to scrolling.

Challenge

- Create header for the pricing cards page
- solution - <https://codesandbox.io/s/inline-cards-enhanced-with-header-8l881u?file=/src/styles.css>
- skeleton - <https://codesandbox.io/s/inline-cards-enhanced-with-header-8l881u?file=/src/styles.css>

Our Plans

BASIC

₹1900/mo

Our Basic plan is perfect for small teams and startups looking to manage a few projects.

✓ 5 Projects

✓ 10GB Storage

✗ Advanced Analytics

✗ Priority Support

✓ Unlimited Users

✓ 24/7 Support

GET STARTED

PREMIUM

₹3900/mo

Our Premium plan is designed for larger organizations with complex project management needs.

✓ 5 Projects

✓ 10GB Storage

✓ Advanced Analytics

✓ Priority Support

✓ Unlimited Users

✓ 24/7 Support

GET STARTED

PRO

₹2900/mo

Our Pro plan is ideal for growing teams who need more advanced features and support.

✓ 5 Projects

✓ 10GB Storage

✓ Advanced Analytics

✗ Priority Support

✓ Unlimited Users

✓ 24/7 Support

GET STARTED