

Курс "Основи web-програмування"

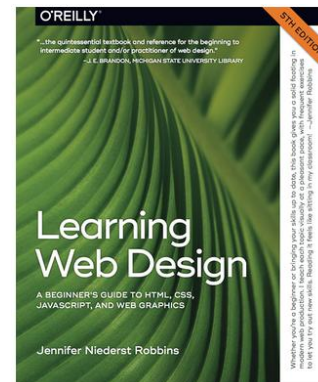
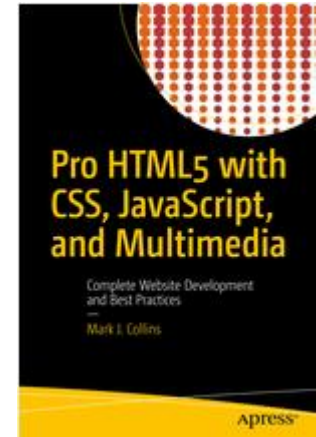
частина 3



Мета – здобуття теоретичних знань і практичних навичок з основ web-технологій.

Література

1. Jörg Krause **Introducing Web Development** – Apress, 2016
2. Jennifer Robbins **Learning Web Design** Fifth Edition– O'Reilly Media, 2018. 790 p.
3. John Dean **Web Programming with HTML5, CSS and JavaScript** - Jones & Bartlett Learning, 2019. 678 p.
4. Mark J. Collins **Pro HTML5 with CSS, JavaScript, and Multimedia** - Apress, 2017. 560 p.
5. Майер Эрик, Уэйл Эстелл **CSS: полный справочник**, 4-е изд.: Пер. с англ. - СПб.: ООО "Диалектика": 2019. - 1088 с.



Корисні посилання

- **W3 Schools** (<http://www.w3schools.com/>)
- **Ресурси для розробників**
(<https://developer.mozilla.org/uk/>)
- **Генератор шаблонів сайтів** - csstemplater.com
- **Інструмент вибору кольору**
https://developer.mozilla.org/ru/docs/Web/CSS/CSS_Colors/Color_picker_tool
- **GoogleFonts** <https://fonts.google.com/>
- **HTML5 BOOK** <https://html5book.ru/>
- **ТАБЛИЦЯ БЕЗПЕЧНИХ КОЛЬОРІВ**
<http://getcolorcode.com/ua/colors/websafe>
- **Free Image Placeholder Service** <https://placeholder.it>
- **Lorem ipsum generator online** http://uk.lorem-ipsum.info/_latin

- CSS ZEN GARDEN <http://www.csszengarden.com/>
- A Complete Guide to Flexbox
<https://css-tricks.com/snippets/css/a-guide-to-flexbox/>
- **Flexbox Froggy** <http://flexboxfroggy.com/#ru>

Стилізація тексту

Властивості шрифту,
що підтримуються в
CSS2.1:

- font-family
- font-size
- font-weight
- font-style
- font-variant
- font

Модуль CSS Font Module Level 3
додає такі властивості:

font-stretch
font-variant-ligatures
font-variant-position
font-variant-caps
font-variant-numeric
font-variant-alternates
font-variant-east-asian
font-size-adjust
font-kerning
font-feature-settings
font-language-override

Fonts

Отримання шрифтів

Існує три способи, за допомогою яких браузер може визначити шрифт, щоб він міг бути відображений.

1) **Web-Safe Fonts** (Веб-безпечні шрифти)

Усі операційні системи мають вбудовані шрифти. Браузери можуть використовувати будь-який із цих вбудованих шрифтів. Однак проблема полягає в тому, що всі операційні системи не підтримують однаковий набір шрифтів.

Веб-безпечний шрифт - це той, що зазвичай підтримується більшістю пристроїв. Це Arial, Verdana, Times New Roman і Courier New.

На сторінці <http://www.cssfontstack.com/> є стаття, що показує наявність шрифту для Windows та Mac (на жаль, не немає покриття Android або iOS).

2) **Web Fonts**

Веб-шрифт завантажується браузером так само, як і інші ресурси, такі як CSS та файли зображень. У цього підходу є два недоліки:

- **Старі браузери не підтримують веб-шрифти**; однак це впливає лише на невелику та скорочувальну сукупність користувачів.
- Додаткове завантаження може **сповільнити час завантаження сторінки**, особливо якщо ви використовуєте кілька веб-шрифтів. Вони, як правило, кешуються браузером, тому зазвичай це впливає лише на початкове завантаження.

Якщо вам цікаво використовувати веб-шрифти, перейдіть на сторінку <https://fonts.google.com/>. Цей сайт пропонує більше 800 сімейств шрифтів.

- Після вибору шрифту сайт надає інструкції, як його включити у свій HTML-документ.

```
<link href="https://fonts.googleapis.com/css?family=Cabin" rel="stylesheet">
```

Крім того, ви можете використовувати правило **@import** для включення шрифту у файл CSS таким чином:

```
@import url(//fonts.googleapis.com/css?family=Cabin);
```

3) **Custom Fonts**

Третій підхід - вбудувати власні визначення шрифту у файл CSS. Так само, як і для зображень, існує кілька форматів для визначення шрифту. Шрифти True Type (**TTF**) існують вже близько 30 років, і це разом із шрифтами Open Type (**OTF**) є стандартом для встановлення шрифтів на настільному комп'ютері. Ці формати також підтримуються основними веб-браузерами. Однак формат шрифтів Web Open (**WOFF**) є рекомендацією W3C, а також підтримується більшістю браузерів.

Font Families

- Шрифти ідентифікуються терміном `family`. Терміни `Arial`, `Courier New` і `Times New Roman` - це шрифтові `family`.

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

```
var { font-family: Courier, monospace; }
```

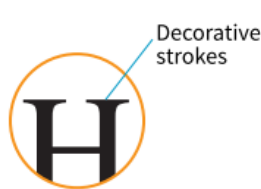
```
p { font-family: "Duru Sans", Verdana, sans-serif; }
```

Існує також набір сімейств загальних шрифтів (`generic font families`), приклад кожного наведено на малюнку:

- `cursive` *cursive*
- `fantasy` `fantasy`
- `monospace` `monospace`
- `sans-serif` `sans-serif`
- `serif` `serif`

- serif - Examples: Times, Times New Roman, Georgia
- sans-serif - Examples: Arial, Arial Black, Verdana, Trebuchet MS, Helvetica, Geneva
- monospace - Examples: Courier, Courier New, and Andale Mono
- cursive - Examples: Apple Chancery, Zapf-Chancery, and Comic Sans
- fantasy - Examples: Impact, Western

serif



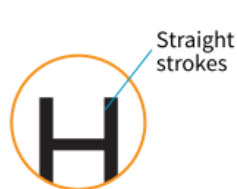
Hello
Times

Hello
Times New Roman

Hello
Georgia

Hello
Lucida

sans-serif



Hello
Verdana

Hello
Arial

Hello
Trebuchet MS

Hello
Arial Black

monospace

W i

Monospace font
(equal widths)

W i

Proportional font
(different widths)

Hello

Courier

Hello

Courier New

Hello

Andale Mono

cursive

Hello

Apple Chancery

Hello

Comic Sans

Hello

Snell

fantasy

Hello

Impact

HELLO

Stencil

HELLO

Mojo

EXERCISE 12-1. Formatting a menu p.268

3 книги Learning Web Design 5th 2018

```
<head>
<title>Black Goose Bistro</title>
<link href="http://fonts.googleapis.com/css?family=Marko+One"
      rel="stylesheet">
<style>
  body {font-family: Verdana, sans-serif;} . . . . .
  h1 {font-family: "Marko One", Georgia, serif;}
</style>
```



Specifying Font Size

font-size

Values: *length unit* | *percentage* | xx-small | x-small | small | medium | large | x-large | xx-large | smaller | larger

Default: medium

Applies to: all elements

Inherits: yes

Приклад.

```
h1 { font-size: 1.5em; }
```

■ AT A GLANCE

CSS Units Cheat Sheet

As a quick reference, here are the CSS length units again:

Relative units

em	ex	rem	ch
vw	vh	vmin	vmax

Absolute units

px	in	mm	cm
q	pt	pc	

Font Style (Italics)

font-style

Values: normal | italic | oblique

Default: normal

Applies to: all elements

Inherits: yes

Приклад:

```
strong { font-style: italic;}
```

Font Weight (Boldness)

font-weight

Values:	normal bold bolder lighter 100 200 300 400 500 600 700 800 900
Default:	normal
Applies to:	all elements
Inherits:	yes

Приклад.

```
dt { font-weight: bold; }
```

CHANGING TEXT COLOR

color

Values: *color value (name or numeric)*

Default: depends on the browser and user's preferences

Applies to: all elements

Inherits: yes

Приклади.

```
h1 { color: gray; }
```

```
h1 { color: #666666; }
```

```
h1 { color: #666; }
```

```
h1 { color: rgb(102,102,102); }
```

■ AT A GLANCE

Color Names

CSS2.1 defines 17 standard color names:

black	white	purple
lime	navy	aqua
silver	maroon	fuchsia
olive	blue	orange
gray	red	green
yellow	teal	

Line Height

line-height

Values: *number* | *length measurement* | *percentage* | `normal`

Default: `normal`

Applies to: all elements

Inherits: yes

The line-height property defines the minimum distance from baseline to baseline in text.

