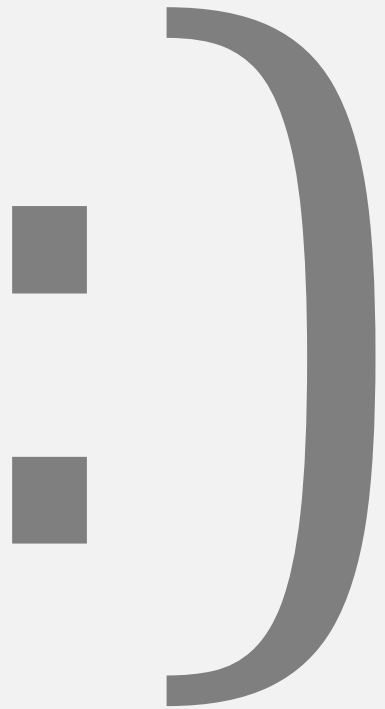


□ □ (





# RESPONSIVE WEB DESIGN

FORM FUNCTION CLASS

Web Design Conference | 10 NOV 2012



JUN SANTOS

Web Developer, Wunderman  
[jun.santos@outlook.com](mailto:jun.santos@outlook.com)

# In this workshop...

## Flexible Grid

Creating liquid web page layouts

## Flexible Media

Making fixed-width elements flow with the grid

## Media Queries

Altering page layout and features dynamically





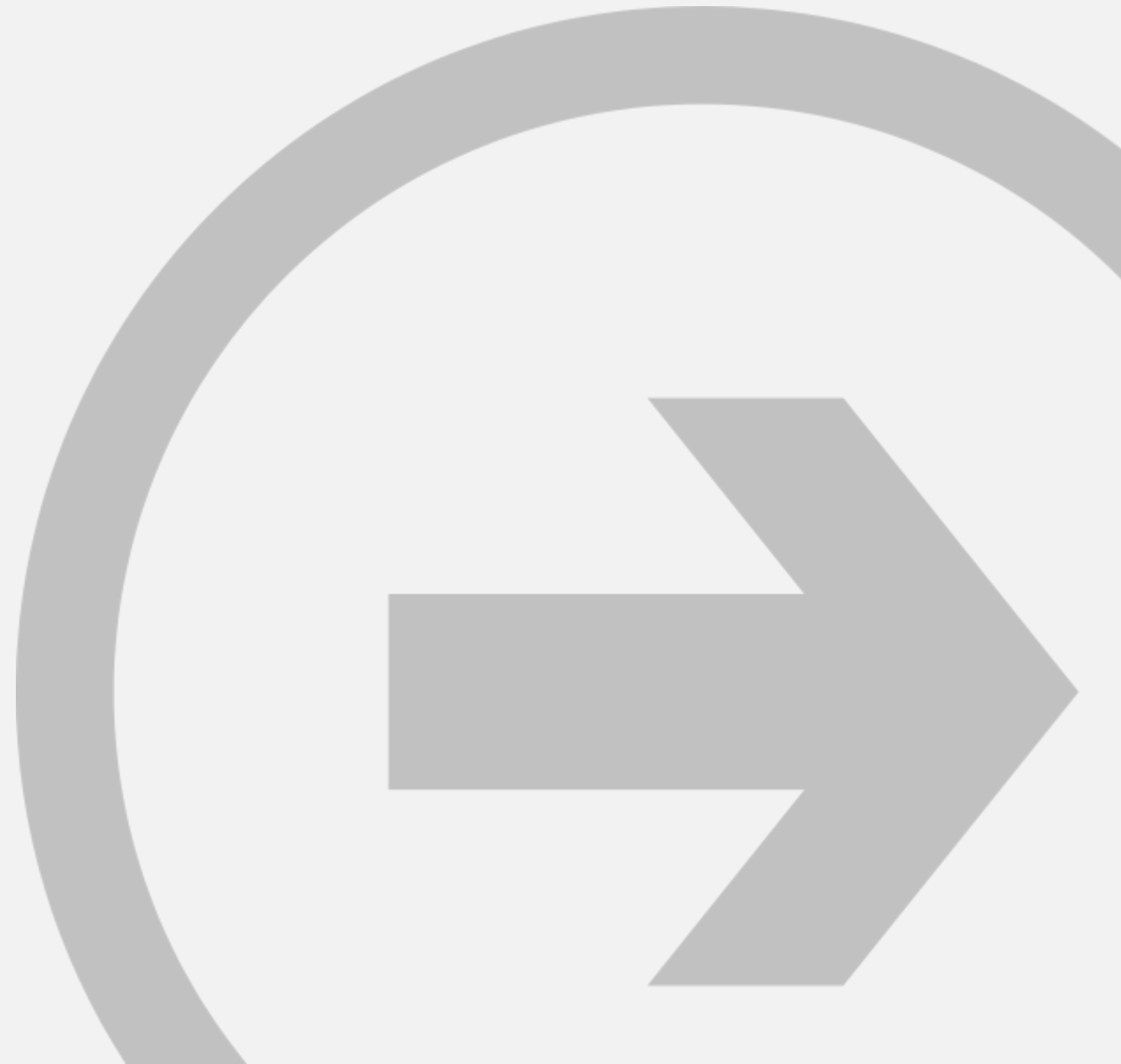
# In this section...

