

## Check in

So far in this discussion of CSS Grid we have set two properties: **display:grid** and **grid-template-columns: ...** . Again, I strongly recommend not setting rows until you are better able to predict how the content of the areas will be set.

## Grid Template Areas

The grid-template-areas property allows you name your grid, which makes it easier to set and structure content. The below example could be used for a grid set up to use 4 columns (like what is in the above discussion)

Ex.

Grid-template-areas:

```
"header header header header"  
"aside main main main"  
"footer footer footer footer";
```

By naming the areas it makes it easier for us to understand the grid, and makes it easier to visualize the grid before it is actually on the page.

**HINT:** You need to make sure the number of names in each row line up correctly with the number of columns, otherwise the browser will not understand how to interpret the value. So, if you have a grid set up to use 6 columns, but only name these 4, your grid will have issues displaying, likewise if you name 7 on one, 6 on one and 4 one 1, you will have bunch of issues with it rendering incorrectly.

You can then assign section to the area by setting grid-area on the element to take up that space in the grid. Let's say you have a >nav> item with an id of "navbar" and you can it to make up the header section in the grid.

```
<nav id="navbar"> ... </nav>
```