



Live Cohort

DAY - 16

CSS Notes: Boilerplate, Transform, Perspective, Hover, and Transition

CSS Notes

1. CSS Boilerplate

A CSS boilerplate resets browser defaults for consistent layouts and easier styling.



```
1 * {
2   margin: 0;
3   padding: 0;
4   box-sizing: border-box;
5 }
6 html, body {
7   height: 100%;
8   width: 100%;
9 }
10
```

💡 Explanation

- `* { margin: 0; padding: 0; }` → Removes unwanted spacing across browsers.
- `box-sizing: border-box;` → Includes padding & borders within total size.
- `html, body { height: 100%; width: 100%; }` → Ensures full page layout.

💡 Like starting with a clean canvas before painting!

2. Transform (All Properties)

The transform property changes how elements appear — move, scale, rotate, or skew — without affecting layout flow.

2.1 translate()

CSS Notes

Moves an element along the X & Y axes.

```
1  div {  
2      transform: translate(50px, 100px);  
3  }  
4
```

→ Moves element 50px right & 100px down.

2.2 scale()

Resizes the element.

```
1  div {  
2      transform: scale(1.5);  
3  }  
4
```

→ Enlarges by 1.5x.

You can also use:

```
1  scaleX(2); /* wider */  
2  scaleY(0.5); /* shorter */  
3
```

CSS Notes

2.3 rotate()

Rotates the element clockwise.



```
1 div {  
2   transform: rotate(45deg);  
3 }  
4
```

2.4 skew()

Tilts the element.



```
1 div {  
2   transform: skew(20deg, 10deg);  
3 }  
4
```



2.5 matrix()

Combines multiple transforms together.



```
1 div {  
2   transform: matrix(1, 0.5, 0.5, 1, 30, 30);  
3 }  
4
```

→ Complex & rarely used directly.

CSS Notes

💡 Common use: animations — rotating icons, enlarging buttons, sliding images.

3. Perspective

Adds 3D depth for transforms.

```
1 .container {  
2   perspective: 600px;  
3 }  
4 .box {  
5   transform: rotateY(45deg);  
6 }
```

➡ Smaller perspective = stronger 3D effect.

↗ Apply to the parent element.

⌚ Use case: 3D card flips, cube rotations.

4. Hover

The :hover pseudo-class styles elements on mouse hover.

CSS Notes

```
1 button {
2     background-color: dodgerblue;
3     color: white;
4     border: none;
5     padding: 10px 20px;
6     cursor: pointer;
7 }
8 button:hover {
9     background-color: navy;
10    transform: scale(1.1);
11 }
12 }
```

💡 Use case: interactive buttons, icons, links.

5. Transition

Makes property changes smooth & animated.

Syntax:

```
1 transition: property duration timing-function delay;
```

Example:

CSS Notes



```
1 button {  
2     background-color: royalblue;  
3     transition: background-color 0.3s ease, transform 0.3s ease;  
4 }  
5 button:hover {  
6     background-color: navy;  
7     transform: scale(1.1);  
8 }  
9
```

Common Properties:

- transition-property → What to animate
- transition-duration → How long it takes
- transition-timing-function → Ease, linear, etc.
- transition-delay → Wait before start

💡 Use case: cards, modals, buttons — for smooth UI motion.

CSS Notes

Summary

Concept	Description	Real-life Example
Boilerplate	Resets default styles	Clean layout base
Transform	Move, rotate, scale, skew	Animation effects
Perspective	Adds 3D depth	3D card flip
Hover	Triggers on mouse hover	Button color change
Transition	Smooth effect change	Animated hover button