

# CSS Property With example

## 1. Text Properties

- **Font**

```
p {  
  font-family: 'Arial', sans-serif; /* Sets the font family */  
  font-size: 16px; /* Sets the size of the font */  
  font-style: italic; /* Italicizes the text */  
  font-weight: bold; /* Makes the text bold */  
  font-variant: small-caps; /* Displays text in small-caps */  
}
```

- **Text**

```
h1 {  
  color: #333; /* Sets the text color */  
  text-align: center; /* Centers the text */  
  text-decoration: underline; /* Underlines the text */  
  text-transform: uppercase; /* Transforms text to uppercase */  
  letter-spacing: 2px; /* Adds space between letters */  
  line-height: 1.5; /* Sets the line height */  
  word-spacing: 4px; /* Adds space between words */  
  white-space: nowrap; /* Prevents text from wrapping */  
  text-shadow: 2px 2px 5px rgba(0,0,0,0.3); /* Adds a shadow to the text */  
}
```

- **List**

```
ul {  
  list-style-type: square; /* Sets the bullet style to square */  
  list-style-position: inside; /* Positions bullets inside the list */  
  list-style-image: url('bullet.png'); /* Uses an image as a bullet */  
}
```

## 2. Box Model Properties

- **Margin**

```
.box {  
    margin: 20px; /* Sets equal margins on all sides */  
    margin-top: 10px; /* Sets the top margin */  
    margin-right: 15px; /* Sets the right margin */  
    margin-bottom: 10px; /* Sets the bottom margin */  
    margin-left: 15px; /* Sets the left margin */  
}
```

- **Padding**

```
.content {  
    padding: 20px; /* Adds equal padding on all sides */  
    padding-top: 10px; /* Adds padding at the top */  
    padding-right: 15px; /* Adds padding on the right */  
    padding-bottom: 10px; /* Adds padding at the bottom */  
    padding-left: 15px; /* Adds padding on the left */  
}
```

- **Border**

```
.bordered-box {  
    border: 2px solid #000; /* Adds a solid black border */  
    border-radius: 5px; /* Rounds the corners */  
    border-top: 2px dashed red; /* Top border is dashed red */  
    border-right: 3px dotted green; /* Right border is dotted green */  
}
```

- **Dimension**

```
.container {  
    width: 100%; /* Sets the width to 100% of the parent */  
    height: 300px; /* Sets a fixed height */  
    max-width: 1200px; /* Maximum width */  
    min-width: 600px; /* Minimum width */  
    max-height: 500px; /* Maximum height */  
    min-height: 200px; /* Minimum height */  
}
```

```
}
```

- **Outline**

```
.button {  
    outline: 2px solid blue; /* Adds an outline */  
    outline-offset: 5px; /* Adds space between element and outline */  
}
```

### 3. Background Properties

```
.body {  
    background-color: #f0f0f0; /* Sets the background color */  
    background-image: url('background.jpg'); /* Sets a background image */  
    background-position: center; /* Centers the background image */  
    background-size: cover; /* Scales the background image */  
    background-repeat: no-repeat; /* Prevents background image from repeating */  
    background-attachment: fixed; /* Fixes the background image */  
    background-clip: padding-box; /* Background confined to padding box */  
    background-origin: border-box; /* Background starts from the border box */  
}
```

### 4. Positioning Properties

```
.box {  
    position: relative; /* Sets the position relative to its normal position */  
    top: 10px; /* Moves the box 10px down */  
    right: 20px; /* Moves the box 20px to the left */  
    z-index: 100; /* Stacks the element above others */  
    float: left; /* Floats the element to the left */  
    clear: both; /* Clears floating elements */  
    overflow: hidden; /* Hides overflowed content */  
}
```

### 5. Display Properties

```
.container {  
    display: block; /* Displays as a block element */  
    visibility: hidden; /* Hides the element but keeps its space */  
}
```

```
opacity: 0.5; /* Sets transparency level */  
vertical-align: middle; /* Aligns the element vertically */  
box-sizing: border-box; /* Includes padding and border in the element's width and height */  
}
```

## 6. Flexbox Properties

- **Container**

```
.flex-container {  
  display: flex; /* Enables flexbox on the container */  
  flex-direction: row; /* Lays out children in a row */  
  flex-wrap: wrap; /* Allows flex items to wrap onto multiple lines */  
  justify-content: space-between; /* Distributes space between items */  
  align-items: center; /* Aligns items vertically */  
}
```

- **Items**

```
.flex-item {  
  order: 2; /* Changes the order of the flex item */  
  flex-grow: 1; /* Allows the item to grow to fill available space */  
  flex-shrink: 1; /* Allows the item to shrink if necessary */  
  flex-basis: 200px; /* Sets the initial size of the item */  
  align-self: flex-start; /* Aligns the item at the start of the container */  
}
```