Baseline ... A baseline is the imaginary line upon which the bottoms of characters sit. Line height in CSS is similar to leading in traditional typesetting.

```
line-height: 2em;
```

```
p { line-height: 2; }
```

```
p { line-height: 2em; }
```

```
p { line-height: 200%; }
```

Indents

text-indent


Values: *length measurement | percentage*

Default: 0

Applies to: block containers


Inherits: yes

2em




Paragraph 1. The text-indent property indents only the first line of text by a specified amount. You can specify a length measurement or a percentage value.

25%



Paragraph 2. The text-indent property indents only the first line of text by a specified amount. You can specify a length measurement or a percentage value.

-35px



Paragraph 3. The text-indent property indents only the first line of text by a specified amount. You can specify a length measurement or a percentage value.

```
p#1 { text-indent: 2em; }  
p#2 { text-indent: 25%; }  
p#3 { text-indent: -35px; }
```

Horizontal Text Alignment

text-align

Values: left | right | center | justify | start | end

Default: start

Applies to: block containers

Inherits: yes

`text-align: left;`

Paragraph 1. The text-align property controls the horizontal alignment of the text within an element. It does not affect the alignment of the element on the page. The resulting text behavior of the various values should be fairly intuitive.

`text-align: right;`

Paragraph 2. The text-align property controls the horizontal alignment of the text within an element. It does not affect the alignment of the element on the page. The resulting text behavior of the various values should be fairly intuitive.

`text-align: center;`

Paragraph 3. The text-align property controls the horizontal alignment of the text within an element. It does not affect the alignment of the element on the page. The resulting text behavior of the various values should be fairly intuitive.

`text-align: justify;`

Paragraph 4. The text-align property controls the horizontal alignment of the text within an element. It does not affect the alignment of the element on the page. The resulting text behavior of the various values should be fairly intuitive.

text-decoration

Values: none | underline | overline | line-through | blink

Default: none

Applies to: all elements

I've got laser eyes.

`text-decoration: underline;`

I've got laser eyes.

`text-decoration: overline;`

~~I've got laser eyes.~~

`text-decoration: line-through;`

CHANGING CAPITALIZATION

text-transform

Values: none | capitalize | lowercase | uppercase | full-width

Default: none

Applies to: all elements

`text-transform: none;`
(as it was typed in the source)

And I know what you're thinking.

`text-transform: capitalize;`

And I Know What You're Thinking.

`text-transform: lowercase;`

and i know what you're thinking.

`text-transform: uppercase;`

AND I KNOW WHAT YOU'RE THINKING.

SPACED OUT

letter-spacing

Values: *length measurement* | normal

Default: normal

word-spacing

Values: *length measurement* | normal

Default: normal

<p>Black Goose Bistro Summer Menu</p>

B l a c k G o o s e B i s t r o S u m m e r M e n u

```
p { letter-spacing: 8px; }
```

Black Goose Bistro Summer Menu

```
p { word-spacing: 1.5em; }
```

TEXT SHADOW

text-shadow

Values: *'horizontal offset' 'vertical offset' 'blur radius' 'color' | none*

Default: none

```
h1 {  
  color: darkgreen;  
  text-shadow: .2em .2em silver;  
}
```

```
h1 {  
  color: darkgreen;  
  text-shadow: -.3em -.3em silver;  
}
```

The Jenville Show

```
text-shadow: .2em .2em silver;
```

The Jenville Show

```
text-shadow: -.3em -.3em silver;
```

COLORS AND BACKGROUNDS

- Color Names



FIGURE 13-1. The 17 standard color names in CSS2.1. Інженерний інститут ЗНУ,
кафедра ІТІСАС, доц. Попівцій В.І., 2020

- CSS3 додає підтримку розширеного набору зі 140 (досить вигадливих) імен кольорів (Дивись Learning Web Design 5th 2018, p.305).

aliceblue 240,248,255 F0F8FF	cornsilk 255,248,220 FFF8DC	darkturquoise 0,206,209 00CED1	hotpink 255,105,180 FF69B4
antiquewhite 250,235,215 FAEBD7	crimson 220,20,60 DC143C	darkviolet 148,0,211 9400D3	indianred 205,92,92 CD5C5C
aqua 0,255,255 00FFFF	cyan 0,255,255 00FFFF	deeppink 255,20,147 FF1493	indigo 75,0,130 4B0082
aquamarine 127,255,212 7FFFD4	darkblue 0,0,139 00008B	deepskyblue 0,191,255 00BFFF	ivory 255,240,240 FFF0F0
azure 240,255,255 F0FFFF	darkcyan 0,139,139 008B8B	dimgray 105,105,105 69,69,69	khaki 240,230,140 F0D58C

RGB Color Values

color: rgb(200, 178, 230);
color: rgb(78%, 70%, 90%);
color: #C8B2E6;
color: #FC0;

■ AT A GLANCE

Specifying RGB Values

There are four formats for providing RGB values in CSS:

rgb(255, 255, 255)

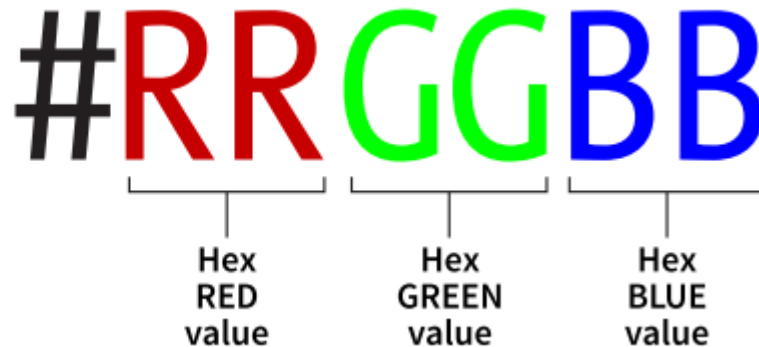
rgb(100%, 100%, 100%)

#FFFFFF

#FFF

All of these examples specify white.

Hexadecimal RGB values must be preceded by the # symbol.



RGBa Color

- "a" у "RGBa" означає alpha, що є додатковим каналом, який контролює рівень прозорості за шкалою від 0 (повністю прозорий) до 1 (повністю непрозорий).

Playing with RGBa

Playing with RGBa

Playing with RGBa

```
color: rgba(0, 0, 0, .1);
```

```
color: rgba(0, 0, 0, .5);
```

```
color: rgba(0, 0, 0, 1);
```

FOREGROUND COLOR (Колір переднього плану)

THE STYLE RULE

```
blockquote {  
  border: 4px dashed;  
  color: green;  
}
```

THE MARKUP

```
<blockquote>
```

In the latitude of central
New England,
cabbages are not
secure ...

```
</blockquote>
```

The foreground of an element consists of its text and border (if one is specified).

In the latitude of central New England, cabbages are not secure from injury from frost with less than a foot of earth thrown over the heads. In mild winters a covering of half that depth will be sufficient; but as we have no prophets to foretell our mild winters, a foot of earth is safer than six inches.

BACKGROUND COLOR

background-color

Values: *color value (name or numeric)* | transparent

Default: transparent

Applies to: all elements

Inherits: no

```
blockquote {  
  border: 4px dashed;  
  color: green;  
  background-color: #c6de89;  
}
```

In the latitude of central New England, cabbages are not secure from injury from frost with less than a foot of earth thrown over the heads. In mild winters a covering of half that depth will be sufficient; but as we have no prophets to foretell our mild winters, a foot of earth is safer than six inches.

CLIPPING THE BACKGROUND

background-clip

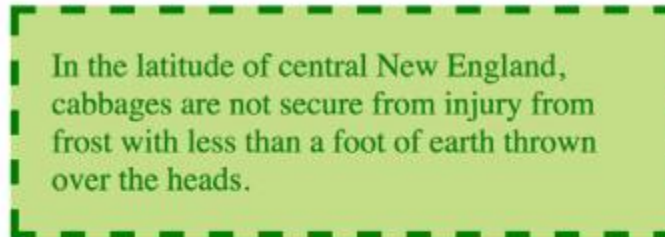
Values: border-box | padding-box | content-box

Default: border-box

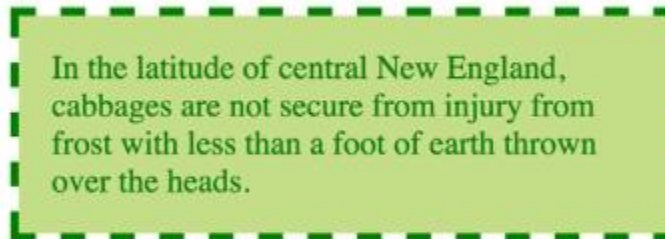
Applies to: all elements

Inherits: no

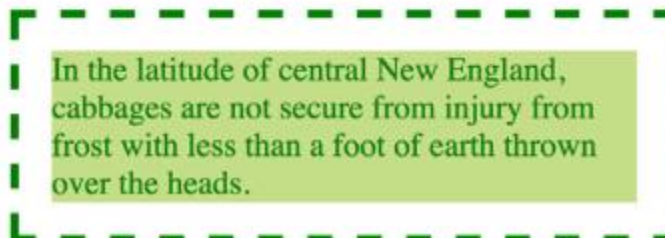
```
blockquote {  
padding: 1em;  
border: 4px  
dashed; color:  
green;  
background-  
color: #C6DE89;}  
}
```



background-clip: border-box;



background-clip: padding-box;



background-clip: content-box;

