Cascading Style Sheets

- A few key points:
  - Elements on a page: **block-level** and **inline**
  - Tags: **<div>** and **<span>**

- Introducing Cascading Style Sheets:
  - How styling works
  - Basic syntax
  - Common selectors
  - How styles are applied (cascade and inheritance)
  - Why you should ALWAYS use external CSS files

- Using CSS:
  - Styling color
  - Styling text

BLOCK-LEVEL ELEMENTS

Appear on a new line

```
<h1>Hiroshi Sugimoto</h1>
<p>The dates of the ORIGIN OF ART
   exhibition are as follows:</p>
<ul>
  <li>Science: 21 Nov - 20 Feb 2010</li>
  <li>Architecture: 6 Mar - 15 May
      2010</li>
  <li>History: 29 May - 21 Aug 2010</li>
  <li>Religion: 28 Aug - 6 Nov 2010</li>
</ul>
```

Hiroshi Sugimoto

The dates for the ORIGIN OF ART exhibition are as follows:

- Science: 21 Nov - 20 Feb 2010/11
- Architecture: 6 Mar - 15 May 2011
- History: 29 May - 21 Aug 2011
- Religion: 28 Aug - 6 Nov 2011

INLINE ELEMENTS

Appear inline!

Timed to a single revolution of the planet around the sun at a 23.4 degrees tilt that plays out the rhythm of the seasons, this `<em>`Origins of Art`</em>` cycle is organized around four themes: `<b>`science, architecture, history`</b>` and `<b>`Religion`</b>`.

Timed to a single revolution of the planet around the sun at a 23.4 degrees tilt that plays out the rhythm of the seasons, this *Origins of Art* cycle is organized around four themes: **science, architecture, history** and **religion**.

GROUPING TEXT & ELEMENTS IN A BLOCK

```
<div id="header">
 <img src="logo.gif" />
 <ul>
  <li><a href="index.html">Home</a></li>
  <li><a href="bio.html">Bio</a></li>
  <li><a href="works.html">Works</a></li>
  <li><a href="contact.html">Contact</a>
  </li>
 </ul>
</div>
```

ANISH KAPOOR

- [Home](#)
- [Biography](#)
- [Works](#)
- [Contact](#)

GROUPING TEXT & ELEMENTS INLINE

```
<p>Anish Kapoor won the Turner Prize in
   1991 and exhibited at the
   <span class="gallery">Tate Modern</span>
   gallery in London in 2003.</p>
```

Anish Kapoor won the Turner Prize in 1991 and exhibited at the TATE MODERN gallery in London in 2003.

UNDERSTANDING CSS: THINKING INSIDE THE BOX

The Cottage Garden

The *cottage garden* is a distinct style of garden that uses an informal design, dense planting and a mixture of ornamental and edible plants.

The Cottage Garden originated in England and its history can be traced back for centuries, although they were re-invented in 1870's England, when stylized versions were formed as a reaction to the more structured and rigorously maintained English estate gardens.

The earliest cottage gardens were more practical than their modern descendants, with an emphasis on vegetables and herbs, along with some fruit trees.

CSS ASSOCIATES STYLE RULES WITH HTML ELEMENTS

```
p {
    font-family:
Arial;}
```

CSS ASSOCIATES STYLE RULES WITH HTML ELEMENTS

SELECTOR

```
p {
    font-family:
Arial;}
```

CSS ASSOCIATES STYLE RULES WITH HTML ELEMENTS

SELECTOR

p {

font-family: Arial;}

DECLARATION

CSS PROPERTIES AFFECT HOW ELEMENTS ARE DISPLAYED

```css
h1, h2, h3 {
          font-family: Arial;
          color: yellow;}
```

CSS PROPERTIES AFFECT HOW ELEMENTS ARE DISPLAYED

```css
h1, h2, h3 {
              font-family: Arial;
              color: yellow;}
```

PROPERTY

CSS PROPERTIES AFFECT HOW ELEMENTS ARE DISPLAYED

```
h1, h2, h3 {
              font-family: Arial;
              color: yellow;}
```

PROPERTY VALUE

INTRODUCING CSS

```html
<h1>From Garden to Plate</h1>
<p>A <i>potager</i> is a French term for
    an ornamental vegetable or kitchen
    garden...</p>

<h2>What to Plant</h2>
<p>Plants are chosen as much for their
    functionality as for their color and
    form...</p>
```

From Garden to Plate

A *potager* is a French term for an ornamental vegetable or kitchen garden. Often flowers (edible and non-edible) and herbs are planted with the vegetables to enhance the garden's beauty. The goal is to make the function of providing food aesthetically pleasing.