## Liquid Layout

Breaking free from conventional fixed-width designs

## Gutters

On margins and paddings





999999px; }

% equivalent of  
desired viewport  
space

$$\text{target} \div \text{context} = \text{result}$$

The desired viewport  
space to allocate to a  
page element (in px)

Measurement of another page element  
(usu. parent) to which target width will be  
expressed relative to

target / context = result

$$\begin{array}{r} 300 \\ \div 1200 \\ \hline 0.25 \end{array}$$

25 %



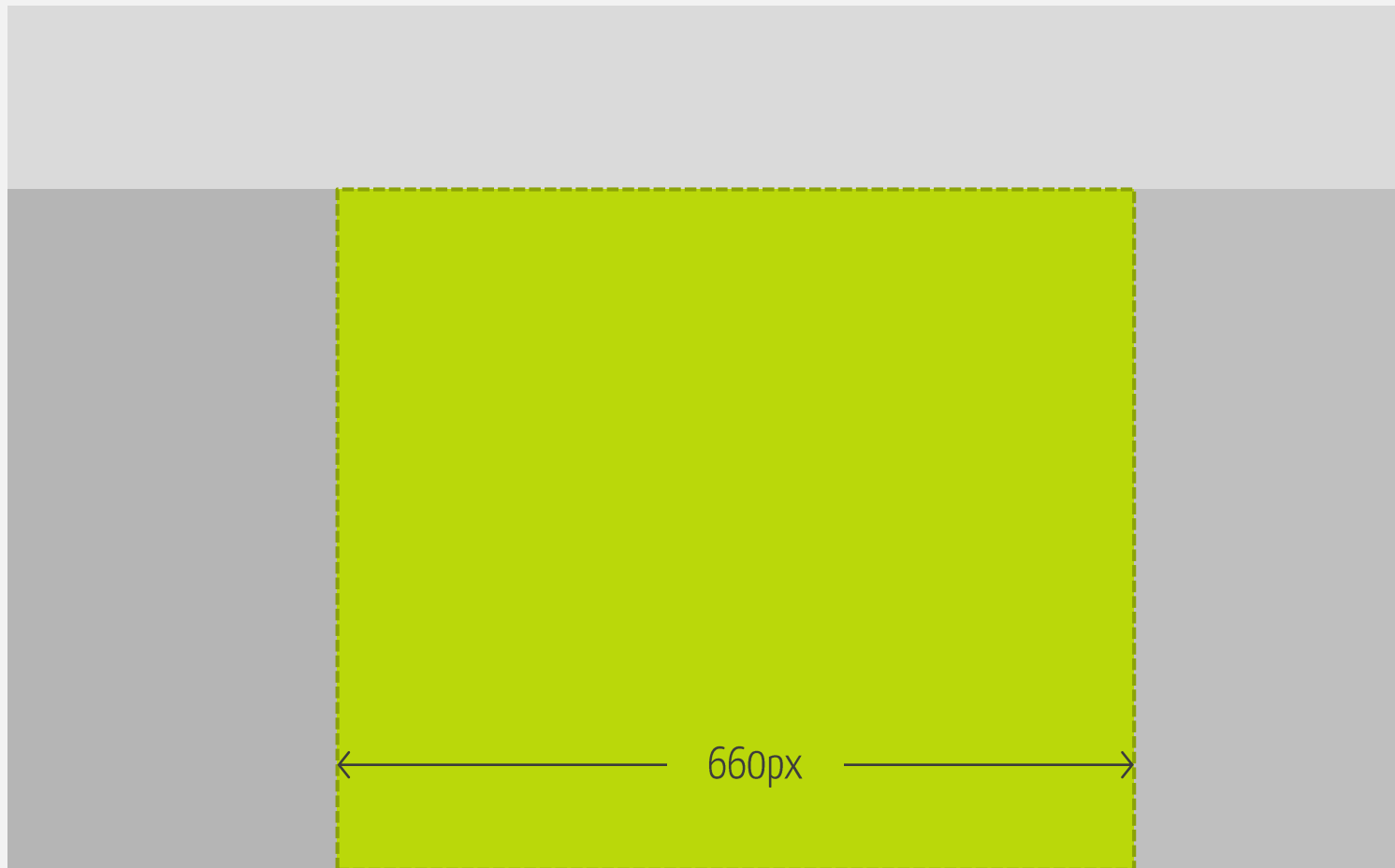
← 300px →

← 1200px →

target / context = result

$$\begin{array}{r} 660 \\ \div \\ 1200 \\ \hline 0.55 \end{array}$$

55 %



← 1200px →

← 660px →

target / context = result

$$\begin{array}{r} 240 \\ \div \\ 1200 \\ \hline 0.20 \end{array}$$

20 %



← 1200px →

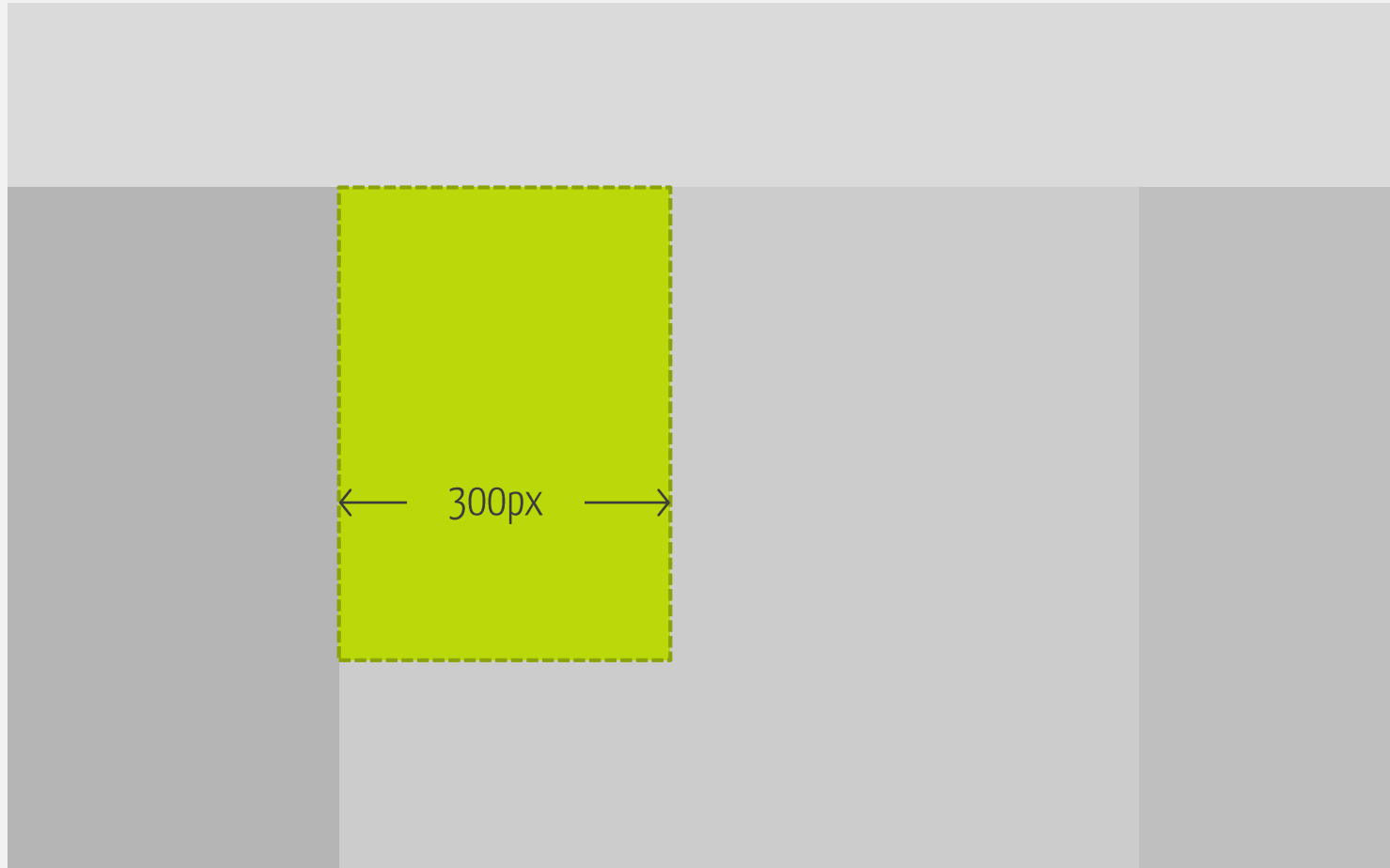
← 240px →

target / context = result

300  
÷  
660

0.454545

45.45 %



660px



## CSS box model

In the CSS box model, paddings, border, and margins are added outside the coded width