OPACITY (непрозорість)

```
h1 {color: gold; background:  
white; opacity: .25;}
```

```
h1 {color: gold; background:  
white; opacity: .5;}
```

```
h1 {color: gold; background:  
white; opacity: 1;}
```

opacity

Values: *number* (0 to 1)

Default: 1

Applies to: all elements

Inherits: no



Playing with opacity

```
opacity: .25;
```



Playing with opacity

```
opacity: .5;
```

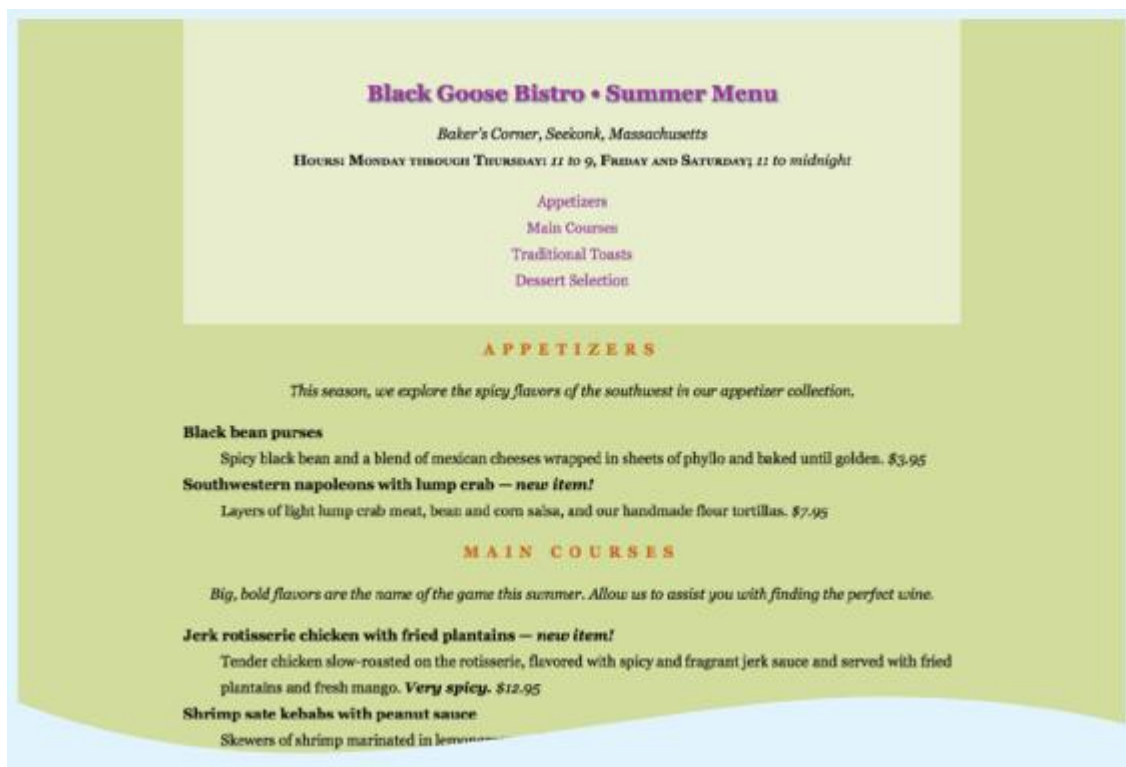


Playing with opacity

```
opacity: 1;
```

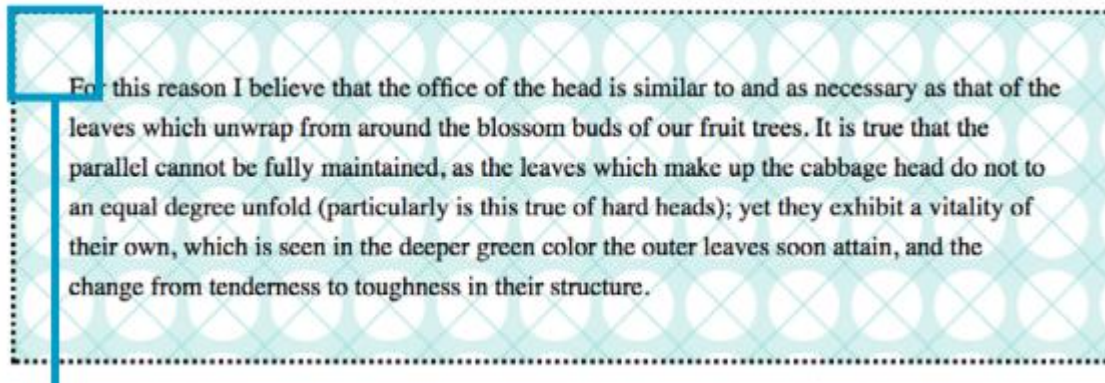
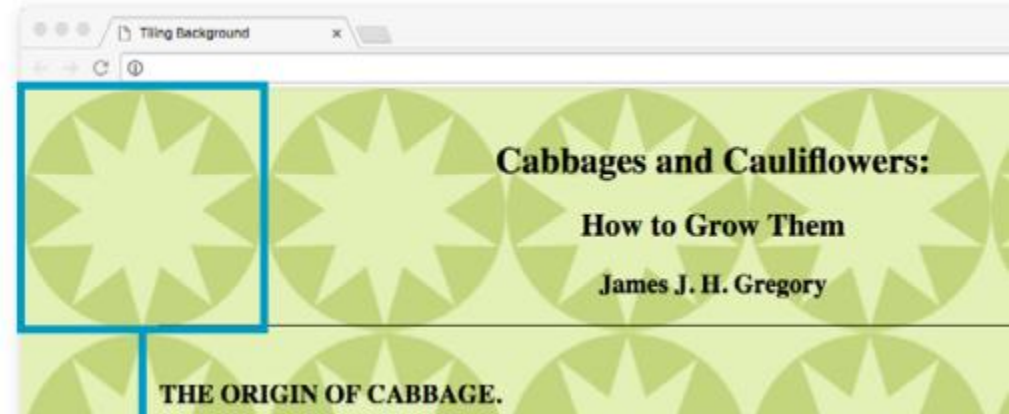
EXERCISE 13-1. Adding color to a document

(Learning Web Design 5th 2018, EXERCISE 13-1. стор.325)



Adding a Background Image

```
body {  
  background-image:  
    url(star.png);  
}  
blockquote {  
  background-image:  
    url(dot.png);  
  padding: 2em;  
  border: 4px dashed;  
}
```



star.png
150 × 150 pixels



dot.png
50 × 50 pixels

Background Repeating

background-repeat

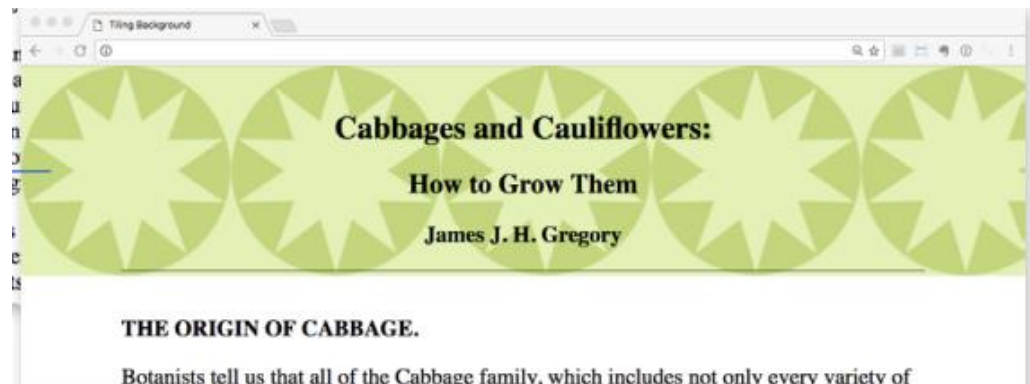
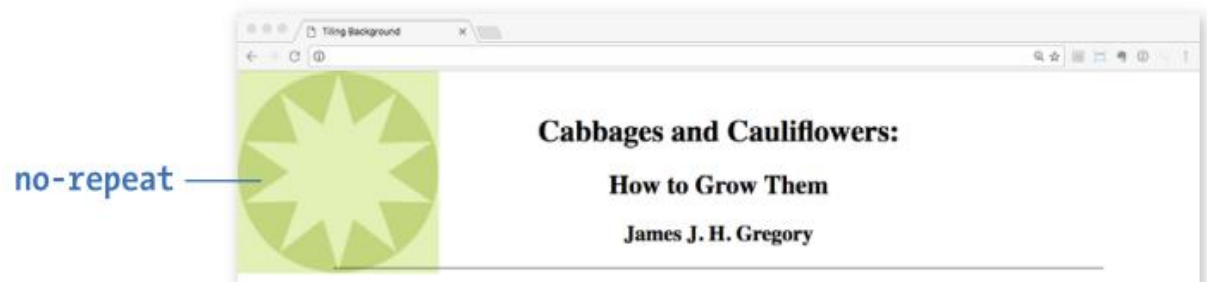
Values: repeat | no-repeat | repeat-x | repeat-y | space | round

Default: repeat

Applies to: all elements

Inherits: no

```
body {  
  background-image:  
    url(star.png);  
  background-repeat:  
    no-repeat;  
}
```



Background Position

background-position

Values: `length measurement | percentage | left | center | right | top | bottom`

Default: `0% 0%` (same as left top)

Applies to: all elements

background-position: left bottom;

For this reason I believe that the office of the head is similar to and as necessary as that of the leaves which unwrap from around the blossom buds of our fruit trees. It is true that the parallel cannot be fully maintained, as the leaves which make up the cabbage head do not to an equal degree unfold (particularly is this true of hard heads); yet they exhibit a vitality of their own, which is seen in the deeper green color the outer leaves soon attain, and the change from tenderness to toughness in their structure.