What to Plant

Plants are chosen as much for their functionality as for their color and form. Many are trained to grow upward. A well-designed potager can provide food, cut flowers and herbs for the home with very little maintenance. Potagers can disguise their function of providing for a home in a wide array of forms - from the carefree style of the cottage garden to the formality of a knot garden.

INTRODUCING CSS

```
body {
  font-family: Arial, Verdana, sans-serif;}


h1, h2 {
  color: #ee3e80;}


p {
  color: #665544;}
```

INTRODUCING CSS

Change the font used

```css
body {
 font-family: Arial, Verdana, sans-serif;}


h1, h2 {
 color: #ee3e80;}


p {
 color: #665544;}
```

INTRODUCING CSS

```
body {
  font-family: Arial, Verdana, sans-serif;}
```

Change the color of the headings

```
h1, h2 {
  color: #ee3e80;}



p {
  color: #665544;}
```

INTRODUCING CSS

```
body {
  font-family: Arial, Verdana, sans-serif;}


h1, h2 {
  color: #ee3e80;}
```

 Change the color of the paragraphs
```
p {
  color: #665544;}
```

From Garden to Plate

A *potager* is a French term for an ornamental vegetable or kitchen garden. Often flowers (edible and non-edible) and herbs are planted with the vegetables to enhance the garden's beauty. The goal is to make the function of providing food aesthetically pleasing.

What to Plant

Plants are chosen as much for their functionality as for their color and form. Many are trained to grow upward. A well-designed potager can provide food, cut flowers and herbs for the home with very little maintenance. Potagers can disguise their function of providing for a home in a wide array of forms - from the carefree style of the cottage garden to the formality of a knot garden.

Common Selectors

*   universal   select **_all_** elements

h1   element   select **_h1_** elements

h1, h2, h3   group   select **_h1, h2, and h3_** elements

.foo   class   select elements with an attribute **_class="foo"_**

#bar   ID   select the one element with an attribute **_id="bar"_**

foo:hover   pseudo   select all elements foo when the mouse cursor is hovering over them (often used for links)

div p   contextual   select all **_p_** elements which are **_descendents of a div_**

More info: https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction_to_CSS/Selectors

ID ATTRIBUTE

Must be unique: only one element per page

```
<p>Water and air. So very commonplace are
these substances, they hardly attraction
attention and yet they vouchsafe our very
existence.</p>
<p id="pullquote">Every time I view the
sea I feel a sense of calming security, as
if visiting my ancestral home; I embark of
a voyage of seeing.</p>
<p>Mystery of mysteries, water and air are
right there before us in the sea.</p>
```

CLASS ATTRIBUTE

Not unique; can have multiple values

```html
<p class="important">For a one-year period
from November 2010, the Marugame
Genichiro-Inokuma Museum of Contemporary
Art (MIMOCA) will host...</p>
<p>Each will showcase works by the artist
thematically contextualised under the
headings...</p>
<p class="important admittance">Hours:
10:00 - 18:00 (No admittance after
17:30)</p>
```

FOR A ONE-YEAR PERIOD FROM NOVEMBER 2010, THE MARUGAME GENICHIRO-INOKUMA MUSEUM OF CONTEMPORARY ART (MIMOCA) WILL HOST A CYCLE OF FOUR HIROSHI SUGIMOTO EXHIBITIONS.

Each will showcase works by the artist thematically contextualized under the headings "Science," "Architecture," "History" and "Religion" so as to present a comprehensive panorama of the artist's oeuvre.

