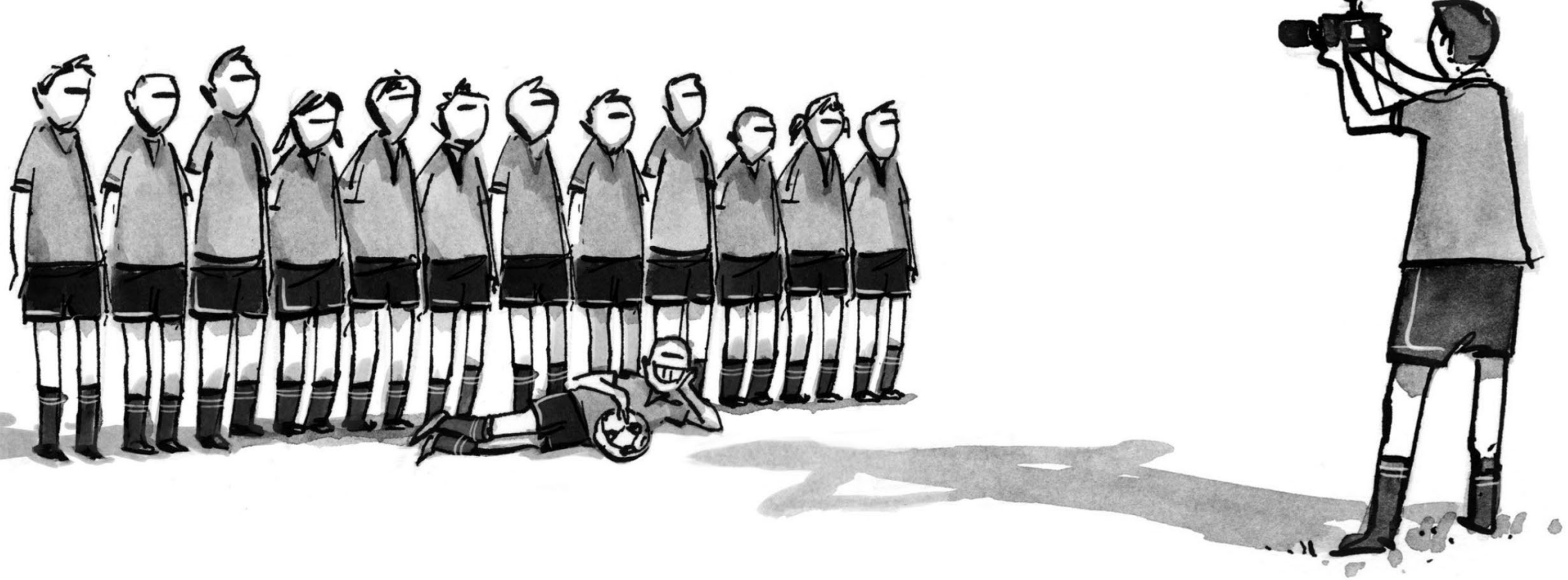
The Evolution of Responsive Design

RACHEL ANDREW AT FRONT END NORTH 2020

How do we support all these different screen sizes?



Issue №
306

Responsive Web Design

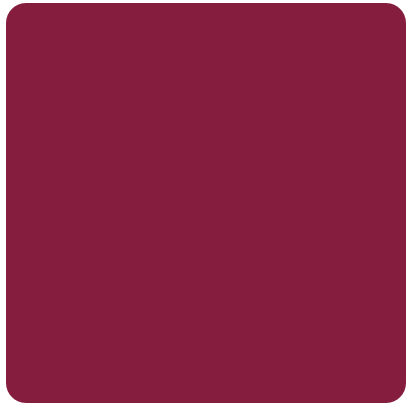
by **Ethan Marcotte** · May 25, 2010

Published in CSS, Interaction Design, Layout & Grids, Mobile/Multidevice, Responsive Design

> **The control which designers know in the print medium, and often desire in the web medium, is simply a function of the limitation of the printed page. We should embrace the fact that the web doesn't have the same constraints, and design for this flexibility. But first, we must 'accept the ebb and flow of things.'**

Fluid Grids

https://alistapart.com/article/fluidgrids/



Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke. Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

target ÷ context = result

Flexible Images

https://alistapart.com/article/fluid-images/

max-width: 100%

Media Queries

How big is this viewport?