Width	660
Padding	20
Border	0
Margin	20

---

Outer Width 700

excess of 40px

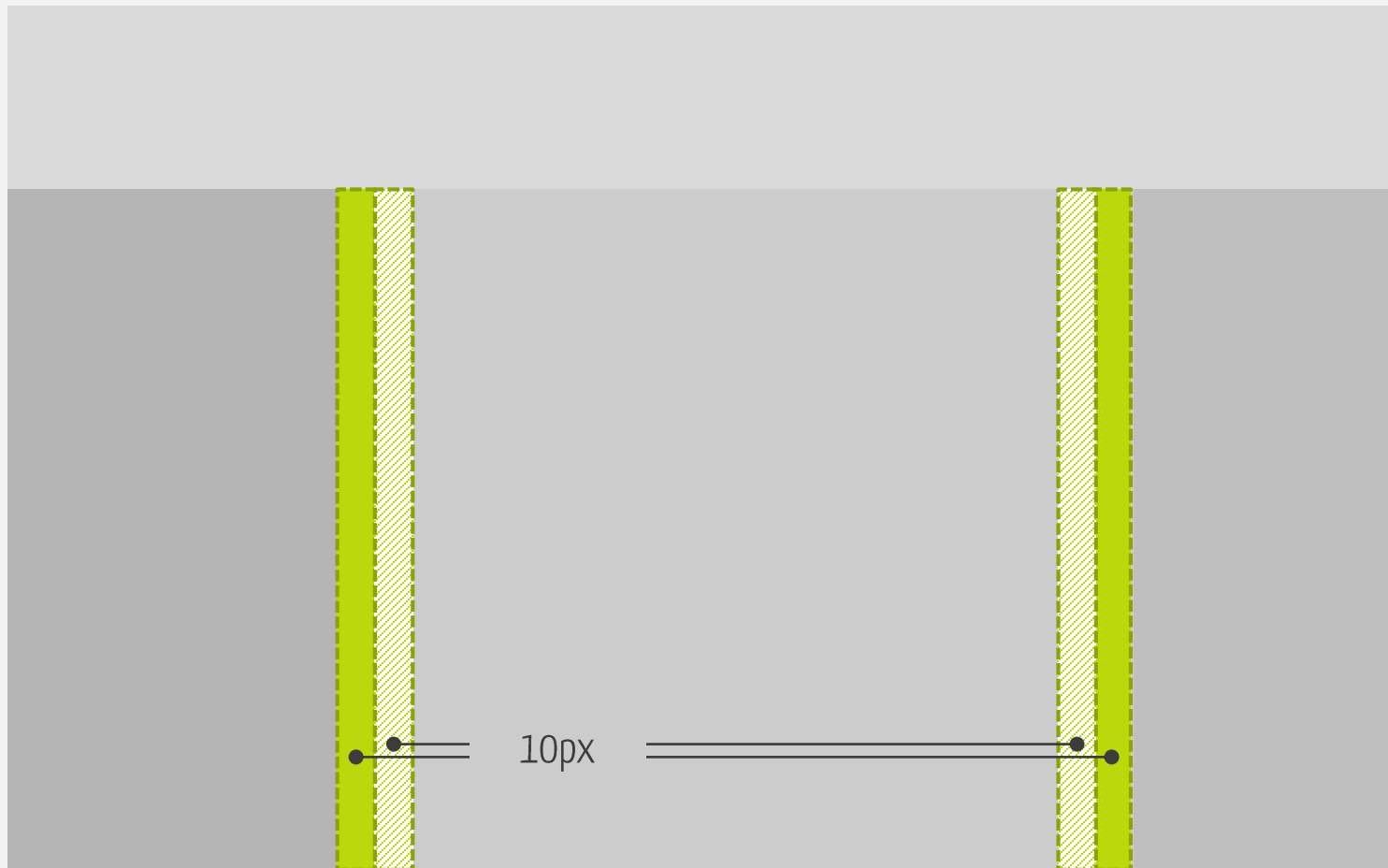


target / context = result

$$\begin{array}{r} 10 \\ \div \\ 1200 \end{array}$$

0.008333

0.083 %

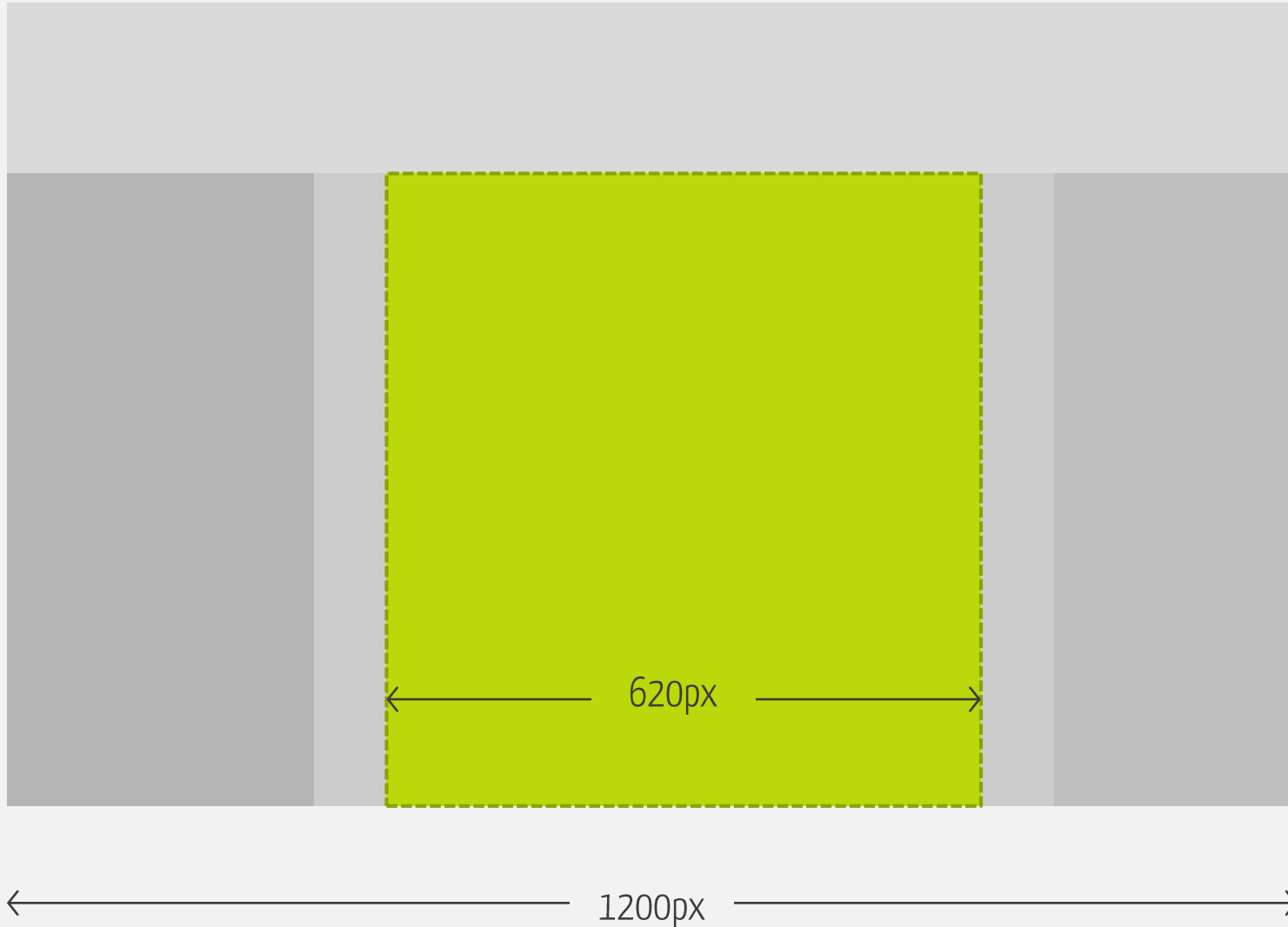


← 1200px →

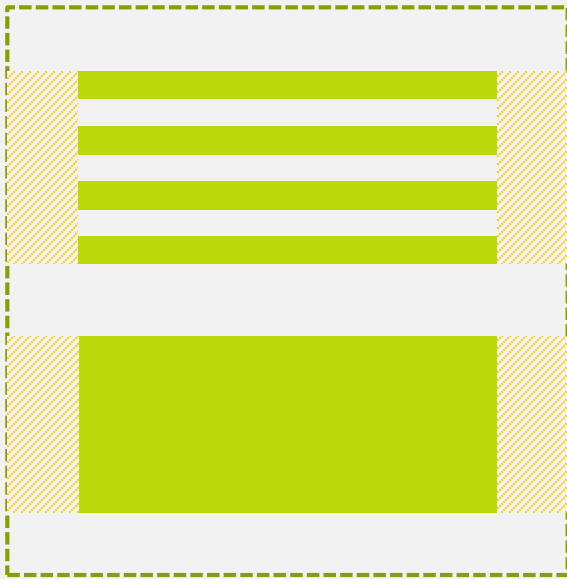
