Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {
  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
}

.ruled-grid > .card {
}

// Row Lines


// Column Lines

```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

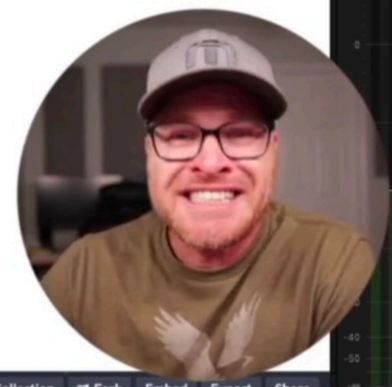
Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

We want lines inside the grid - nothing on the outside

Video



Console Assets Comments ⌘ Keys

Last saved SEPTEMBER 18, 2023 – 9:38:55 AM Delete Add to Collection Fork Embed Export Share

00:38

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
p {}
.container {}

// Ruled Grid

.ruled-grid {
  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
}

.ruled-grid > .card {
}

// Row Lines


// Column Lines

```

Our basic setup

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

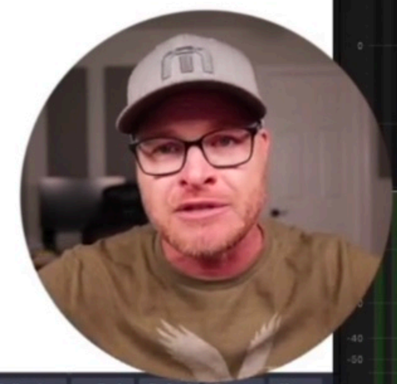
Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



JS

Console Assets Comments ⌘ Keys

Last saved SEPTEMBER 18, 2023 – 9:56:41 AM Delete Add to Collection ⌥ Fork Embed Export Share

05:30

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {
  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
}

.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
}

// Column Lines
```

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

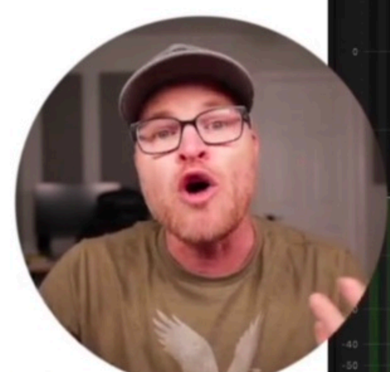
Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Our starter for the row lines

Because we're using absolutely positioned elements in the card



JS

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

08:00

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```scss
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {
  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
}

.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
  top: 0;
  left: 0;
}

// Column Lines
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

08:58

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

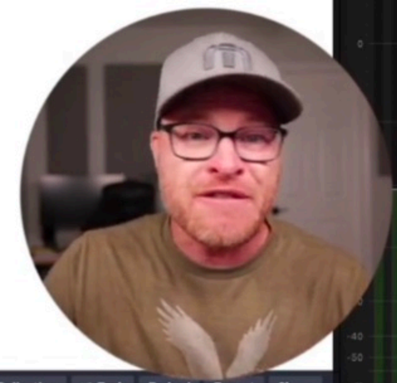
```
}

.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
  top: 0;
  left: 0;
}
```