Responsive Web Design

- Calculating percentages for floated layouts
- Squishing images
- Leaning on Media Queries

column-count: 3

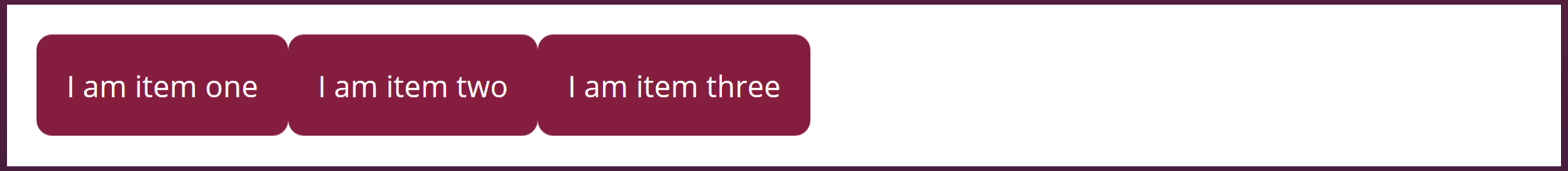
Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

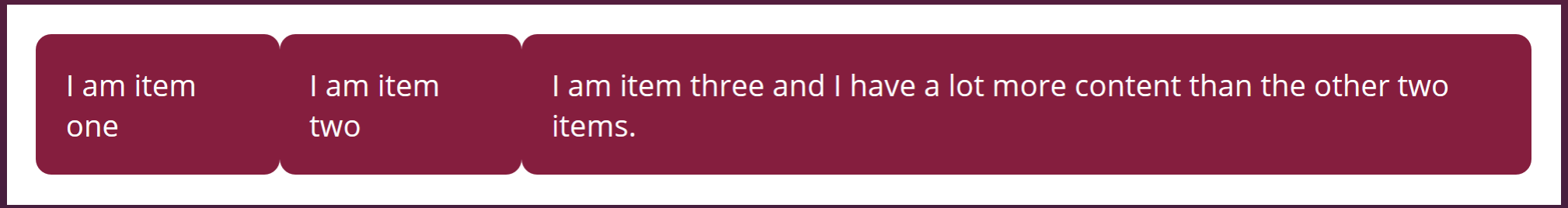
Celery quandong swiss chard chicory earthnut pea potato. Salsify taro catsear garlic gram celery bitterleaf wattle seed collard greens nori. Grape wattle seed kombu beetroot horseradish carrot squash brussels sprout chard.