target / context = result

$$\begin{array}{r} 620 \\ \div 1200 \\ \hline 0.516666 \end{array}$$

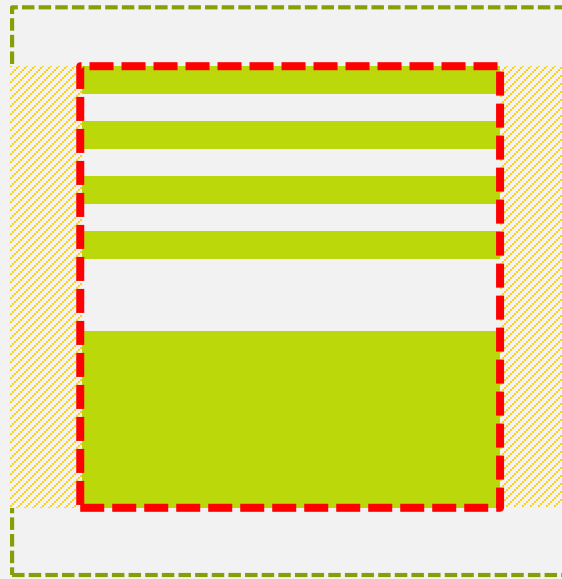
51.66 %



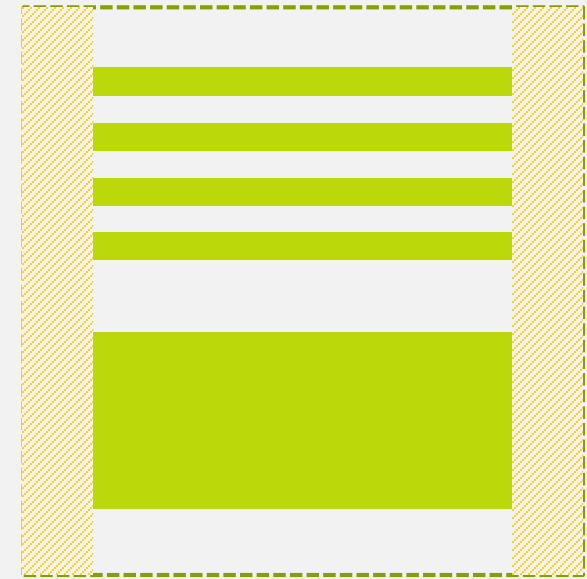
# Approaches to fixed-unit gutters



Margins on every immediate child



Margin on a single buffer container



Border-box

# Exercise

## Building the foundations of Kaffesatz

From the techniques discussed in this section, set up the foundation of the Kaffeesatz page. Make it liquid and ensure proper gutters in place without breaking the layout

You may refer to the following file for the finished version  
`// kaffeesatz / lab / default.html`