HOURS: 10:00 - 18:00 (NO ADMITTANCE AFTER 17:30)

The Cascade and Inheritance

How the CSS rules are applied:

- Importance
  - !important
- Specificity
  - A measure of how specific a selector is
- Source order
  - Later rules have more weight
- Inheritance
  - **font size, color** versus **margin, padding, border**

More info:
https://developer.mozilla.org/en-US/docs/Learn/CSS/Introduction_to_CSS/Cascade_and_inheritance

HOW CSS RULES CASCADE

```css
h1 {
  font-family: "Courier New", monospace;
}
* {
  font-family: Arial;
  color: darkgrey;
}


i {color: blue;}
i {color: red;}

p b {color: blue !important;}
p b {color: yellow;}
```

Potatoes

There are *dozens* of different **potato** varieties.

They are usually described as early, second early and maincrop potatoes.

```
<h1>Potatoes</h1>
<p>There are <i>dozens</i> of different <b>potato</b> varieties.</p>
<p>They are usually described as early, second early and maincrop
potatoes</p>
```

USING EXTERNAL CSS

```html
<html>
 <head>
  <title>Using External CSS</title>
  <link href="css/styles.css" rel="stylesheet">
 </head>
 <body>
  <h1>Potatoes</h1>
  <p>There are dozens of...</p>
 </body>
</html>
```

USING INTERNAL CSS

```
<head>
 <title>Using Internal CSS</title>
 <style type="text/css">
   body {
    font-family: arial;
    background-color: #rgb(185,179,175);}
   h1 {
    color: rgb(255,255,255);}
 </style>
</head>
```

Why we almost always use external style sheets

No need to copy style code into each web page

We don't violate the **DRY** principle: **D**on't **R**epeat **Y**ourself

Code easier to maintain

Less chance of human error

Changes to CSS automatically apply to the entire website

Easier for many people to create new pages in same style

etc...