```
// Column Lines

.ruled-grid > .card::before {
  content: "";
  background-color: red;
  position: absolute;
  width: 2px;
  height: 100%;
  top: 0;
  left: 0;
}
```

The basics for our column lines

Ruled Grid? Is it possible to create a ruled grid that's flexible, scalable, and maintainable?	**No Media Queries?** If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?	**Any Column Count?** It would be great if the technique would work regardless of the number of columns & rows!
Any # of Grid Items? If we have an uneven number of grid items in the grid, will it still work?	**Avoid :nth() selectors?** We must avoid :nth() selectors as they'll break column count flexibility.	**Works with any background?** Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



JS

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

10:06

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

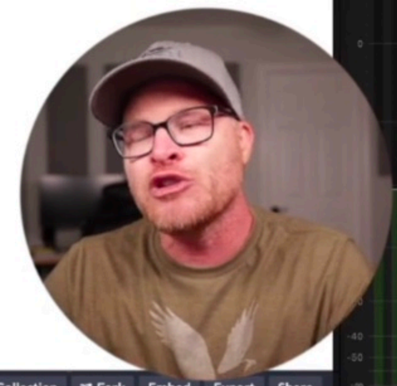
```
.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
  top: 0;
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  content: "";
  background-color: red;
  position: absolute;
  width: 2px;
  height: 100%;
  top: 0;
  left: 0;
}
```

JS

Ruled Grid?	No Media Queries?	Any Column Count?
		If the technique would work regardless of the number of columns & rows!
Any # of Grid Items?	**Avoid :nth() selectors?**	**Works with any background?**
If we have an uneven number of grid items in the grid, will it still work?	We must avoid :nth() selectors as they'll break column count flexibility.	Can't use background-color & gap technique because it's too limiting.

We don't want the left and top line to show

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

10:38

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {
  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
  gap: 2em;
}

.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
  top: 0;
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  content: "";
  background-color: red;
  position: absolute;
  width: 2px;
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

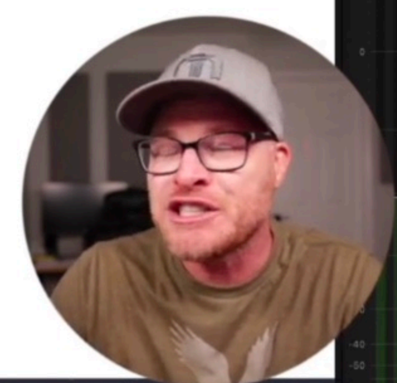
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

We need a gap to center the lines in the middle of the gaps



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

11:21

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```scss

.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
  top: 0;
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  content: "";
  background-color: red;
  position: absolute;
  width: 2px;
  height: 100%;
  top: 0;
  left: -1em;
}
```

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

1em is the half of our line thickness an
now the line is in the middle of the gap

JS

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

12:25

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genl.us InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```scss

.ruled-grid > .card {
  position: relative;
}

// Row Lines

.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
  width: 100%;
  height: 2px;
  top: -1em;
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  content: "";
  background-color: red;
  position: absolute;
  width: 2px;
  height: 100%;
  top: 0;
  left: -1em;
}
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

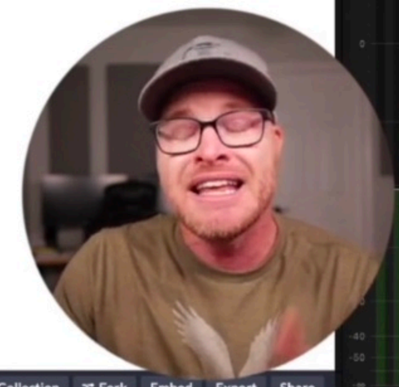
Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

12:42

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
  position: relative;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: red;
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100%;
  height: 2px;
  top: -1em;
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  width: 2px;
  height: 100%;
  top: 0;
  left: -1em;
}
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

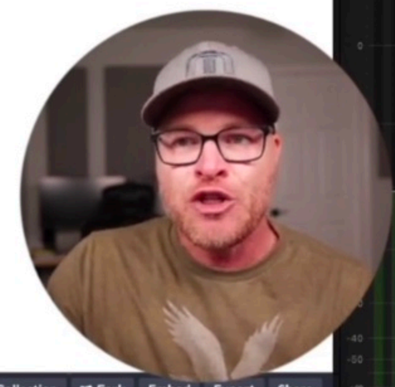
It would be great if the technique would work regardless of the number of columns & rows!

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force grid cells to be symmetrical?

These are the common styles for both pseudo elements and only the relevant information is left in the details



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

14:18

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {

  --gap: 3em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
  gap: var(--gap);
}

.ruled-grid > .card {
  position: relative;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100%;
```

JS

With these tokens we can manipulate the lines from one place

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

# of Grid Items?

e have an uneven number of grid s in the grid, will it still work?

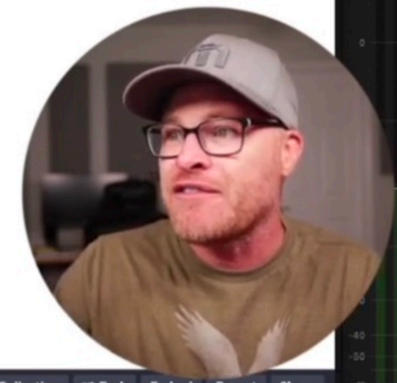
Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

17:52

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
.ruled-grid > .card::after {
  width: 100%;
  height: var(--line-thickness);
  top: var(--line-offset);
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  width: var(--line-thickness);
  height: 100%;
  top: 0;
  left: var(--line-offset);
}
```