Flexbox responds to the content

display: flex

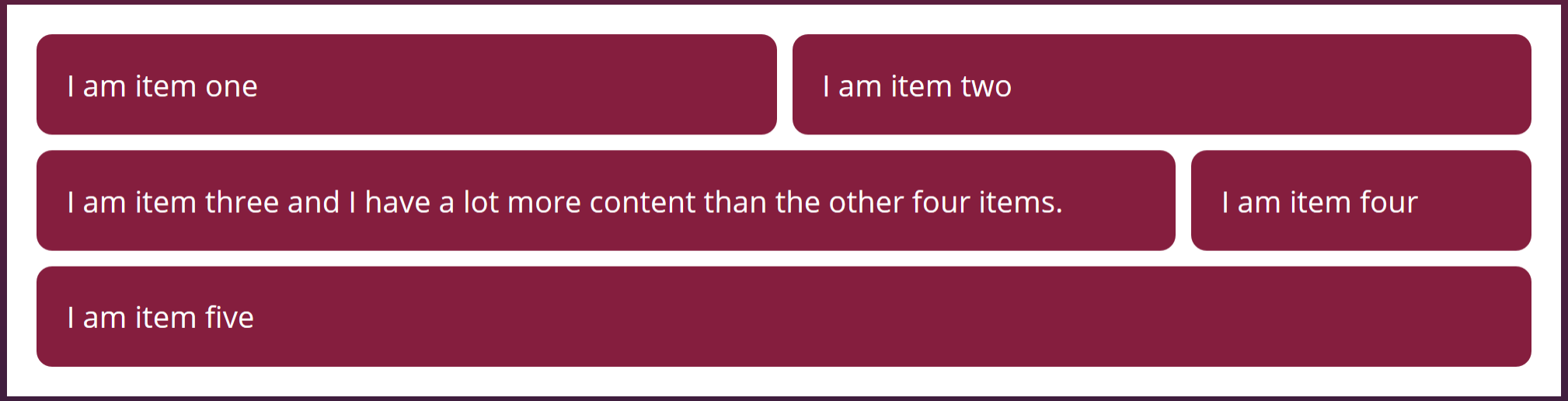


display: flex



Flexbox is one-dimensional

flex-wrap: wrap



flex: 0 0 32%

I am item one

I am item two

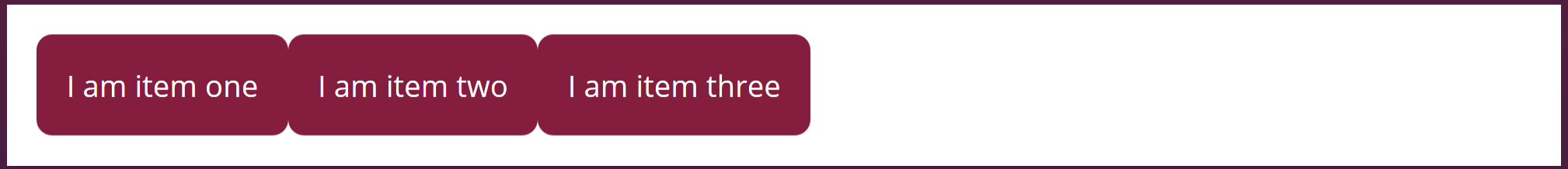
I am item three and I have a lot more content than the other four items.

I am item four

I am item five

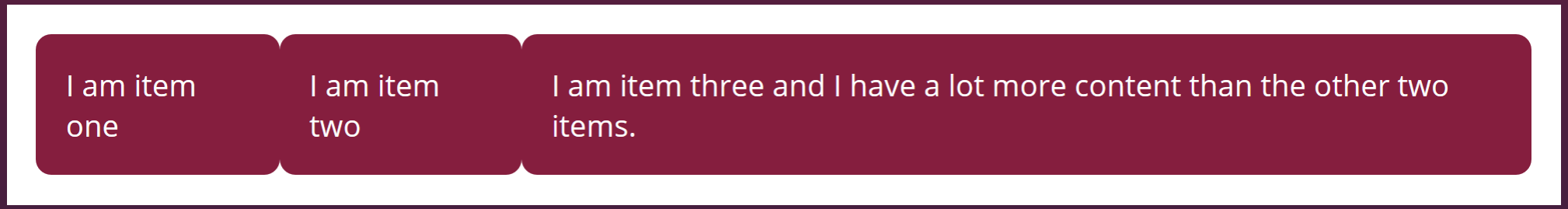
What does it mean to respond to content?

flex-basis: auto



max-content

flex-basis: auto



Space reduced from max-content until all boxes fit

We don’t have to line everything up.

Sometimes what we need is just to make things fit nicely.

CSS Grid

For when you do want to line everything up.

```css
.grid {
    display: grid;
    grid-template-columns: repeat(12, 1fr);
}
```

repeat(12, 1fr)

```
.grid {
    display: grid;
    grid-template-columns:
     repeat(12, minmax(auto, 1fr));
}
```

repeat(12, 1fr)

```
.grid {
    display: grid;
    grid-template-columns:
     repeat(12, minmax(0, 1fr));
}
```

repeat(12, minmax(0,1fr))

Responding to content in a grid layout world

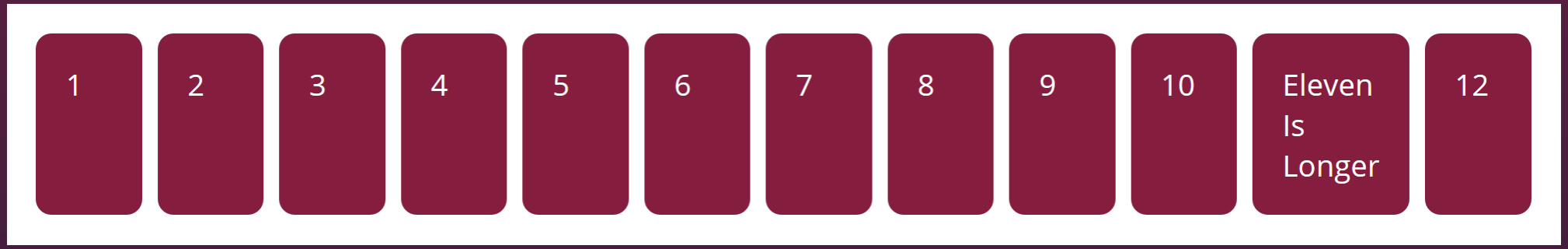
Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke. Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

auto

grid-template-columns: repeat(3, auto)

justify-content: start

repeat(12, 1fr)



flex-basis: auto

I am item one

I am item two

I am item three and I have a lot more content than the other two items.