Styling
Color

How to represent: RGB

6 Digit Hexadecimal notation
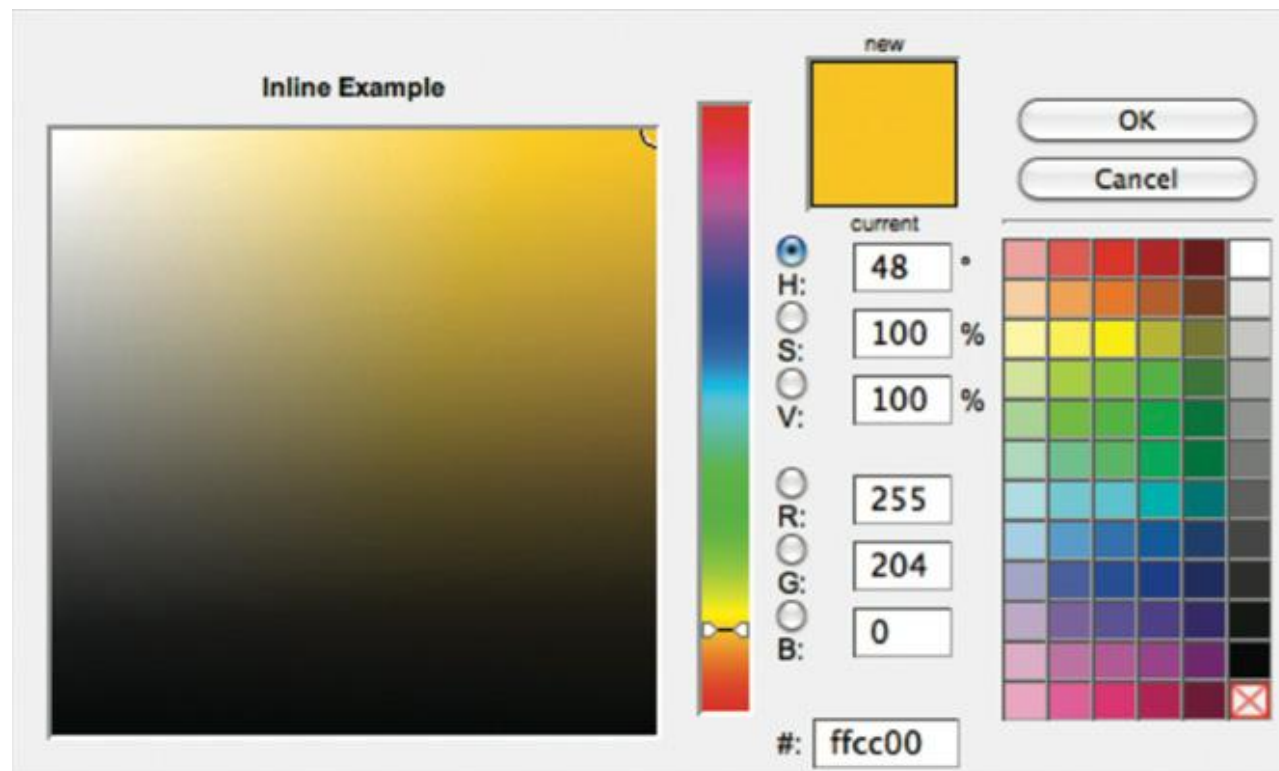
 #RRGGBB, where each digit 0-F hex
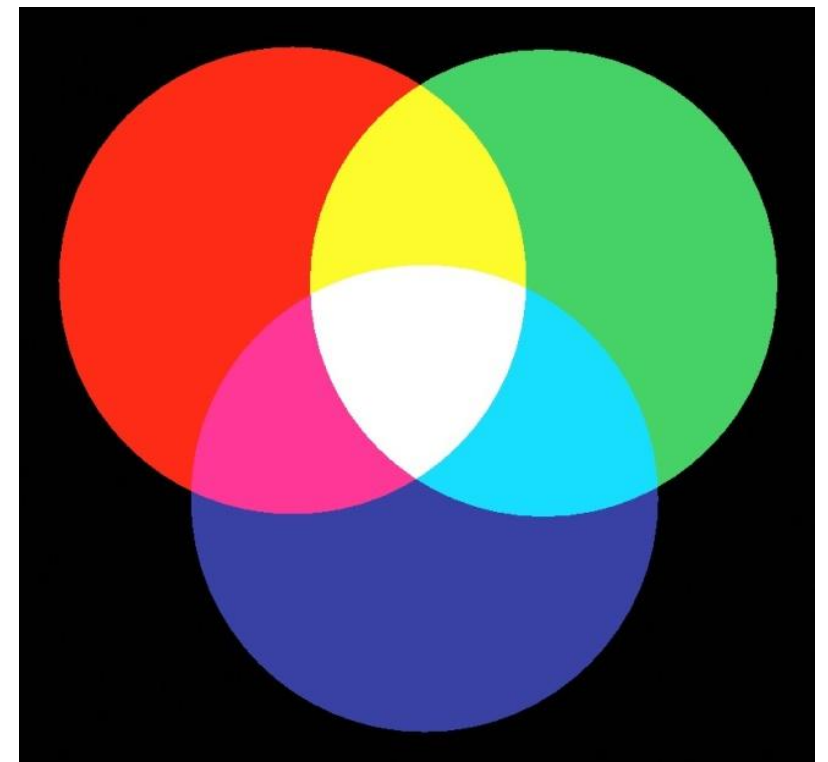
 Example #FF0000

CSS rgb function

- rgb(red, green, blue), values 0-255

- Example rgb(255, 0, 0);

How to represent: RGB

Because the RGB colors combine to create white, they are also called additive colors.

The absence of colored light is black

Adding all colors together creates white.