Because these are now positive values they're bringing the lines in

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

JS

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

18:22

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
// Ruled Grid

.ruled-grid {

  --gap: 5em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
  gap: var(--gap);
}

.ruled-grid > .card {
  position: relative;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100%;
  height: var(--line-thickness);
  top: var(--line-offset);
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

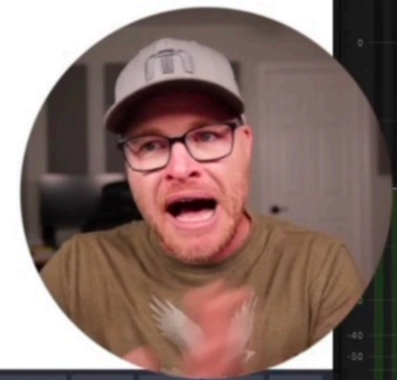
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Best practice is not to assign a negative value to a token but making it negative in the place where it needs to be!



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

18:49

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
// Row Lines

.ruled-grid > .card::after {
  width: 100%;
  height: var(--line-thickness);
  top: calc(var(--line-offset) * -1);
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  width: var(--line-thickness);
  height: 100%;
  top: 0;
  left: calc(var(--line-offset) * -1);
}
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

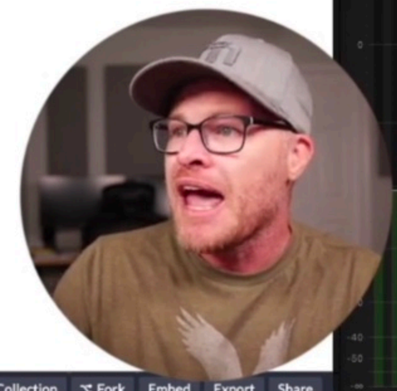
Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



Console Assets Comments ⌘ Keys

Building Pen preview Delete Add to Collection Fork Embed Export Share

19:10

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```scss
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
  gap: var(--gap);
}

.ruled-grid > .card {
  position: relative;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100%;
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

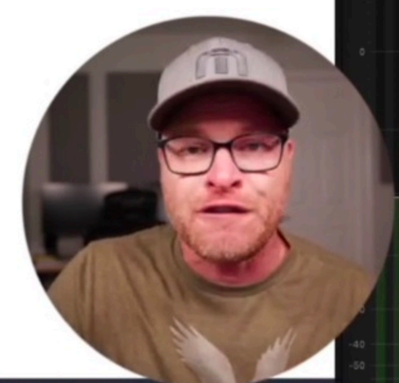
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Now we can change the gap and the lines are always centered



