# ARIA: aria-level attribute


- Description
  - With heading role
  - Within treegrid role
- Values
- Associated interfaces
- Associated roles
- Specifications
- See also


The `aria-level` attribute defines the hierarchical level of an element within a structure.

# Description

Levels of hierarchy appear in headings, trees, nested grid, nested tablists, and more. If the DOM ancestry does not accurately represent the level, the `aria-level` attribute should be used to define the hierarchical level elements within their hierarchical structures. Levels increase with depth. The value for aria-level is an integer greater than or equal to `1`.

When it comes to the headings in a document structure, you can have first-level headings, second-level headings, third-level headings, etc. In trees, you have the root element, its children, the children's children (or grandchildren), and so on.

The `aria-level` attribute exposes hierarchy to assistive technologies so it can be communicated to the users. Like all ARIA attributes, it has no impact on the user agent and therefore no impact on the user agent's determination of the document structure.

If the DOM ancestry accurately represents the level, the user agent can calculate the level of an item from the document structure, making `aria-level` not only redundant but a risk of creating misinformation. `aria-level` should really only be used to provide an explicit indication of level when it is not possible to calculate from the document structure. Test to see if this attribute is needed. If the user agent can calculate the level, it is best to omit the `aria-level` attribute.

# With `heading` role

The `aria-level` attribute is a required attribute of the `heading` role which indicates to assistive technologies that the element should be treated as a heading. `<div role="heading" aria-level="1">` defines the `<div>` as being the main heading of the page. A level 2 heading, defined with `aria-level="2"` would be the first subsection, a level 3 is a subsection of that, and so on.

```
<div role="heading" aria-level="3">Heading for this sub section</div>
```

Opt for using the h1 through h6 elements instead.

# Within `treegrid` role

In the case of a `treegrid`, `aria-level` is supported on elements with the role `row`, not elements with role `gridcell`. Rows act as leaf nodes within the vertical orientation of the grid. Gridcells are leaf nodes within the horizontal orientation of each row. `Aria-level` is not supported on cells within rows. So, in treegrids, the `aria-level` attribute is applied to the element with the role `row`.

If a complete set of available nodes is not present in the DOM due to dynamic loading as the user moves focus in or scrolls the tree, each node has `aria-level`, `aria-setsize`, and `aria-posinset` included.

# Values

- `<integer>`
  - : An integer greater than or equal to 1

# Associated interfaces

- Element.ariaLevel
  - : The `ariaLevel` property, part of the Element interface, reflects the value of the `aria-level` attribute.
- ElementInternals.ariaLevel
  - : The `ariaLevel` property, part of the ElementInternals interface, reflects the value of the `aria-level` attribute.

# Associated roles

Used in roles:

- `associationlistitemkey`
- `comment`
- `heading`
- `row`

# Specifications

# See also

- `<h1>` through `<h6>`: The HTML Section Heading elements (h1, h2, h3, h4, h5, and h6)

---

□□□ #3
□ □□□ □□□ 2026-10-31 03:27:53 UTC
□ cocobook-sync □□□ 2026-11-04 17:33:07 UTC