

CSS Overflow

1. Overflow:

- a. The **overflow** property is used to control what happens when the content inside a box (like a `<div>`) is too big to fit in the space you've given it (height or width).
- b. With overflow, you can control whether to clip content or add scrollbars when an element's content is too big to fit in a specified area.

2. It is a shorthand for:

- a. It is shorthand for the **overflow-x** and **overflow-y** properties.
- b. **Note** ⇒ that this property only works for *block elements* with a specified height.

3. Syntax:

```
.box {  
    overflow: value;  
}
```

4. What can overflow do?

- a. When content is too big for a box, you can decide:
 - Should the content be cut off (hidden)?
 - Should the content flow outside the box?
 - Should you add scrollbars so users can scroll and see it?

5. Values:

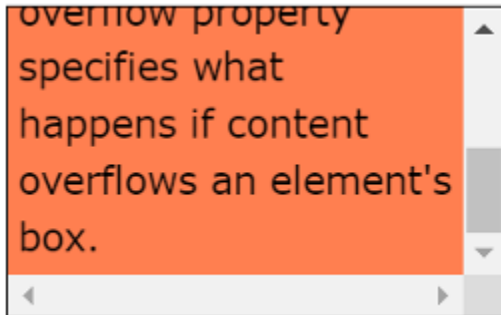
- a. *visible*: This is the default value of the property. Content is not cut off. It will overflow outside the box.

You can use the overflow property when you want to have better control of the layout. The overflow property specifies what happens if content overflows an element's box.

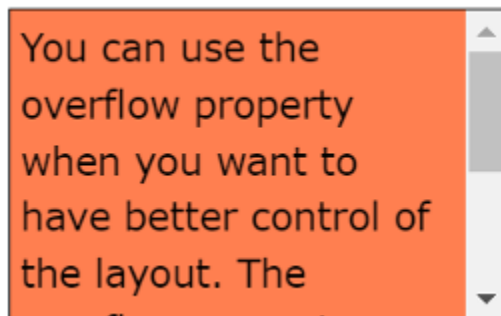
- b. *hidden*: Content that overflows is hidden. Extra content is cut off and not shown. No scrollbar will appear.

You can use the overflow property when you want to have better control of the layout. The overflow property specifies what

- c. *scroll*: Content is cut off, but you can scroll to see the hidden parts. Scrollbars always appear, even if they're not needed.



- d. *auto*: Only adds scrollbars when needed (if content is too much). Smart and most commonly used.



6. `overflow-x`:

- a. Sometimes you only want to control one direction.
- b. `overflow-x` controls how content behaves horizontally (left to right) if it overflows the container.

7. `overflow-y`:

- a. `overflow-y` controls how content behaves vertically (top to bottom) if it overflows the container.

8. Example ⇒

```
.box {  
  overflow-x: auto; /* scrolls left-right if needed */  
  overflow-y: hidden; /* hides top-bottom overflow */  
}
```