Background Attachment

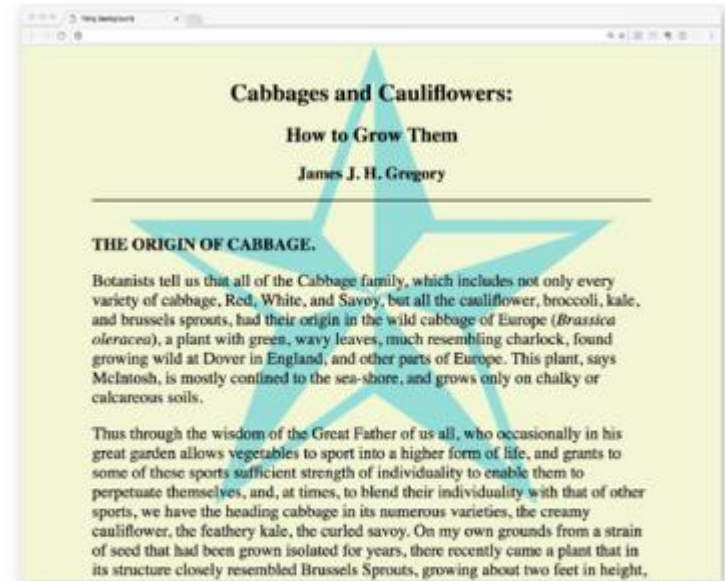
background-attachment

Values: scroll | fixed | local

Default: scroll

Applies to: all elements

```
body {  
  background-image:  
    url(images/bigstar.gif);  
  background-repeat: no-repeat;  
  background-position: center 300px;  
  background-attachment: fixed;  
}
```



A large non-repeating background image in the body of the document.

Background Size

background-size

Values: *length* | *percentage* | *auto* | *cover* | *contain*

Default: *auto*

```
div#A {  
  background-image:  
  url(target.png);  
  background-size: cover; }  
div#B {  
  background-image:  
  url(target.png);  
  background-size: contain; }
```

```
div#B {  
  background-image:  
  url(target.png);  
  background-size: contain; }
```

WHAT A CABBAGE IS.

If we cut vertically through the middle of the head, we shall find it made up of successive layers of leaves, which grow smaller and smaller, almost ad infinitum. Now, if we take a fruit bud from an apple-tree and make a similar section of it, we shall find the same structure. If we observe the development of the two, as spring advances, we shall find another similarity (the looser the head the closer will be the resemblance),—the outer leaves of each will unwrap and unfold, and a flower stem will push out from each. Here we see that a cabbage is a bud, a seed bud (as all fruit buds may be termed, the production of seed being the primary object in nature, the fruit enclosing it playing but a secondary part), the office of the leaves being to cover, protect, and afterwards nourish the young seed shoot.

The outer leaves which surround the head appear to have the same office as the leaves which surround the growing fruit bud, and that office closes with the first year, as does that of the leaves surrounding fruit buds, when each die and drop off. In my locality the public must have perceived more or less clearly the analogy between the heads of cabbage and the buds of trees, for when they speak of small heads they frequently call them "buds." That the close wrapped leaves which make the cabbage head and surround the seed germ, situated just in the middle of the head at the termination of the stump, are necessary for its protection and nutrition when young, is proved, I think, by the fact that those cabbages, the heads of which are much decayed, when set out for seed, no matter how sound the seed germ may be at the end of the stump, never make so large or healthy a seed shoot as those do the heads of which are sound; as a rule, after pushing a feeble growth, they die.

WHAT A CABBAGE IS.

If we cut vertically through the middle of the head, we shall find it made up of successive layers of leaves, which grow smaller and smaller, almost ad infinitum. Now, if we take a fruit bud from an apple-tree and make a similar section of it, we shall find the same structure. If we observe the development of the two, as spring advances, we shall find another similarity (the looser the head the closer will be the resemblance),—the outer leaves of each will unwrap and unfold, and a flower stem will push out from each. Here we see that a cabbage is a bud, a seed bud (as all fruit buds may be termed, the production of seed being the primary object in nature, the fruit enclosing it playing but a secondary part), the office of the leaves being to cover, protect, and afterwards nourish the young seed shoot.

The outer leaves which surround the head appear to have the same office as the leaves which surround the growing fruit bud, and that office closes with the first year, as does that of the leaves surrounding fruit buds, when each die and drop off. In my locality the public must have perceived more or less clearly the analogy between the heads of cabbage and the buds of trees, for when they speak of small heads they frequently call them "buds." That the close wrapped leaves which make the cabbage head and surround the seed germ, situated just in the middle of the head at the termination of the stump, are necessary for its protection and nutrition when young, is proved, I think, by the fact that those cabbages, the heads of which are much decayed, when set out for seed, no matter how sound the seed germ may be at the end of the stump, never make so large or healthy a seed shoot as those do the heads of which are sound; as a rule, after pushing a feeble growth, they die.

BACKGROUND PROPERTY

background

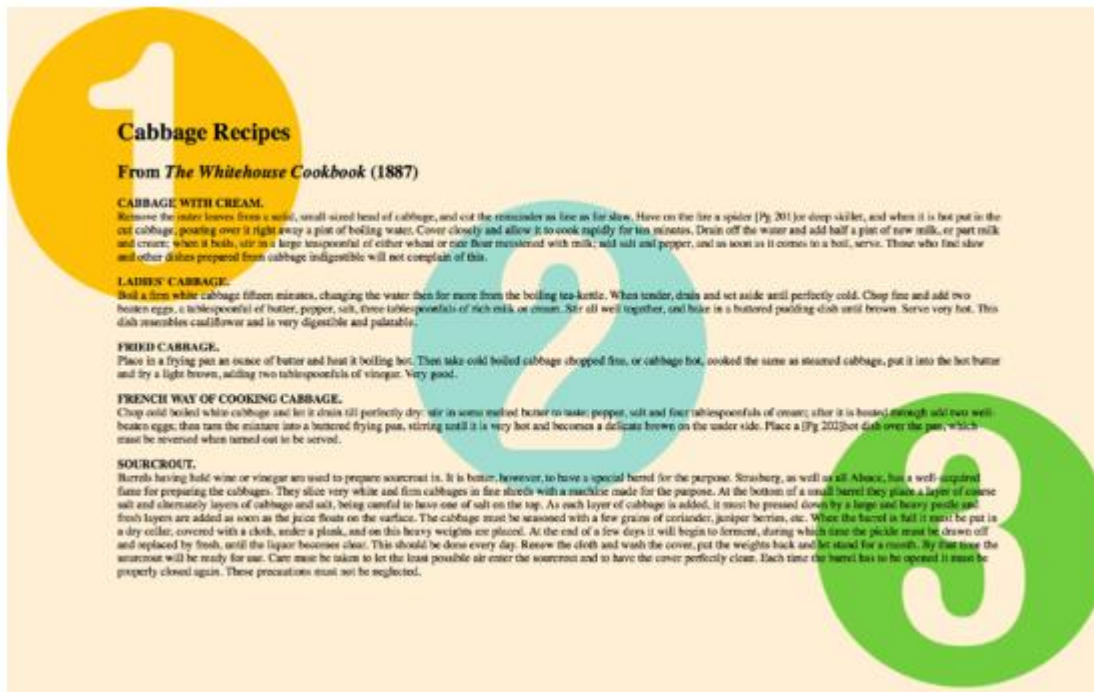
Values: *background-color background-image background-repeat
background-attachment background-position background-clip
background-origin background-size*

```
body { background: white url(star.png) no-repeat right top fixed; }
```

```
body {  
  background-color: white;  
  background-image: url(star.png);  
  background-repeat: no-repeat;  
  background-position: right top;  
  background-attachment: fixed;  
}
```

Multiple Backgrounds

```
body {  
  background:  
  url(image1.png) left top no-repeat,  
  url(image2.png) center center no-repeat,  
  url(image3.png) right bottom no-repeat;  
}
```



GRADIENTS

- **Градiєнт** - це перехід від одного кольору до іншого, іноді через кілька кольорів.
- Градiєнти можуть бути застосовані в будь-якому місці де зображення може бути застосовано: `background-image`, `border-image`, та `list-style-image`. Ми будемо розглядати приклади фонових зображень.