How to specify color in your stylesheet

```
div {
    color: black;
    background-color: white;
}
```
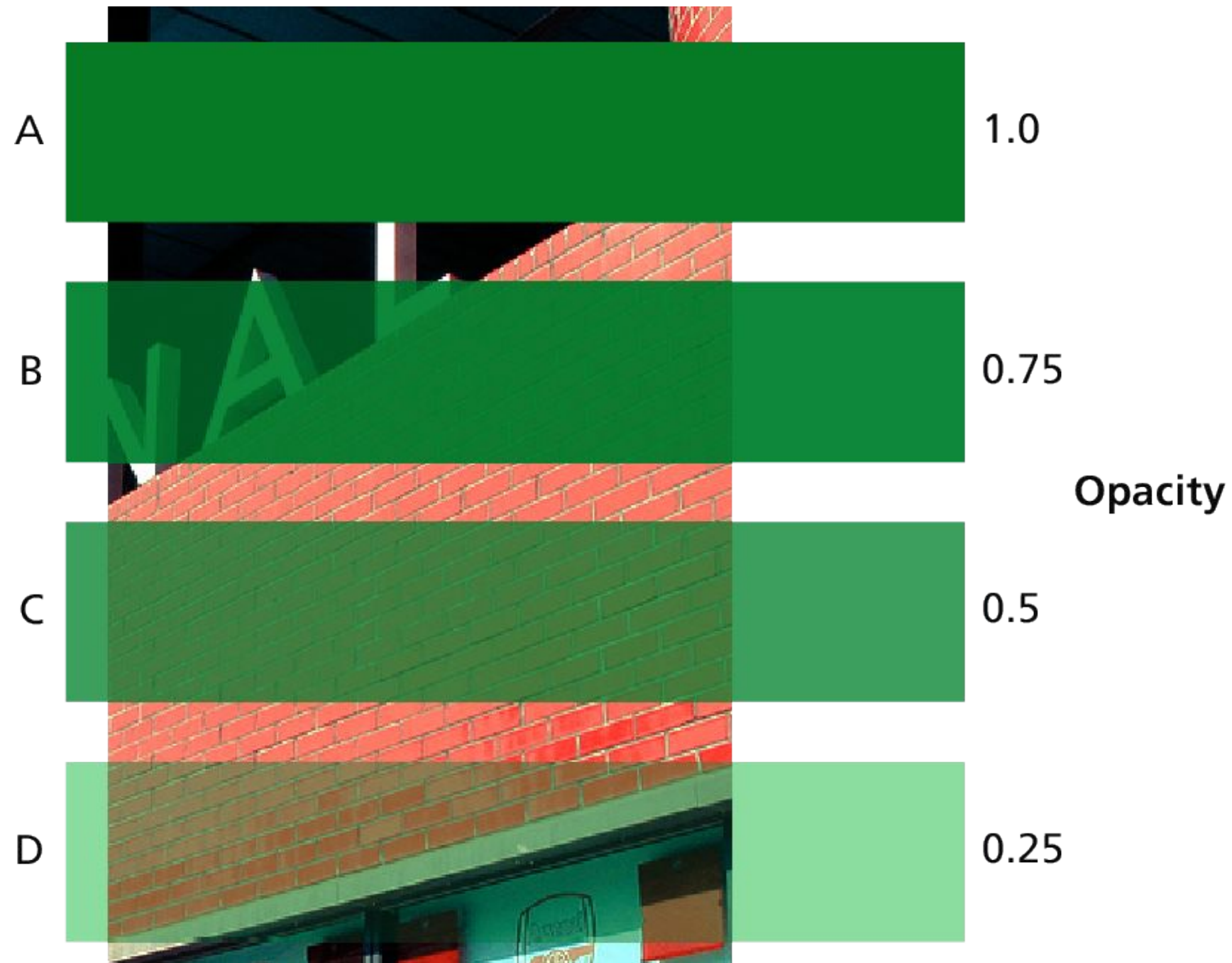
or

```
div {
    color: #000000;
    background-color: #FFFFFF;
}
```

Opacity

Opacity is the degree of transparency in the color. This value is also referred to as **alpha transparency**.

- Opacity is typically a percentage value between 0 and 100 (or between 0 and 1.0)

    - 0 means no opacity (transparent)

    - 100 means that the element is fully opaque (no transparency).

- For example: `p { opacity: 0.75; }`

Opacity

A — 1.0

B — 0.75

C — 0.5

D — 0.25

Opacity

Styling
Text

Text Properties

Two basic types

CSS provides two types of properties that affect text.

- **font properties** that affect the font and its appearance.