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

19:21

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100%;
  height: var(--line-thickness);
  top: calc(var(--line-offset) * -1);
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  width: var(--line-thickness);
  height: 100%;
  top: 0;
  left: calc(var(--line-offset) * -1);
}
```

100% in this case is the size of the card and the gaps will never be filled

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

cells have an uneven amount of content, can we force all grid cells to be symmetrical?

JS

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

20:27

Kevin Geary - How to Create a “Ruled” Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100vw;
  height: var(--line-thickness);
  top: calc(var(--line-offset) * -1);
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  width: var(--line-thickness);
  height: 100%;
  top: 0;
  left: calc(var(--line-offset) * -1);
}
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

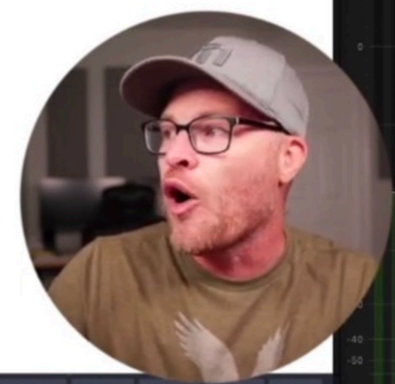
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

rid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

The solution to this is ‘vw’



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

20:43

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100vw;
  height: var(--line-thickness);
  top: calc(var(--line-offset) * -1);
  left: 0;
}

// Column Lines

.ruled-grid > .card::before {
  width: var(--line-thickness);
  height: 100vh;
  top: 0;
  left: calc(var(--line-offset) * -1);
}
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Look at the middle of the grid: That's exactly what we want

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

20:55

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100vw;
  height: var(--line-thickness);
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

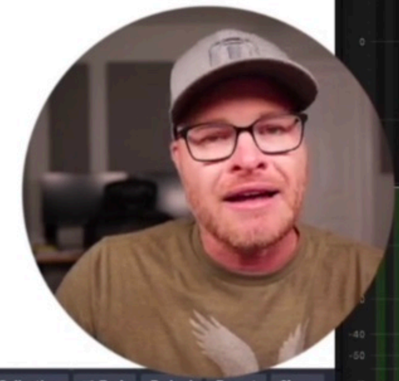
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

The magic trick



Console Assets Comments ⌘ Keys

Pen Preview finished Delete Add to Collection ⌥ Fork Embed Export Share

21:57

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(3, minmax(0, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
  padding: 1.5em;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  width: 100vw;
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Additional padding doesn't break the layout

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

22:18

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

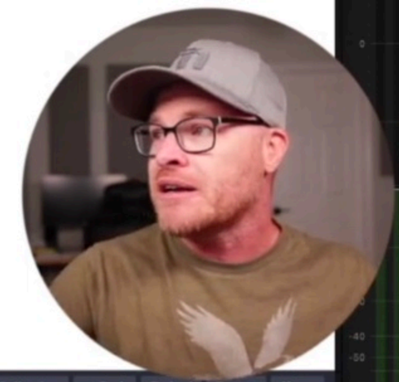
```
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(0, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
  padding: 1.5em;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
```

Doesn't break

JS

Ruled Grid?	No Media Queries?	Any Column Count?	Any # of Grid Items?	Avoid :nth() selectors?	Works with any background
Is it possible to create a ruled grid that's flexible, scalable, and maintainable?	If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?	It would be great if the technique would work regardless of the number of columns & rows!	If we have an uneven number of grid items in the grid, will it still work?	We must avoid :nth() selectors as they'll break column count flexibility.	Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

22:41

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit,
minmax(200px, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
  padding: 1.5em;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

23:01

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
work regardless of the number of columns & rows!</p>
```

CSS (SCSS)

```
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit,
minmax(320px, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
  padding: 1.5em;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

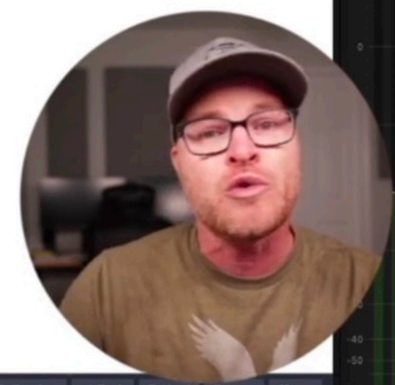
Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?



Console Assets Comments ⌘ Keys

Last saved SEPTEMBER 18, 2023 – 10:16:21 AM Delete Add to Collection Fork Embed Export Share

23:14

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Dimensions: Responsive 1670 x 665 100% Custom

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

CSS (SCSS)

```scss
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit,
minmax(280px, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}
```

JS

Console Assets Comments ⌘ Keys

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

Works with any background?

Fork Embed Export Share

We don't need any media queries

Elements Console Sources Network Performance Memory Application Security Lighthouse Animations 49 1

```
loading="lazy" spellcheck="false">
▼#document
  <!DOCTYPE html>
▶<html lang="en" class> … </html> == $0
</iframe>
▶<grammarly-extension data-grammarly-shadow-root="true" style="position: absolute; top: 0px; left: 0px; po
```

Styles Computed Layou

Filter

```
element.style {
}
html {
```

html.mac.chrome116.js body.room-editor.editor.state-htmlOn-cssOn-jsOn.twilight.layout-left.layout-side.logged-in div.page-wrap div

Console What's New Issues

23:33

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Dimensions: Responsive 1235 x 665 100% Custom

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

CSS (SCSS)

```scss
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit,
minmax(280px, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}
```

JS

Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Elements Console Sources Network Performance Memory Application Security Lighthouse Animations 49 1