# In this section...

## Scaling

Adjusting image size as the viewport width changes

## Swapping

Using scripts to dynamically swap image paths

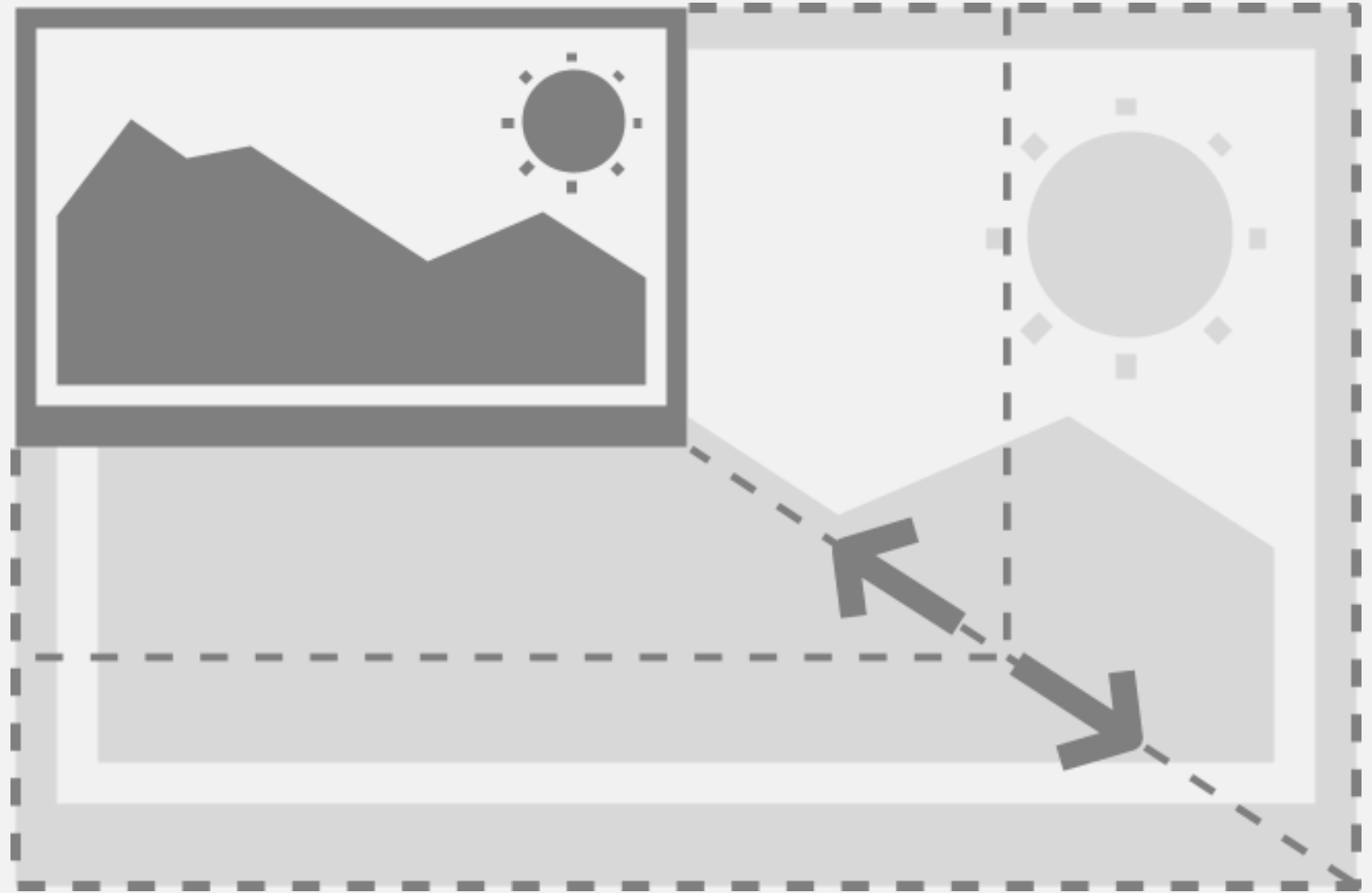
## Cropping

Hiding and revealing portions of the image



# Scaling

A technique that employs assigning **variable measurement units** to one of the image dimensions, allowing the browser to calculate the corresponding measure for the other dimension to maintain image aspect ratio.