- **paragraph properties** that affect the text in a similar way no matter which font is being used.
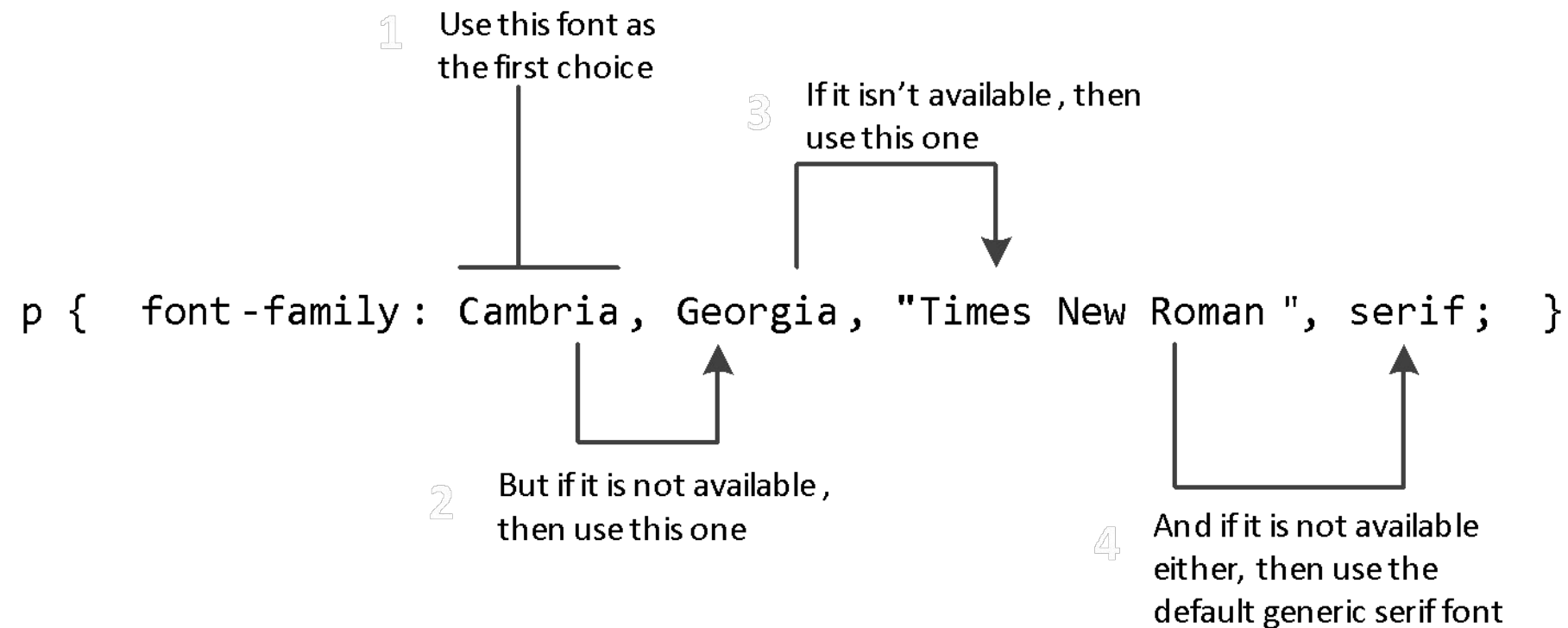
Font-Family

A few issues here

A word processor on a desktop machine can make use of any font that is installed on the computer; browsers are no different.

However, just because a given font is available on the web developer's computer, it does not mean that that same font will be available for all users who view the site.

For this reason, it is conventional to supply a so-called **web font stack**, that is, a series of alternate fonts to use in case the original font choice in not on the user's computer.
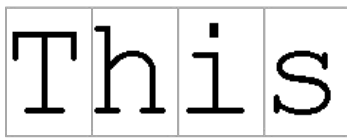
Specifying the Font-Family

