CSS Transformations

CSS **transforms** allow you to visually manipulate elements by rotating, scaling, skewing, or translating them. Transforms are applied using the `transform` property.

---

1. `transform` Property

Syntax:

```
selector {
  transform: function(value);
}
```

Multiple functions can be combined:

```
transform: translateX(50px) rotate(45deg) scale(1.2);
```

---

2. Types of Transformations

a. `translate()`

Moves an element from its current position.

```
.box {
  transform: translateX(50px); /* Move 50px to the right */
}
```

Other variations:

- `translateY(30px)` – moves vertically
- `translate(50px, 30px)` – moves on both axes

b. `rotate()`

Rotates the element clockwise by default.

```
.box {
  transform: rotate(45deg); /* Rotate 45 degrees */
}
```

Use negative values to rotate counter-clockwise:

```
transform: rotate(-45deg);
```

c. `scale()`

Scales the size of an element.

```
.box {
  transform: scale(1.5); /* Increase size by 1.5x */
}
```

- `scaleX(2)` – scales horizontally
- `scaleY(0.5)` – scales vertically

d. `skew()`

Slants an element along the X and/or Y axis.

```css
.box {
  transform: skew(20deg, 10deg); /* Skew in X and Y */
}
```

Individual axis:

- `skewX(20deg)`
- `skewY(10deg)`

---

e. `matrix()`

A shorthand to apply multiple transformations using a 2D matrix. Rarely used directly because it’s less readable.

---

3. Transform Origin

By default, transforms are applied relative to the **center** of the element. You can change this with `transform-origin`.

```css
.box {
  transform: rotate(45deg);
  transform-origin: top left;
}
```

---

4. Combining Multiple Transforms

```css
.box {
  transform: translateX(100px) rotate(30deg) scale(1.2);
}
```

The **order matters**: transforms are applied from left to right.

5. 3D Transforms (Intro Only)

- `rotateX()`, `rotateY()`, and `rotateZ()` add 3D rotation.
- `perspective` property is needed to see 3D depth.

Example:

```
.box {
  transform: rotateY(45deg);
  transform-style: preserve-3d;
}
```

Summary

Transform Function	Description
`translate()`	Moves element
`rotate()`	Rotates element
`scale()`	Resizes element
`skew()`	Slants element
`matrix()`	Combines multiple transforms

CSS Transforms are foundational for building modern UI effects — often combined with **transitions** and **animations**.