Існує два типи градiєнтів:

- Лінійні градiєнти (**Linear gradients**) змінюють кольори по лінії, від одного краю елемента до іншого.
- Радіальні градiєнти (**Radial gradients**) починаються в точці і поширюються назовні круглої або еліптичної форми.

Linear Gradients

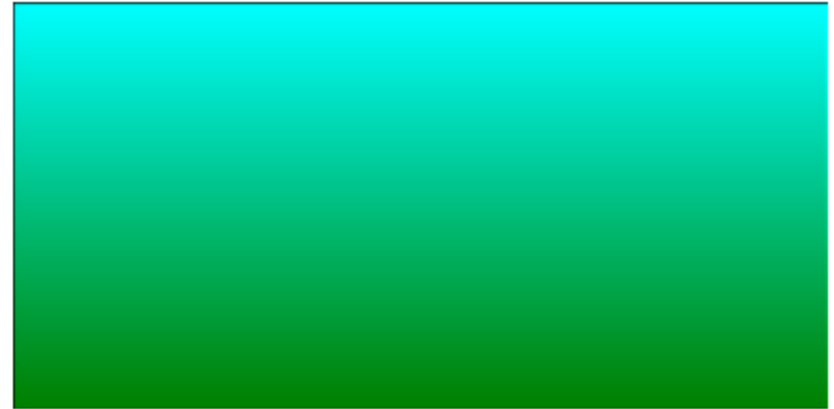
- Означення `linear-gradient()` задає **кут** лінії градієнта та одну або більше **точок** вздовж тієї лінії, де розміщений чистий **колір** (color stops).
- Ви можете використовувати **назви кольорів** або будь-яке числове значення кольорів, включаючи прозорість.
- Кут лінії градієнта задається в градусах (`ndeg`) або за допомогою ключових слів.
- З градусами, `0deg` точки вгору, а позитивні кути йдуть навколо годинникової стрілки, так що `90deg` вказує **праворуч**. Тому, якщо ви хочете перейти від аква на верхньому краю до зеленого на нижньому краю, встановіть обертання на `180deg`:

`background-image: linear-gradient(180deg, aqua, green);`

The keywords describe direction in increments of 90° (to top, to right, to bottom, to left).

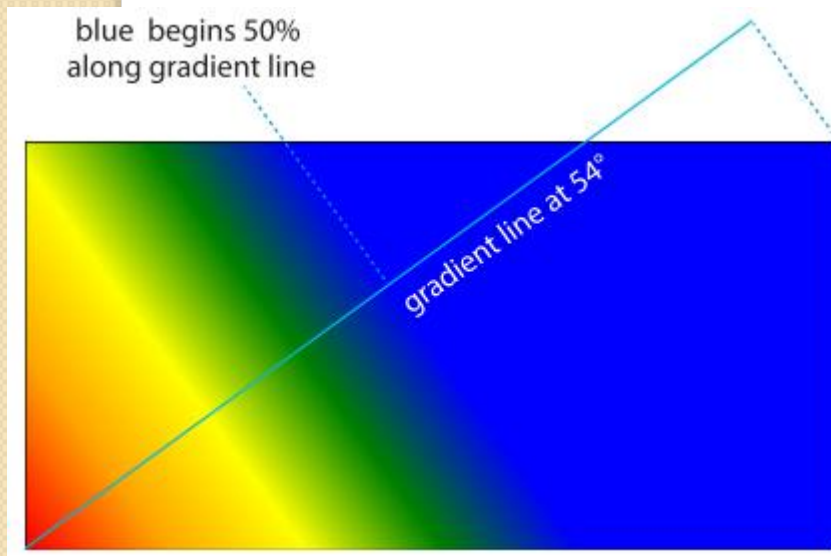
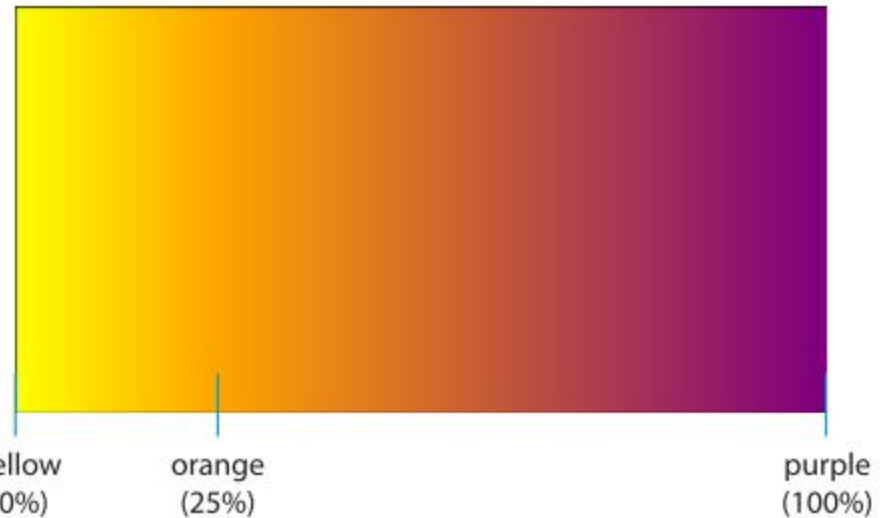
`background-image: linear-gradient(to bottom, aqua, green);`

linear-gradient(to bottom, aqua, green);



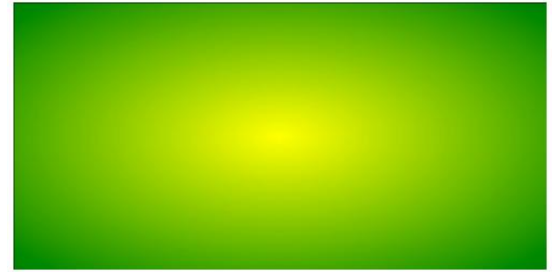
linear-gradient(90deg, yellow, orange 25%, purple);

linear-gradient(54deg, red, orange, yellow, green, blue 50%);



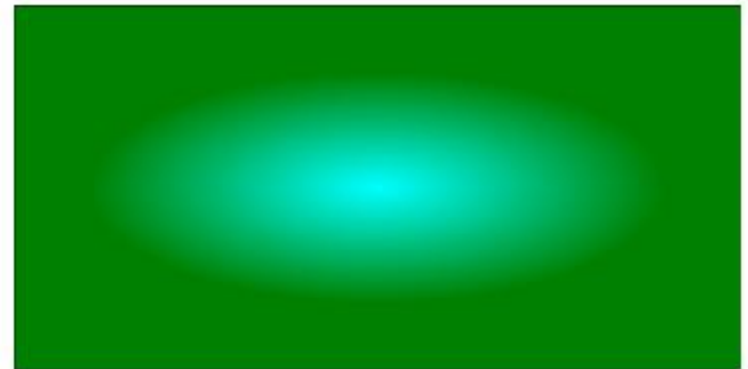
Radial Gradients

background-image: radial-gradient(yellow, green);



radial-gradient(200px 80px, aqua, green);

radial-gradient(farthest-side at right bottom, yellow, orange 50%, purple);



Repeating Gradients

- background: repeating-linear-gradient(to bottom, white, silver 30px);



- repeating-linear-gradient(45deg, orange, orange 12px, white 12px, white 24px);



BORDERS

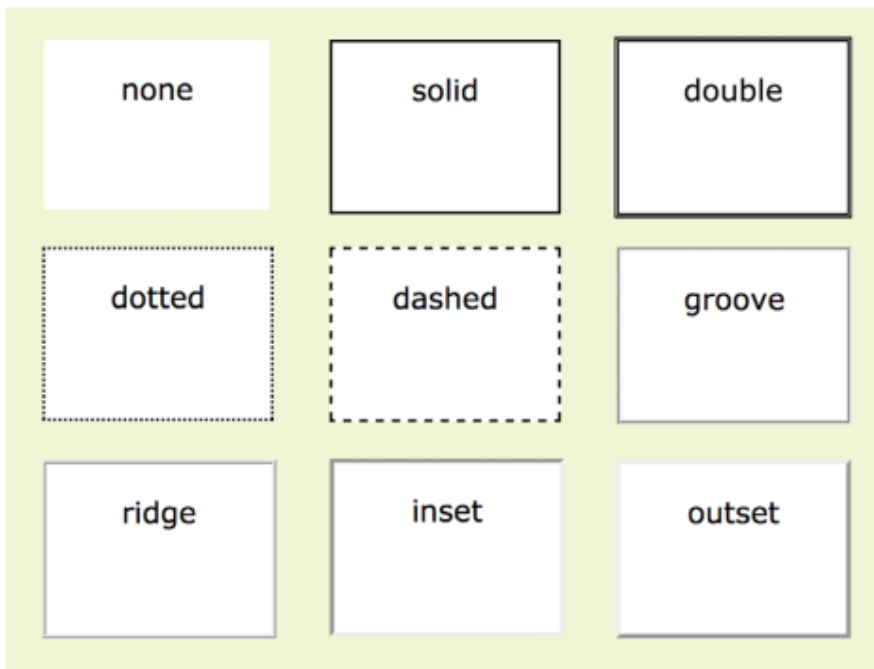
Border Style

`border-top-style`, `border-right-style`,
`border-bottom-style`, `border-left-style`

