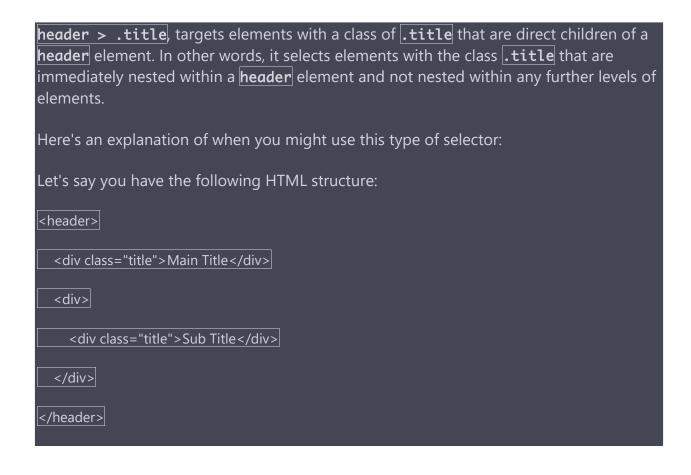Why we use Hr tag

The <hr /> tag in HTML stands for horizontal rule and is used to insert a horizontal rule or a thematic break in an HTML page to divide or separate document sections1. It is an empty tag, meaning that it only has an opening tag and does not require an end tag1. Starting in HTML5, we now need to attach a slash to the tag of an empty element. So, instead of having just <hr>, you should make it <hr />2. In browsers, the <hr /> tag is displayed as a horizontal rule or line2.

`<blockquote>` It is used to indicate that a section of text is a quotation from another source. Browsers typically render text within a `<blockquote>` element with indentation to visually set it apart from the surrounding content. It's commonly used for citing longer passages or quotes within an article or document.

In CSS, the asterisk `*` symbol is used as a universal selector. It selects and targets all elements within an HTML document. This means that any CSS rules you apply using the `*` selector will affect every element on the page unless more specific selectors override them.

`header > .title`, targets elements with a class of `.title` that are direct children of a `header` element. In other words, it selects elements with the class `.title` that are immediately nested within a `header` element and not nested within any further levels of elements.

Here's an explanation of when you might use this type of selector:

Let's say you have the following HTML structure:

`<header>`

`    <div class="title">Main Title</div>`

`    <div>`

`        <div class="title">Sub Title</div>`

`    </div>`

`</header>`

In this example, you have a `header` element that contains two elements with the class `.title`. One is a direct child of the `header`, and the other is nested within another `div` inside the `header`.

If you want to apply specific styling only to the `.title` element that is a direct child of the `header` (in this case, the "Main Title"), you would use the selector `header > .title`. This ensures that the styling only targets the immediate child element and not the nested one.

```
header > .title {

    margin-bottom: 1rem;

}
```

This CSS rule will add a bottom margin to the "Main Title" but will not affect the "Sub Title" since it is not a direct child of the `header` element.

In summary, the `header > .title` selector is useful when you want to target specific elements that are direct children of a particular parent element and not apply the styling to elements that are nested further down the hierarchy.

The CSS property `grid-template-columns` is used in conjunction with CSS Grid Layout to define the sizing and arrangement of columns within a grid container. It allows you to create a grid of columns with different widths and proportions.

In the example you provided, `grid-template-columns: 2fr 1fr;`, you are defining a grid with two columns. Here's a breakdown of what each part means:

1. `2fr`: This specifies the width of the first column. The unit `fr` stands for "fractional unit" and represents a portion of the available space in the grid container. In this case, the first column will take up twice as much available space as the second column. So, if the total available width is 100 units, the first column will take 2/3 (66.67 units), and the second column will take 1/3 (33.33 units).
2. `1fr`: This specifies the width of the second column. As mentioned above, `fr` represents a fraction of the available space. In this case, the second column will take up one-third of the available space.

Using `fr` units in `grid-template-columns` allows for flexible and responsive layouts where columns adjust their widths based on the available space while maintaining the specified proportions. This is particularly useful for creating responsive designs that adapt to different screen sizes and orientations.