

Training Day 15

Day 15– 11th July 2025

CSS Animations and Transitions

Detailed Description:

On Day 15, we explored **CSS animations and transitions**, which make web pages **dynamic and visually engaging**.

The instructor emphasized that subtle animations improve **user interaction**, highlight important content, and make websites **modern and professional**.

◆ 1. Introduction to CSS Transitions

CSS Transitions allow elements to **change smoothly from one style to another** over a specific duration. We learned the basic syntax:

```
selector {  
  
    transition: property duration timing-function delay;  
  
}
```

Example practiced:

```
button {  
  
    background-color: teal;  
  
    color: white;
```

```
padding: 10px 20px;

transition: background-color 0.5s ease;

}
```

```
button:hover {

background-color: orange;

}
```

- Hovering over the button smoothly changes its background color.
- Transition properties include all, background-color, transform, opacity, etc.

◆ 2. CSS Transformations

Transformations allow **movement, scaling, and rotation** of elements.
We practiced common transformations:

- **Translate** – Moves elements along X and Y axes

```
div:hover {

transform: translateX(50px);

}
```

- **Scale** – Enlarges or shrinks elements

```
div:hover {
```

```
transform: scale(1.2);  
  
}
```

- **Rotate** – Rotates elements

```
div:hover {  
  
transform: rotate(15deg);  
  
}
```

- **Skew** – Tilts elements

```
div:hover {  
  
transform: skewX(20deg);  
  
}
```

Transformations combined with transitions create **smooth visual effects**.

◆ 3. CSS Animations

CSS Animations allow elements to **change styles repeatedly or in sequence**.

We learned the `@keyframes` rule to define animations:

Example practiced:

```
@keyframes example {  
  
0% {background-color: red; left: 0px;}  
  
50% {background-color: yellow; left: 100px;}  
  
}
```

```
100% {background-color: green; left: 0px;}  
  
}
```

```
div {  
  
    width: 100px;  
  
    height: 100px;  
  
    position: relative;  
  
    animation-name: example;  
  
    animation-duration: 4s;  
  
    animation-iteration-count: infinite;  
  
}
```

- The box **moves and changes color repeatedly**.
- Animations can have properties like duration, delay, iteration-count, direction, and timing-function.

◆ 4. Practical Exercises

- Created **hover effects on buttons** using transitions.
- Animated **banners and text** using @keyframes.
- Practiced combining **transformations and transitions** for interactive elements.
- Explored **infinite looping animations** for banners and attention-grabbing effects.

Learning Outcomes:

- Learned to apply **CSS transitions** for smooth style changes.
- Practiced using **transformations** like translate, scale, rotate, and skew.
- Learned the basics of **CSS animations** using @keyframes.
- Combined animations and transformations for **dynamic and interactive designs**.
- Developed skills to enhance **user engagement and visual appeal** on webpages.