Values: none | solid | hidden | dotted | dashed | double | groove | ridge | inset | outset

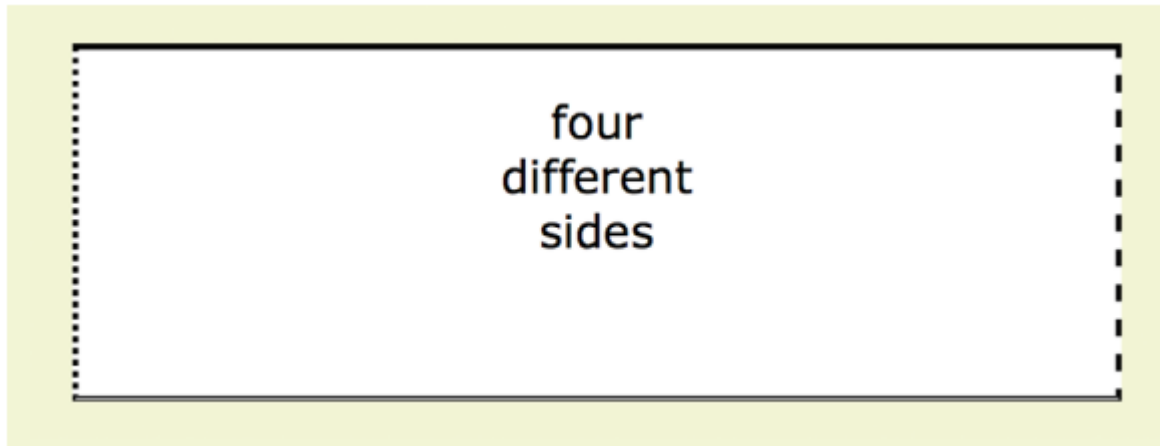
`border-style`

Values: none | solid | hidden | dotted | dashed | double | groove | ridge | inset | outset



```
div#silly {  
  border-top-style: solid;  
  border-right-style: dashed;  
  border-bottom-style: double;  
  border-left-style: dotted;  
  width: 300px;  
  height: 100px;  
}
```

A6o: border-style: solid dashed double dotted;



Border Width (Thickness)

`border-top-width`, `border-right-width`,
`border-bottom-width`, `border-left-width`

Values: `length` | `thin` | `medium` | `thick`

Default: `medium`

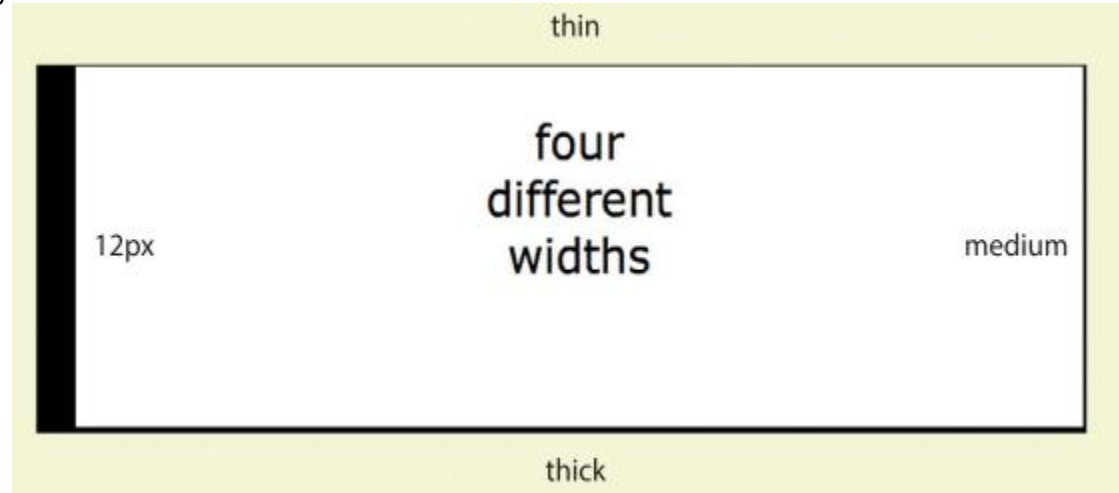
`border-width`

Values: `length` | `thin` | `medium` | `thick`

Default: `medium`

```
div#help {  
  border-top-width: thin;  
  border-right-width: medium;  
  border-bottom-width: thick;  
  border-left-width: 12px;  
  border-style: solid;  
  width: 300px;  
  height: 100px;  
}
```

```
div#help {  
  border-width: thin medium thick 12px;  
  border-style: solid;  
  width: 300px;  
  height: 100px;  
}
```

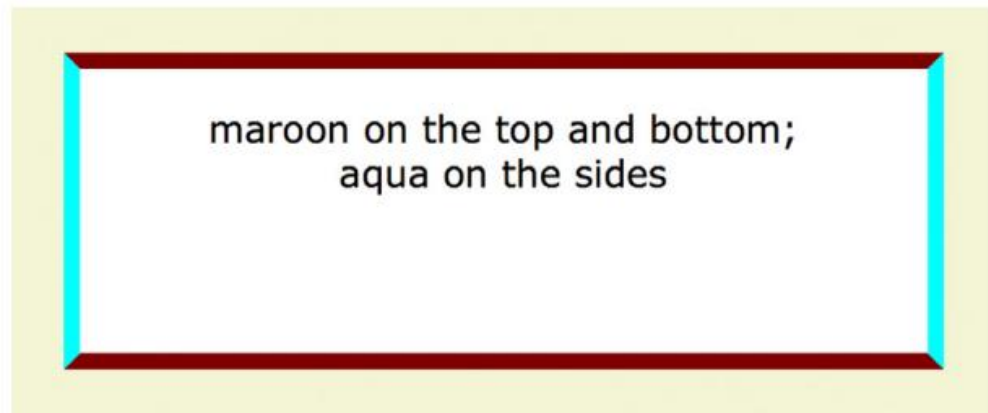


Border Color

`border-top-color`, `border-right-color`,
`border-bottom-color`, `border-left-color`

`border-color`

```
div#special {  
    border-color: maroon aqua;  
    border-style: solid;  
    border-width: 6px;  
    width: 300px;  
    height: 100px;  
}
```



Rounded Corners with border-radius

`border-top-left-radius`, `border-top-right-radius`,
`border-bottom-right-radius`, `border-bottom-left-radius`

Values: *length | percentage*

Default: 0

Applies to: all elements

`border-radius`

Values: 1, 2, 3, or 4 length or percentage values

```
p {  
width: 200px;  
height: 100px;  
background: darkorange;  
}
```

`border-radius: 1em;`



`border-radius: 50px;`



`border-top-right-radius: 50px;`



BOX DROP SHADOWS

box-shadow

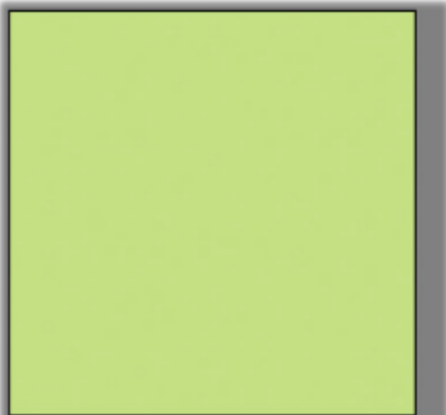
Values:

'horizontal offset' 'vertical offset' 'blur distance' 'spread distance' color
inset | none

A box-shadow: 6px 6px gray;

B box-shadow: 6px 6px 5px gray;

