**CSS grids**

CSS Grid layouts are important

when it comes to designing a good webpage.

This is because layouts are

a way to provide visual cues for

a user by organizing

relevant content to make it easier to comprehend.

Let's learn more about layouts.

When someone says the word grid,

you probably think of lines that cross each

other to form squares or rectangles.

CSS Grids are two-dimensional design layouts that

are responsive and compatible with browser variations.

They are an alternative to

other options such as Flexboxes and tables,

especially when you are working

with larger scale layouts.

Columns are the vertical tracks and

rows are the horizontal tracks in your viewport.

Grids divide the page into rows and columns,

and the space between these tracks

are called gutters or gaps.

A cell is the space in

a grid container where a row and column intersect.

There are several different configurations you

can add to define and modify grids.

Let's examine an example that uses

a HTML document to

demonstrate how to iteratively make changes to a grid.

First, let's create a HTML document called index.html.

Right now, the contents displayed in the viewport are

just a vertical series of letters

from A to E without any styling.

Let's start adding content to the CSS file.

The first thing to do is set

the values for the properties of

the different box classes inside

the container object to

make the layout look more presentable.

The letters now have a better visual design,

but their arrangement on screen is unchanged.

The result is that each letter occupies

more screen space than is necessary for its size.

What's displayed in the viewport may appear to be a grid,

but it's not actually one.

It's just the default settings of CSS for a layout.

Once you convert this into the grid layout,

you'll be able to recognize the flexibility it can add.

Let's convert it into a grid layout.

Set the properties of the container class by

first assigning a grid value to the display property.

The display property is also used to set

display types for other designs such as flex,

block, inline, and so on.

It's usually a part of

container elements inside our code.

The updated container now

has the display property value set to

grid and the property values

configured for three columns and two rows.

That's two changes from

the previous configuration. Let's examine them.

The grid template columns property has been added into

the CSS code to set the size of

each of the three columns using pixel values.

The use of fr,

which is an abbreviation for

fraction, has been introduced.

Fraction effectively divides the grid.

The page now displays five separate grid cells

around the letters which are

arranged in three columns and two rows.

Where applied, the grid tracks are divided

proportionately to the ratio of

all the fraction values present.

As there are two rows,

the defined values are sized in the ratio of two to one.

Fraction adds flexibility to the grid

without needing to deal with actual pixel sizes.

It must be noted though,

that fraction and pixel sizes can be used

interchangeably with both rows and columns.

Now to add a couple of properties such as grid gap or

gutter and background color to define the grid,

the updated view displays a red box representing

the grid because that's what

the background colors RGB value was set to.

The grid stretches by default

the entire width of the page and

the size of the grid track or

gutter has been adjusted to 10 pixels,

which leaves more space between the grid cells.

You can also opt to use the auto properties such as grid,

auto rows and grid auto columns

collectively called the implicit grid.

Let's replace the grid template rows

property in the code with grid auto rows.

All the rows are now auto-resized to 100 pixels.

Now let's examine some of the functions

that can make configuring the layout easier.

First, the repeat function passes the number of

repeats required for a given number of rows and columns.

The result of the code adjustment

is an unchanged webpage from

the last instance because

the repeat function didn't change anything,

it just reduced the amount of code you need to write.

Therefore, the repeat function helps reduce

redundancy and provides ease of code modification.

Next, the min-max function sets

the minimum and maximum values

expected for the sizes of rows and columns.

Let's set the value for the grid auto rows to 150 pixels.

The output is modified so that

the minimum size of each row is 150 pixels.

One more important concept is grid frameworks.

There are a number of commonly

used grid frameworks and layouts.

Two such grid design layouts are

known as the 12 and 16-column grids.

They divide the page into 12 and 16 tracks respectively,

along the vertical columns.

The properties can be modified

accordingly to target a specific track.

CSS Grid layouts are so

important when it comes to good webpage design.

This video has demonstrated that layer

to provide visual cues for the user

by organizing content in a way that makes it

easier to understand. Who knows?

Soon you'll be building your own

web pages and you'll be able to

test out some of the techniques shown in this video.

Don't forget the repeat and min-max functions,

they'll save you time and effort.