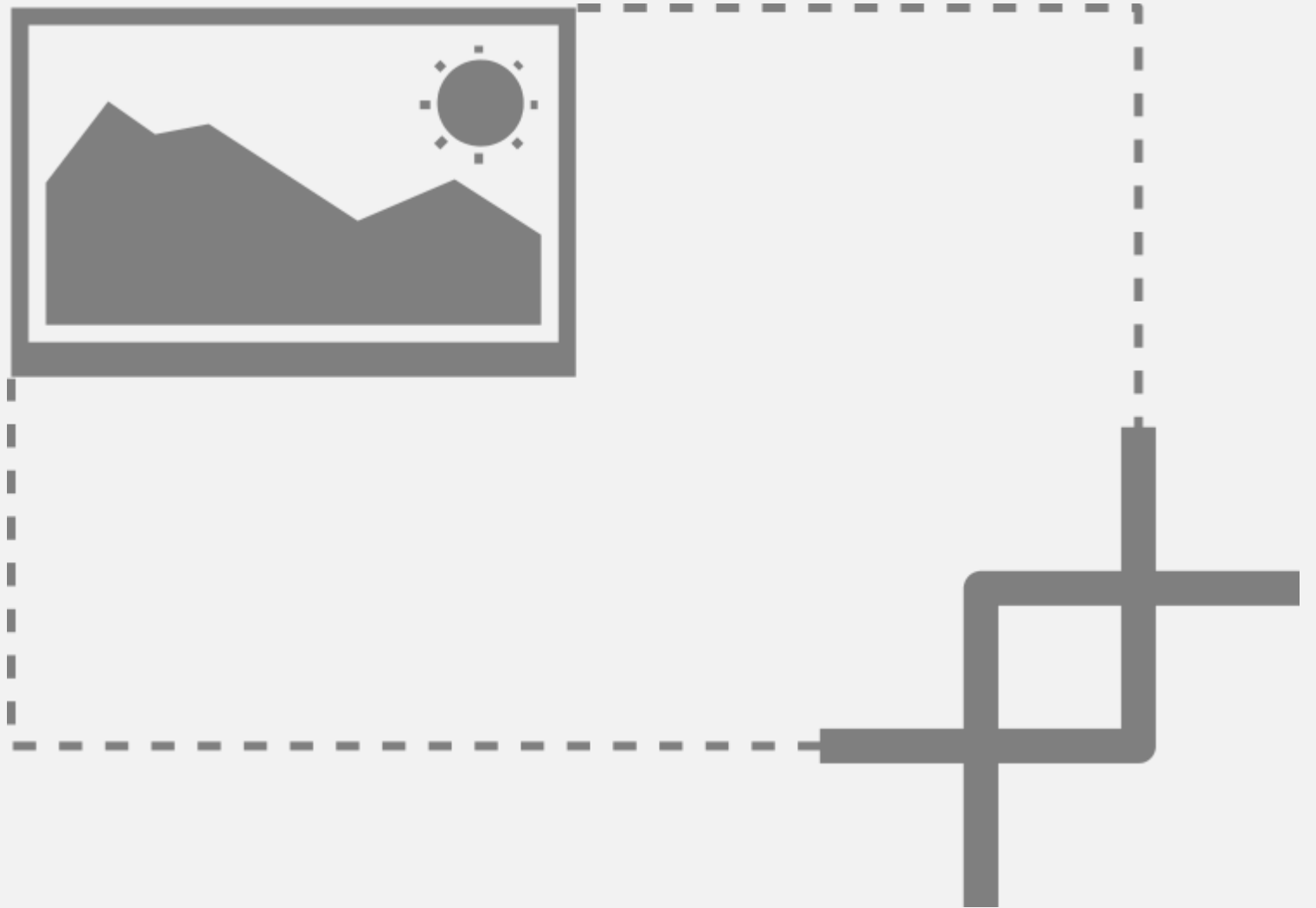
# Swapping

A technique that **swaps referenced image files** as through client- or server-side scripting



# Cropping

A technique that employs **hiding** or **revealing** portions of the image in the screen area without changing the image size.



# Exercise

## Incorporating Images to Kaffesatz

As discussed in this section, continue with the Kaffesatz web page by adding images to locations as specified in the mock up.

Make them flexible! 😊

You may refer to the following file for the finished version  
`// kaffeesatz / lab / default.html`



# MEDIA QUERIES

Altering page layout and features dynamically

# In this section...

## Media Queries

What they are and the significant role they play

## Breakpoints

Setting up boundaries for assumed device widths





a **media query** consists of a media type and zero or more expressions that check for the conditions of particular media features

device-width

device-height

**width**

height

orientation

aspect-ratio

device-aspect-ratio

color

color-index

monochrome

resolution

scan

grid

```
@media only screen and (min-width: 00px) and (max-width: 00px) {  
    /* /// TO DO: Set up styling rules here /// */  
}
```

← Sets the minimum and maximum viewport widths where style rules will apply →



320  
mobile portrait

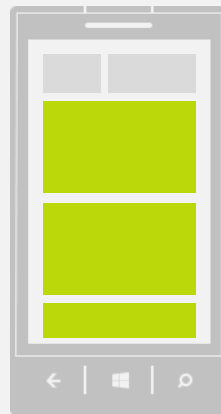
480  
mobile landscape

768  
tablet portrait

1024  
tablet landscape

1200+  
widescreens

a common  
example...



```
/* /// MOBILE PORTRAIT /// */
@media only screen and (min-width: 320px) {

    .unit { width: 100%; }
}

/* /// TABLET LANDSCAPE /// */
@media only screen and (min-width: 1024px) {

    .unit { width: 33.33%; }
} |
```

# Exercise

## Employing Media Queries

Rearranging layout of page elements to accommodate narrow screens

You may refer to the following file for the finished version  
`// kaffeesatz / lab / default.html`



...and that's pretty much everything!

Questions?



JUN SANTOS  
Web Developer, Wunderman  
[jun.santos@outlook.com](mailto:jun.santos@outlook.com)

# References



Responsive Web Design

Ethan Marcotte



Flexible Web Design

Zoe Mickley Gillenwater