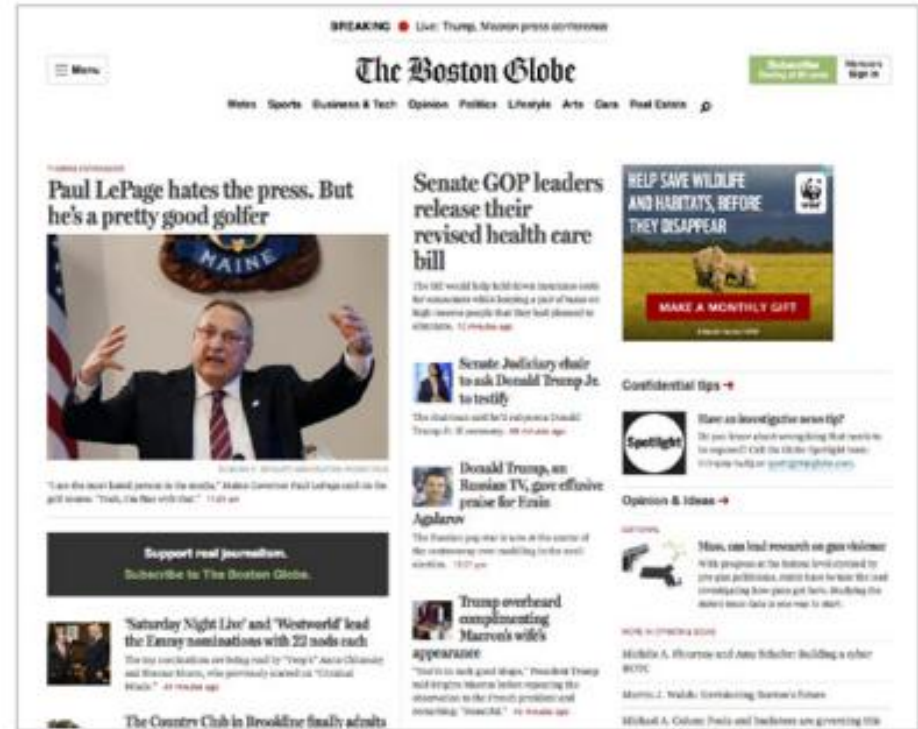
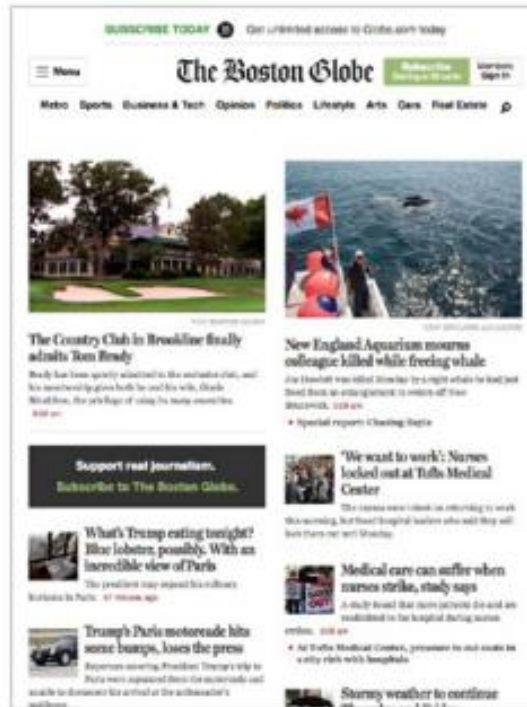
C box-shadow: 6px 6px 5px 10px gray;



Responsive Web Design

- Responsive Web Design (Адаптивний веб-дизайн, або RWD) - це дизайн та виробничий підхід, який дозволяє комфортно переглядати та використовувати веб-сайт на будь-яких пристроях.
- Основний принцип полягає в тому, що всі пристрої отримують одне і те ж джерело HTML, розташоване за **однією URL-адресою**, але застосовуються різні стилі на основі розміру вікна перегляду (viewport) для перестановки компонентів та оптимізації використання.

- The Boston Globe bostonglobe.com



Рецепт RWD

Методика має три основні компоненти:

- **A flexible grid** (Гнучка сітка)

Замість того, щоб залишатися на статичній ширині, чуйні сайти використовують методи, які дозволяють стискуватися та перетікати у доступний простір браузера.

- **Flexible images**

Зображення та інші вбудовані носії інформації повинні мати можливість масштабування відповідно до елементів, що містять їх.

- **CSS media queries**

Медіа-запити дають нам можливість доставляти набори правил лише на пристрої, які відповідають певним критеріям, наприклад ширині та орієнтації.

До цього списку можна ще додати **viewport meta** елемент, завдяки якому ширина веб-сторінки відповідає ширині екрана.

Setting the Viewport

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

Flexible Grids (Fluid Layouts)

У дискусіях про Flexbox та Grid у попередньому розділі ми бачили приклади елементів, які розширюються та стискаються для заповнення наявного простору їх контейнерів. Ця плинність - це саме та поведінка, яка вам потрібна, щоб вміст акуратно відповідав широкому діапазону розмірів вікна перегляду.

Гнучкі сітки є основою чуйного дизайну.



w3c.org

Making Images Flexible

```
img {  
max-width: 100%;  
}
```

In this book, Philadelphia Ice Creams, comprising the first group, are very palatable, but expensive. In many parts of the country it is quite difficult to get good cream. For that reason, I have given a group of creams, using part milk and part cream, but it must be remembered that it takes smart "juggling" to make ice cream from milk. By far better use condensed milk, with enough water or milk to rinse out the cans.

Ordinary fruit creams may be made with condensed milk at a cost of about fifteen cents a quart, which, of course, is cheaper than ordinary milk and cream.



If pure raw cream is stirred rapidly, it swells and becomes frothy, like the beaten whites of eggs, and is "whipped cream." To prevent this in making Philadelphia Ice Cream, one-half the cream is scalded, and when it is very cold, the remaining half of raw cream is added. This gives the smooth, light and rich consistency which makes these creams so different from others.

The time for freezing varies according to the quality of cream or milk or water; water ices require a longer time than ice creams. It is not well to freeze the mixtures too rapidly; they are apt to be coarse, not smooth, and if they are churned before the mixture is icy cold they will be greasy or

In this book, Philadelphia Ice Creams, comprising the first group, are very palatable, but expensive. In many parts of the country it is quite difficult to get good cream. For that reason, I have given a group of creams, using part milk and part cream, but it must be remembered that it takes smart "juggling" to make ice cream from milk. By far better use condensed milk, with enough water or milk to rinse out the cans.

Ordinary fruit creams may be made with condensed milk at a cost of about fifteen cents a quart, which, of course, is cheaper than ordinary milk and cream.



If pure raw cream is stirred rapidly, it swells and becomes frothy, like the beaten whites of eggs, and is "whipped cream." To prevent this in making Philadelphia Ice Cream, one-half the cream is scalded, and when it is very cold, the remaining half of raw cream is added. This gives the smooth, light and rich consistency which makes these creams so different from others.

In this book, Philadelphia Ice Creams, comprising the first group, are very palatable, but expensive. In many parts of the country it is quite difficult to get good cream. For that reason, I have given a group of creams, using part milk and part cream, but it must be remembered that it takes smart "juggling" to make ice cream from milk. By far better use condensed milk, with enough water or milk to rinse out the cans.

Ordinary fruit creams may be made with condensed milk at a cost of about fifteen cents a quart, which, of course, is cheaper than ordinary milk and cream.



If pure raw cream is stirred rapidly, it swells and becomes frothy, like the beaten whites of eggs, and is "whipped cream." To prevent this in making Philadelphia Ice Cream, one-half the cream is scalded, and when it is very cold, the remaining half of raw cream is added. This

Media Query Magic

- Тепер ми переходимо до справжнього рушія чуйного дизайну - медіа-запитів!
- **Медіа-запити застосовують різні стилі на основі характеристик браузера:** його ширину, вертикальну чи горизонтальну орієнтацію, роздільну здатність тощо.
- Саме вони дозволяють надсилати макет з одним стовпчиком на невеликі екрани, а розкладку з декількома стовпчиками - на більші екрани.
- **Структура медіа-запиту,** що використовується в таблиці стилів, виглядає так:

```
@media type and (feature: value) {  
  /* styles for browsers that meet this criteria */  
}
```

```
@media screen and (orientation: landscape) {  
  body {  
    background: skyblue;  
  }  
}
```

```
@media screen and (orientation: portrait) {  
  body {  
    background: coral;  
  }  
}
```



Media types

- Типи медіа, включені в першу частину запиту, були введені в CSS2 як спосіб орієнтації стилів на певні медіа. Наприклад, це правило

```
@media print {  
/* print-specific styles here */  
}
```

надає набір стилів лише тоді, коли документ надруковано.

Найновіші визначені типи носіїв інформації – це **all**, **print**, **screen** та **speech**.

Media feature queries (Запити медіа-функцій)

- CSS3-медіа-запити роблять націлювання на перевірку певної особливості (feature) вікна перегляду чи пристрою.
- Найпоширеніша особливість, яку слід перевірити, - це ширина вікна перегляду (viewport width). Ви також можете перевірити наявність мінімальної ширини (min-width) та максимальної ширини (max-width).

Приклад.

```
h1 {  
  font-family: Georgia, serif;  
}  
@media screen and (min-width: 40em) {  
  h1 {  
    font-family: 'Lobster', cursive;  
  }  
}
```

TABLE 17-1. Media features you can evaluate with media queries

Feature	Description
width	The width of the display area (viewport). Also min-width and max-width .
height	The height of the display area (viewport). Also min-height and max-height .
orientation	Whether the device is in portrait or landscape orientation.
aspect-ratio	Ratio of the viewport's width divided by height (width/height). Example: aspect-ratio: 16/9 .
color	The bit depth of the display; for example, color: 8 tests for whether the device has 8-bit color.
color-index	The number of colors in the color lookup table.
monochrome	The number of bits per pixel in a monochrome device.
resolution	The density of pixels in the device. This is increasingly relevant for detecting high-resolution displays.
scan	Whether a tv media type uses progressive or interlace scanning. (Does not accept min-/max- prefixes.)
grid	Whether the device uses a grid-based display, such as a terminal window. (Does not accept min-/max- prefixes.)