repeat(3, 1fr)

I am item one

I am item two

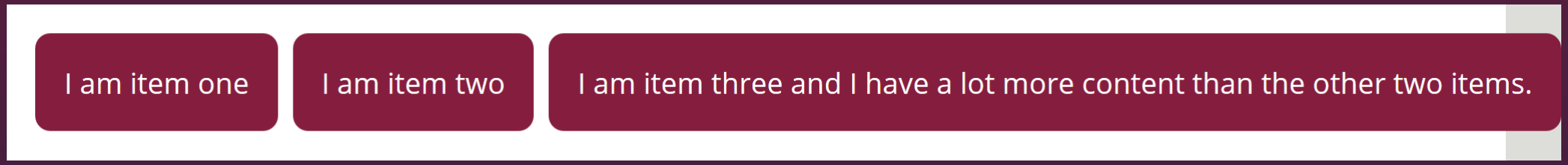
I am item three and I have a lot more content than the other two items.

min-content, max-content, fit-content

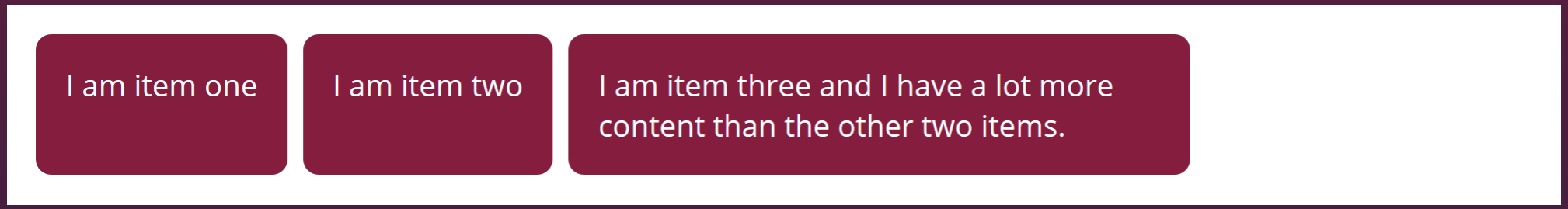
Directing content-based sizing

repeat(3, min-content)

repeat(3, max-content)



repeat(3, fit-content(400px))



max-content

400px

```css
.media {
    display: grid;
    grid-template-columns: fit-content(300px) 1fr;
    gap: 20px;
}
```

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea.

Flexible images

More than just max-width: 100%

Responsive Images

Solving the janky images problem

Mapping HTML attributes width and height to an aspect ratio.

It’s time to start using the width and height attributes on the <img> element again

Mapping the width and height attributes of media container elements to their aspect-ratio

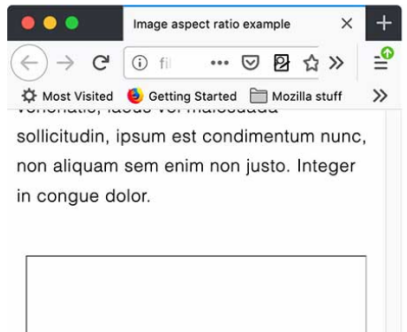
On this Page

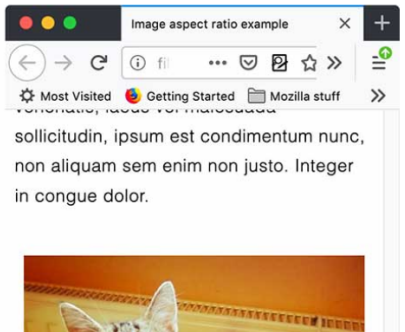
This article explains a change that has occured in the way sizes are worked out on web documents when width and height attributes are set on them.