```
loading="lazy" spellcheck="false">
  ▼#document
    <!DOCTYPE html>
  ▶<html lang="en" class> … </html> == $0
</iframe>
▶<grammarly-extension data-grammarly-shadow-root="true" style="position: absolute; top: 0px; left: 0px; po
```

Styles Computed Layou

Filter

```
element.style {
}
html {
```

html.mac.chrome116.js body.room-editor.editor.state-htmlOn-cssOn-jsOn.twilight.layout-left.layout-side.logged-in div.page-wrap div

Console What's New Issues

23:39

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Genius InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

```
      <p>If we have an uneven number of grid items in the grid, will it still work?</p>
    </div>
    <div class="card">
      <h3>Avoid :nth() selectors?</h3>
      <p>We must avoid :nth() selectors as they'll break column count flexibility.</p>
    </div>
<!--      <div class="card">
      <h3>Works with any background?</h3>
      <p>Can't use background-color & gap technique because it's too limiting.</p>
    </div> -->
  </div>
</div>
<div class="container">
  <p style="text-align: center; max-width: 30rem; margin-inline: auto;"><strong>Also!</strong> If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?</p>
</div>
```

CSS (SCSS)

```
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

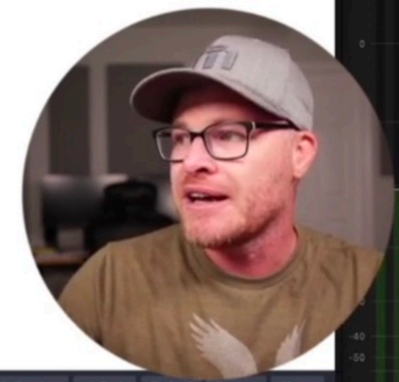
If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Also! If grid cells have an uneven amount of content, can we force

Even if we take a card away it doesn't break



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection Fork Embed Export Share

23:57

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=0100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

CSS (SCSS)

```scss
p {∞}
.container {∞}

body {
  background: linear-gradient(#ddd, #aaa);
}

// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(280px, 1fr));
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
  padding: 1.5em;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
```

```
abc e
  lute
activeborder
additive
activecaption
afar
after-white-space
ahead
alias
all
all-scroll
alphabetic
alternate
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

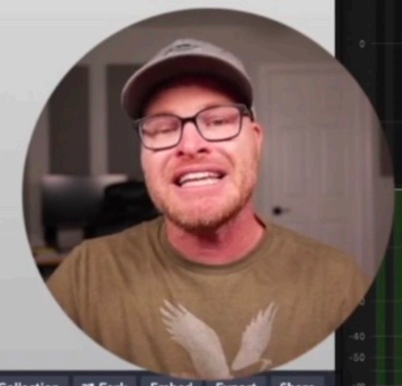
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Works with background gradient or color as well



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

24:44

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=1100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

```
<div class="ruled-grid">
  <div class="card">
    <h3>Ruled Grid?</h3>
    <p>Is it possible to create a ruled grid that's flexible, scalable, and maintainable? Is it possible to create a ruled grid that's flexible, scalable, and maintainable?</p>
  </div>
  <div class="card">
    <h3>No Media Queries?</h3>
    <p>If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?</p>
  </div>
  <div class="card">
    <h3>Any Column Count?</h3>
    <p>It would be great if the technique would work regardless of the number of columns & rows!</p>
  </div>
  <div class="card">
    <h3>Any # of Grid Items?</h3>
    <p>If we have an uneven number of grid items in the grid, will it still work?</p>
  </div>
  <div class="card">
    <h3>Avoid :nth() selectors?</h3>
    <p>We must avoid :nth() selectors as they'll break column count flexibility.</p>
```

CSS (SCSS)

```
p {∞}
.container {∞}

// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable? Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

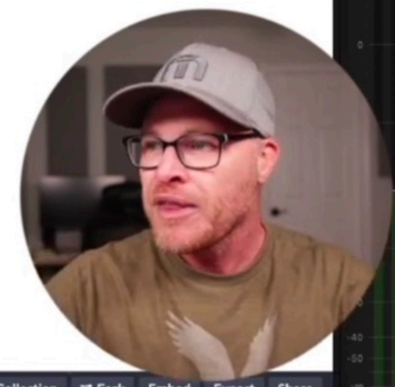
We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Problem: If 1 card has more content the grid is off balance



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

25:58

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=0100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

