CSS Grid Essentials

**Review**

At this point, we've covered a great deal of different ways to manipulate the grid and the items inside it to create interesting layouts.

* grid-template-columns defines the number and sizes of the columns of the grid
* grid-template-rows defines the number and sizes of the rows of the grid
* grid-template is a shorthand for defining both grid-template-columns and grid-template-rows in one line
* grid-gap puts blank space between rows and/or columns of the grid
* grid-row-start and grid-row-end makes elements span certain rows of the grid
* grid-column-start and grid-column-end makes elements span certain columns of the grid
* grid-area is a shorthand for grid-row-start, grid-column-start, grid-row-end, and grid-column-end, all in one line

You have seen how to set up and fill in a grid and you now have one more CSS positioning technique to add to your toolkit! Let's do some practice to solidify these skills.

Advanced CSS Grid

**Review**

Great work! You have learned many new properties to use when creating a layout using CSS Grid! Let's review:

* grid-template-areas specifies grid named grid areas
* grid layouts are two-dimensional: they have a row, or inline, axis and a column, or block, axis.
* justify-items specifies how individual elements should spread across the row axis
* justify-content specifies how groups of elements should spread across the row axis
* justify-self specifies how a single element should position itself with respect to the row axis
* align-items specifies how individual elements should spread across the column axis
* align-content specifies how groups of elements should spread across the column axis
* align-self specifies how a single element should position itself with respect to the column axis
* grid-auto-rows specifies the height of rows added implicitly to the grid
* grid-auto-columns specifies the width of columns added implicitly to the grid
* grid-auto-flow specifies in which direction implicit elements should be created

This is a great time to experiment with the code in the code editor and try any of the properties you want to practice more! When you're ready, move on!