1   Use this font as
    the first choice

3   If it isn't available, then
    use this one

p {   font-family : Cambria , Georgia , "Times New Roman ", serif;   }

2   But if it is not available,
    then use this one

4   And if it is not available
    either, then use the
    default generic serif font
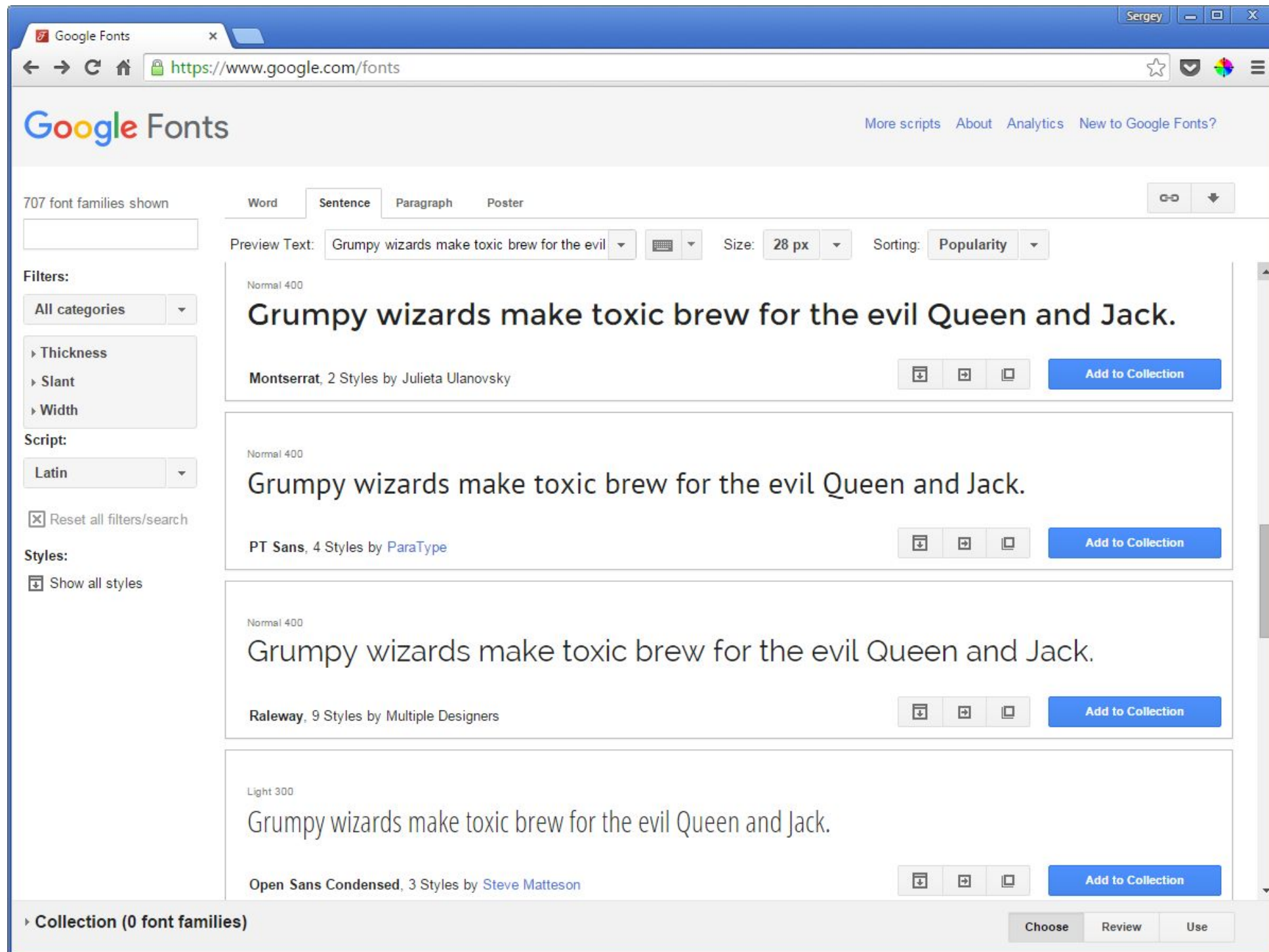
Generic Font-Family

The font-family property supports five different generic families.

The browser supports a typeface from each family.