CSS (SCSS)

```scss
// Ruled Grid

.ruled-grid {

  --gap: 2em;
  --line-thickness: 2px;
  --line-color: red;
  --line-offset: calc(var(--gap) / 2);

  display: grid;
  grid-template-columns: repeat(auto-fit,
minmax(280px, 1fr));
  grid-auto-rows: minmax(min-content, 1fr);
  align-items: stretch;
  gap: var(--gap);
  overflow: hidden;
}

.ruled-grid > .card {
  position: relative;
  padding: 1.5em;
}

// Pseudo Elements

.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable? Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

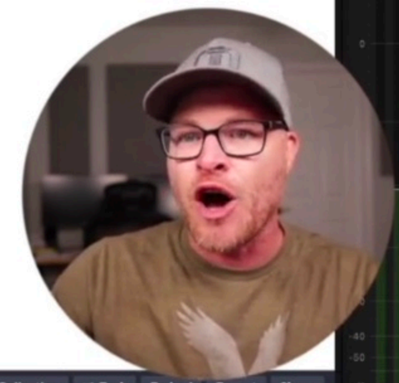
Works with any background?

Can't use background-color & gap technique because it's too limiting.

The solution

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

It takes the largest content and applies the size to every item in the grid



Console Assets Comments ⌘ Keys

Last saved LESS THAN A MINUTE AGO Delete Add to Collection ⌥ Fork Embed Export Share

26:25

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=0100

Update

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Ruled Grid
Kevin Geary PRO

Pen saved.

Save Settings

HTML

CSS (SCSS)

```
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  inline-size: 100vw;
  block-size: var(--line-thickness);
  inset-block-start: calc(var(--line-offset) * -1);
  inset-inline-start: 0;
}

// Column Lines

.ruled-grid > .card::before {
  inline-size: var(--line-thickness);
  block-size: 100vh;
  inset-block-start: 0;
  inset-inline-start: calc(var(--line-offset) * -1);
} inset-inline-start
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable? Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

We converted everything to logical properties

Also! If grid cells have an uneven amount of content, can we force all grid cells to be symmetrical?

Console Assets Comments ⌘ Keys

Processing code Delete Add to Collection ⌥ Fork Embed Export Share

29:19

Kevin Geary - How to Create a "Ruled" Grid (Lines / Grid Borders) in CSS the Right Way

Ruled Grid

codepen.io/Kevin-Geary/pen/BavwqYX?editors=0100

Facebook Hey! Business Personal Hosting Resources StreamYard Basecamp Automatic.css Frames Hubspot HelpScout Geni.us InnerCircle ACSS UserMaven JS Softball

Update

Ruled Grid
Kevin Geary PRO

Save Settings

HTML

CSS (SCSS)

```scss
.ruled-grid > .card::before,
.ruled-grid > .card::after {
  content: "";
  background-color: var(--line-color);
  position: absolute;
}

// Row Lines

.ruled-grid > .card::after {
  inline-size: 100vw;
  block-size: var(--line-thickness);
  inset-block-start: calc(var(--line-offset) * -1);
  inset-inline-start: 0;
}

// Column Lines

.ruled-grid > .card::before {
  inline-size: var(--line-thickness);
  block-size: 100vh;
  inset-block-start: 0;
  inset-inline-start: calc(var(--line-offset) * -1);
}
```

JS

Ruled Grid?

Is it possible to create a ruled grid that's flexible, scalable, and maintainable? Is it possible to create a ruled grid that's flexible, scalable, and maintainable?

No Media Queries?

If possible, can we avoid media queries? Can it work flawlessly at any breakpoint?

Any Column Count?

It would be great if the technique would work regardless of the number of columns & rows!

Any # of Grid Items?

If we have an uneven number of grid items in the grid, will it still work?

Avoid :nth() selectors?

We must avoid :nth() selectors as they'll break column count flexibility.

Works with any background?

Can't use background-color & gap technique because it's too limiting.

Also! If grid cells have an uneven amount of content, can we force

This solution works in any professional page builder (so not in Elementor for example because of the bloat and number of divs you can't control).

It's a perfect solution for Bricks Builder

Console Assets Comments ⌘ Keys

Last saved SEPTEMBER 18, 2023 – 10:22:49 AM Delete Add to Collection Fork Embed Export Share

30:38