## 7. Grid Properties

- **Container**

```
.grid-container {  
  display: grid; /* Enables grid layout */  
  grid-template-columns: 1fr 2fr; /* Defines two columns with different sizes */  
  grid-template-rows: 100px auto; /* Defines rows with fixed and flexible sizes */  
  gap: 10px; /* Adds space between grid items */  
  justify-items: center; /* Centers grid items horizontally */  
  align-items: stretch; /* Stretches grid items to fill the cell */  
}
```

- **Items**

```
.grid-item {  
    grid-column: 1 / 3; /* Spans the item across two columns */  
    grid-row: 2 / 4; /* Spans the item across two rows */  
    justify-self: end; /* Aligns the item to the right end */  
    align-self: start; /* Aligns the item to the top */  
}
```

## **8. Table Properties**

```
table {  
    table-layout: fixed; /* Fixes the table layout */  
    border-collapse: collapse; /* Collapses table borders into a single border */  
    border-spacing: 10px; /* Adds space between table cells */  
    caption-side: bottom; /* Moves the table caption to the bottom */  
    empty-cells: hide; /* Hides empty cells */  
}
```

## **9. Animation Properties**

```
@keyframes slidein {  
    from {  
        transform: translateX(100%);  
    }  
    to {  
        transform: translateX(0%);  
    }  
}
```

```
.box {  
    animation-name: slidein; /* Refers to the keyframe animation */  
    animation-duration: 2s; /* Sets the duration of the animation */  
    animation-timing-function: ease-in-out; /* Smoothly accelerates and decelerates */  
    animation-delay: 1s; /* Delays the start of the animation */  
    animation-iteration-count: infinite; /* Repeats the animation infinitely */  
}
```

```
    animation-direction: alternate; /* Alternates the animation direction */
    animation-fill-mode: forwards; /* Retains the end state after animation */
}
```

## 10. Transition Properties

```
.button {
    transition-property: background-color, transform; /* Specifies properties to transition */
    transition-duration: 0.3s; /* Sets the duration of the transition */
    transition-timing-function: ease-in-out; /* Smooth transition */
    transition-delay: 0.1s; /* Adds a slight delay before starting */
}
```

```
.button:hover {
    background-color: #ff6347; /* Changes background color on hover */
    transform: scale(1.1); /* Scales the button on hover */
}
```

## 11. Transform Properties

```
.box {
    transform: rotate(45deg); /* Rotates the element by 45 degrees */
    transform-origin: center; /* Sets the transformation origin */
    perspective: 500px; /* Adds perspective to the 3D effect */
    backface-visibility: hidden; /* Hides the backface of the element */
}
```

## 12. Other Visual Properties

```
.element {
    visibility: visible; /* Makes the element visible */
    cursor: pointer; /* Changes the cursor to a pointer */
    overflow: hidden; /* Hides overflowing content */
    resize: both; /* Allows the element to be resized both horizontally and vertically */
    box-shadow: 0px 4px 6px rgba(0, 0, 0, 0.1); /* Adds a subtle shadow */
    text-shadow: 2px 2px 4px #000000; /* Adds a shadow to the text */
    filter: blur(2px); /* Applies a blur effect */
}
```

```
clip-path: circle(50%); /* Clips the element into a circle */
}
```

### 13. Content Properties

```
blockquote:before {
  content: "''"; /* Inserts custom content before the element */
  font-size: 2em;
}
```

```
ol {
  counter-reset: list-counter; /* Resets the counter */
}
```

```
li {
  counter-increment: list-counter; /* Increments the counter */
  list-style: none;
}
```

```
li:before {
  content: counter(list-counter) ". "; /* Displays the counter before list items */
}
```

### 14. Media Queries & Responsive Design

```
@media (max-width: 768px) {
  .responsive-container {
    flex-direction: column; /* Changes flex direction on small screens */
  }
}
```

```
@supports (display: grid) {
  .grid-container {
    display: grid; /* Uses grid layout if supported */
  }
}
```

```
}
```

```
@keyframes fadeIn {
```

```
  from {
```

```
    opacity: 0;
```

```
  }
```

```
  to {
```

```
    opacity: 1;
```

```
  }
```

```
}
```

## 15. Miscellaneous Properties

```
.element {
```

```
  all: unset; /* Resets all inherited properties */
```

```
  will-change: transform; /* Hints to the browser of impending changes */
```

```
  appearance: none; /* Removes default styling */
```

```
  user-select: none; /* Disables text selection */
```

```
  pointer-events: none; /* Disables all pointer events */
```

```
  mix-blend-mode: multiply; /* Defines how content blends with the background */
```

```
}
```