Generic
Font-Family
Name

This   serif

This   sans-serif

This   monospace

This   cursive

This   fantasy

serif Th

Without ("sans") serif Th

This   In a monospace font, each letter has the same width

This   In a regular, proportionally-spaced font, each letter has a variable width

Decorative and cursive fonts vary from system to system; rarely used as a result .

Google Fonts

https://www.google.com/fonts

Font Sizes

The issue of font sizes is unfortunately somewhat tricky.

In a print-based program such as a word processor, specifying a font size in points is unproblematic.

However, absolute units such as points and inches do not translate very well to pixel-based devices.

Somewhat surprisingly, pixels are also a problematic unit.

Newer mobile devices in recent years have been increasing pixel densities so that a given CSS pixel does not correlate to a single device pixel.

Font Sizes

If we wish to create web layouts that work well on different devices, we should learn to use relative units such as **em** units or **percentages** for our font sizes (and indeed for other sizes in CSS as well).

One of the principles of the web is that the user should be able to change the size of the text if he or she so wishes to do so.

Using percentages or em units ensures that this user action will work.

How to use ems and percents

`<body>`   Browser's default text size is usually 16 pixels

`<p>`   100% or 1em is 16 pixels

`<h3>`   125% or 1.125em is 18 pixels

`<h2>`   150% or 1.5em is 24 pixels

`<h1>`   200% or 2em is 32 pixels

```
/* using 16px scale */

body { font-size: 100%; }
h3 { font-size: 1.125em; }  /* 1.25 x 16 = 18 */
h2 { font-size: 1.5em; }    /* 1.5 x 16  = 24 */
h1 { font-size: 2em; }      /* 2 x 16 = 32 */
```

```
<body>
  <p>this will be about 16 pixels</p>
  <h1>this will be about 32 pixels</h1>
  <h2>this will be about 24 pixels</h2>
  <h3>this will be about 18 pixels</h3>
  <p>this will be about 16 pixels</p>
</body>
```

How to use ems and percents

When used to specify a font size, both em units and percentages are relative to the parent's font size.

How to use ems and percents

It might seem easy ... but it's not ...

While this looks pretty easy to master, things unfortunately can quickly become quite complicated.

Remember that percents and em units are relative to their parents, so if the parent font size changes, this affects all of its contents.

The rem unit

Solution to font sizing hassles?

CSS3 now supports a new relative measure, the **rem** (for root em unit).

This unit is always relative to the size of the root element (i.e., the <html> element).

However, since Internet Explorer prior to version 9 do not support the rem units, you need to provide some type of fallback for those browsers.

Specifying font family

```css
body {
  font-family: Georgia, Times, serif;}

h1, h2 {
  font-family: Arial, Verdana, sans-serif;}

.credits {
  font-family: "Courier New", Courier,
               monospace;}
```

Briards

by Ivy Duckett

The briard, or berger de brie, is a large breed of dog traditionally used as a herder and guardian of sheep.

Breed History

The briard, which is believed to have originated in France, has been bred for centuries to herd and to protect sheep. The breed was used by the French Army as sentries, messengers and to search for wounded soldiers because of its fine sense of hearing. Briards were used in the First World War almost to the point of extinction. Currently the population of briards is slowly recovering. Charlemagne, Napoleon, Thomas Jefferson and Lafayette all owned briards.

Specifying font size

```
body {
  font-family: Georgia, Times, serif;
  font-size: 12px;}


h1 {
  font-size: 200%;}


.credits {
  font-size: 1.3rem;}
```

Briards

by Ivy Duckett

The [briard](), or berger de brie, is a large breed of dog traditionally used as a herder and guardian of sheep.

Breed History

The briard, which is believed to have originated in France, has been bred for centuries to herd and to protect sheep. The breed was used by the French Army as sentries, messengers and to search for wounded soldiers because of its fine sense of hearing. Briards were used in the First World War almost to the point of extinction. Currently the population of briards is slowly recovering. Charlemagne, Napoleon, Thomas Jefferson and Lafayette all owned briards.

Some useful properties

- font-weight; font-style

- text-transform; text-decoration; text-align

- text-shadow

- :first-letter, first:line

- :link, :visited, :hover

- Here's a reference: http://www.w3schools.com/cssref