This change means that the aspect ratio of the image is calculated by the browser early on and can then be used to correct the size needed to display the image before it has loaded, if CSS has been applied that causes problems with its display size. Read on to find out more.

Jank problems when loading images

In the olden days of web development, it was always seen as a good practice to add `width` and `height` attributes to your HTML `<img>` elements, so that when browsers first loaded the page, they could put a correctly-sized placeholder box in the layout for each image to appear in when it finally loads.





Media Queries

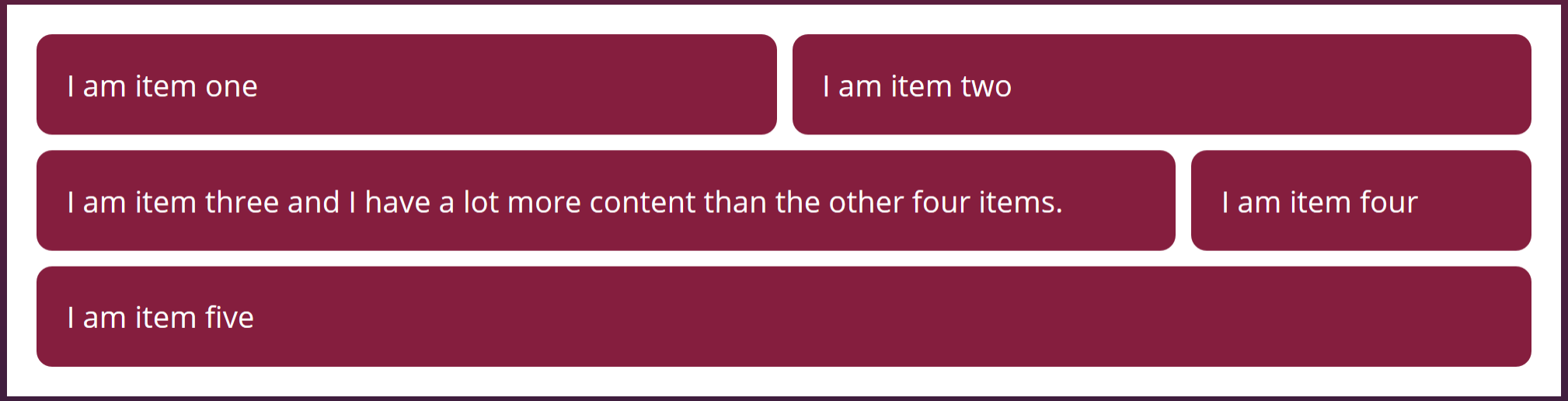
column-width: 200px

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko. Lotus root water spinach fennel kombu maize bamboo shoot green bean swiss chard seakale pumpkin onion chickpea gram corn pea. Brussels sprout coriander water chestnut gourd swiss chard wakame kohlrabi beetroot carrot watercress. Corn amaranth salsify bunya nuts nori azuki bean chickweed potato bell pepper artichoke.

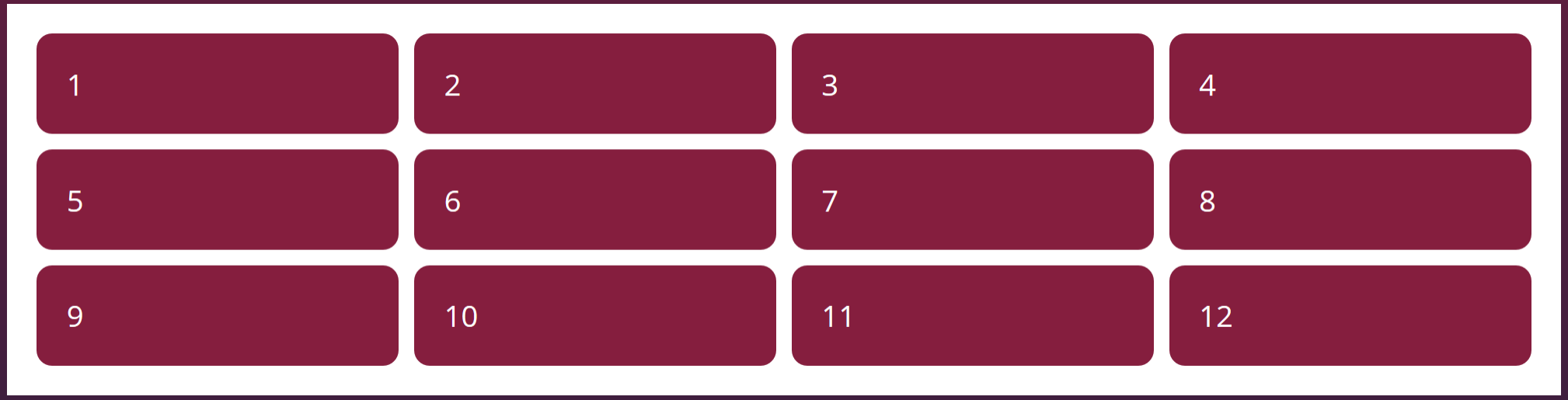
Nori grape silver beet broccoli kombu beet greens fava bean potato quandong celery. Bunya nuts black-eyed pea prairie turnip leek lentil turnip greens parsnip. Sea lettuce lettuce water chestnut eggplant winter purslane fennel azuki bean earthnut pea sierra leone bologi leek soko chicory celtuce parsley jícama salsify.

Celery quandong swiss chard chicory earthnut pea potato. Salsify taro catsear garlic gram celery bitterleaf wattle seed collard greens nori. Grape wattle seed kombu beetroot horseradish carrot squash brussels sprout chard.

flex-wrap: wrap



repeat(auto-fill, minmax(200px, 1fr)



The first rule of Media Queries is ...

Do I need a media query?

```
.wrapper {
    display: grid;
    grid-template-areas:
      "hd"
      "bd"
      "sd"
      "ft";
  }
```

The header.

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko.

The sidebar

The footer.

```
@media (min-width: 600px) {
  .wrapper {
    grid-template-columns: 3fr 1fr;
    grid-template-areas:
      "hd hd"
      "bd sd"
      "ft ft";
  }
}
```

The header.

Turnip greens yarrow ricebean rutabaga endive cauliflower sea lettuce kohlrabi amaranth water spinach avocado daikon napa cabbage asparagus winter purslane kale. Celery potato scallion desert raisin horseradish spinach carrot soko.

The sidebar

The footer.

More than just screen size

What else could we respond to?

What type of pointer do I have?

```
@media (pointer:coarse) {
  .pointer::before {
    content: "You have a coarse pointer";
  }
}
@media (pointer:fine) {
  .pointer::before {
    content: "You have a fine pointer";
  }
}
```

Can I hover?

```
@media (hover) {
    .can-i-hover::after {
      content: "You look like you can hover.";
    }
}

@media  (hover:none) {
    .can-i-hover::after {
      content: "I don't think you can hover.";
    }
}
```

```
@media (any-pointer:coarse) {
    .any-pointer::before {
      content: "One of your pointers is coarse. Yo
ur device has a touchscreen, though you might not
be using it.";


    }
}
```

You have a fine pointer, are you using a mouse or trackpad?

One of your pointers is coarse. Your device has a touchscreen, though you might not be using it.

> “Designing a page that relies on hovering or accurate pointing only because any-hover or any-pointer indicate that at least one of the available input mechanisms has these capabilities is likely to result in a poor experience.”

Media Queries Level 4

https://drafts.csswg.org/mediaqueries-4/#any-input

Testing capabilities

Check for screen size, but also pointer type and hover ability to understand the environment your site is being used in.

Overflow detection

Display Quality Media Queries

```
@media (overflow-block: paged) {
    /* CSS for paged media */
}
```

Responding to the wishes of the visitor

Media Queries Level 5 and User Preference Media Features

prefers-reduced-motion

```
@media (prefers-reduced-motion: reduce) {
    .animation {
      animation: none;
    }
}
```

prefers-color-scheme

```
@media (prefers-color-scheme: dark) {
  body {
    background: #333;
    color: white;
  }
}
```

prefers-reduced-data

```
@media (prefers-reduced-data: reduce) {
    /* CSS loading in smaller images */
}
```

Feature Queries

What does this browser support?

```
@supports (display: grid) {
    .grid {
        display: grid;
        grid-template-columns: repeat(3, 1fr);
    }
}
```

What does it mean to do responsive design today?

Respond to content

Use new layout and image features

Use media features to enhance your site.

It’s not about screen size

Responding to needs and environment

Stop expecting people to fix something to use your website. Respond to meet them where they are.

Thank you!

https://noti.st/rachelandrew