New in Media Queries Level 4

These features have been added in the Working Draft of MQ4. Some may gain browser support or be dropped from future drafts. I include them here to show you where the W3C sees media queries. See drafts.csswg.org/mediaqueries-4.

update-frequency	How quickly (if at all) the output device modifies the appearance of the document.
overflow-block	How the device handles content that overflows the viewport along the block axis.
overflow-inline	Whether the content that overflows the viewport along the inline axis is visible.
color-gamut	The approximate range of colors that are supported by the user agent.
pointer	Whether the primary input mechanism is a pointing device and whether it is used to activate elements.
hover	Whether the input mechanism allows the user to hover over elements.
any-pointer	Whether any available input mechanism is a pointing device, and whether it is used to activate elements.
any-hover	Whether any available input mechanism allows hovering.

Як використовувати медіа-запити

Ви можете використовувати медіа-запити в таблиці стилів або умовно завантажити зовнішні таблиці стилів. Медіа-запити не можна використовувати з вбудованими стилями.

- **Within a style sheet**

Найпоширеніший спосіб використання медіа-запитів - це використання правила `@media` прямо в таблиці стилів.

- **With external style sheets**

Для великих або складних сайтів розробники можуть вибирати стилі для різних пристроїв в окремі таблиці стилів і викликати весь файл `.CSS`, коли дотримуються певні умови. Один з методів - використовувати атрибут `media` в елементі `link` для умовного завантаження окремих файлів `.CSS`.

Приклад.

```
<head>  
  <link rel="stylesheet" href="styles.css">  
  <link rel="stylesheet" href="2column-styles.css"  
    media="screen and (min-width:1024px)">  
</head>
```

Аналогічно, ви можете здійснювати медіа-запити за допомогою правил `@import`, які витягують зовнішні таблиці стилів із таблиці стилів. Зауважте, що слово "media" не відображається в цьому синтаксисі, а лише тип та запит.

```
<style>  
@import url("/default-styles.css");  
@import url("/wide-styles.css") screen and (min-width:  
  1024px);  
/* other styles */  
</style>
```


Browser support

- Медіа-запити не підтримує IE8 на Windows XP (а це 1-2% користувачів)
- Якщо ви очікуєте відвідувачів, які використовують старі версії IE, у вас є пара варіантів. По-перше, ви можете скористатися поліфайлом Respond.js, який додає підтримку min-width та max-width для не підтримуючих браузерів. Він був створений Скоттом Джелом та доступний на веб-сайті github.com/scottjehl/Respond.
- Інший варіант - створити окремих аркуш стилів з не надмірною компонованням робочого столу та доставити його лише користувачам з IE8 або раніше, використовуючи умовний коментар. Інші веб-переглядачі ігнорують вміст цього коментаря, що стосується IE:

```
<!-- [if lte IE 8]>  
  <link rel="stylesheet" href="/path/IE_fallback.css">  
<![endif]-->
```

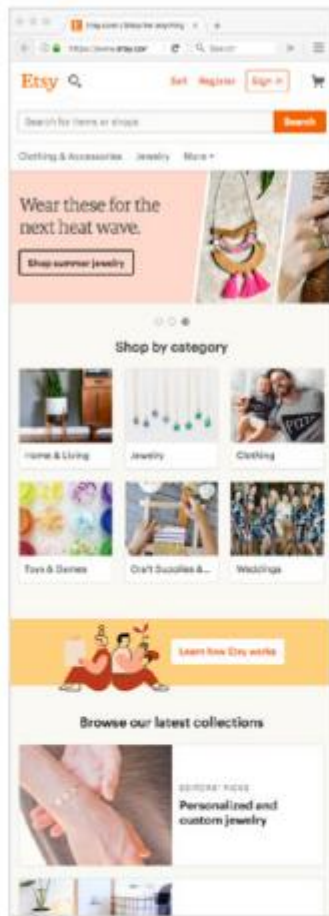
CHOOSING BREAKPOINTS

- Точка перерви - це точка, в якій ми використовуємо медіа-запит для введення зміни стилю. Коли ви задаєте `min-width:800px` пікселів у медіа-запиті, ви говорите, що 800 пікселів - це "точка розриву", при якій слід використовувати ці конкретні стилі. На малюнку показані деякі точки перерви, на яких Etsy.com робить як основні зміни макета, так і тонкі зміни дизайну на своїй домашній сторінці.

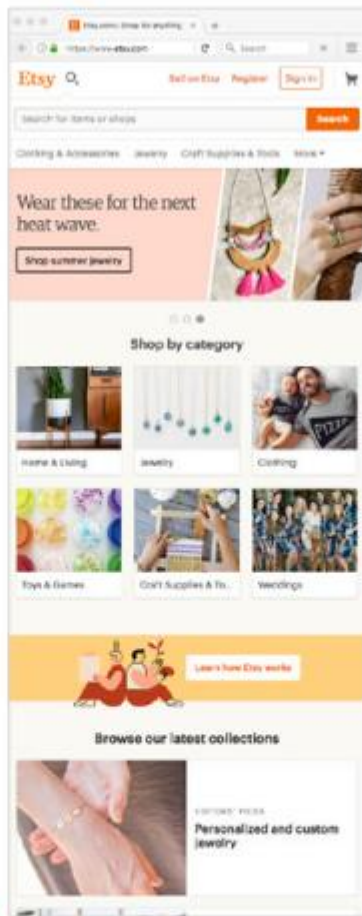
(Дивись Learning Web Design 5th 2018, p.496)



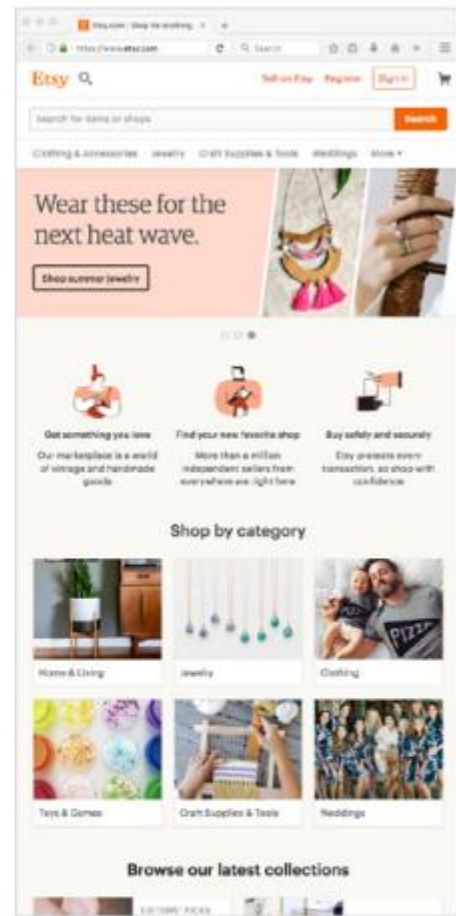
At the **480-pixel breakpoint**, the category navigation



At **501 pixels**, “Sell” becomes “Sell on Etsy” (a very subtle adjustment). You



At **640 pixels**, the “How Etsy Works” images and messages move above the categories. In smaller views,



EXERCISE 17-1. Making the bakery home page responsive (Learning Web Design 5th 2018, p.508)




MENU NEWS
ABOUT CONTACT



BLACK GOOSE BAKERY

Fresh from the Oven

BREADS



Our breads are made daily from highest-quality whole grain flour, water, salt, and yeast or sourdough starter. Simply and naturally, and never any preservatives. Patience is key to achieving the

chocolate.
made from scratch
each day. Stop by to see our
seasonal muffin flavors!

[LEARN MORE ABOUT HOW WE MAKE OUR MUFFINS...](#)

Hours

MONDAY: 5am to 3pm
TUESDAY: 5am to 3pm
WEDNESDAY: 5am to 3pm
THURSDAY: 5am to 3pm
FRIDAY: 5am to 3pm
SATURDAY: 6am to 4pm
SUNDAY: 6am to 4pm

All content copyright © 2017, Black Goose Bistro.



Трансформації, переходи та анімація

- **Трансформацією** (Transform, перетворення) вважається операція зсуву елемента в положення, відмінне від займаного за замовчуванням.
- В CSS операція **переходу** (Transition) визначається як анімація стану елемента внаслідок тимчасової зміни значень одного або декількох властивостей. Зазвичай перехід виконується у відповідь на певну дію з боку користувача.
- **Анімація** до CSS3 здійснювалась тільки з використанням JavaScript або Flash. Тепер це можна здійснити за рахунок властивостей CSS3.

Переходи (Transitions)

- Почнемо з переходів CSS, чудового способу зміни стилів, що плавно переходять з одного в інший.
- Потім ми обговоримо перетворення CSS для перестановки, масштабування, обертання та перекосу елементів та розглянемо, як можна анімувати їх переходами.

transition-property

Values: `property-name | all | none`

Default: `all`

Applies to: `all elements, :before and :after pseudo-elements`

transition-duration

Values: `time`

Default: `0s`

transition-timing-function

Values: `ease | linear | ease-in | ease-out | ease-in-out | step-start | step-end | steps | cubic-bezier(##,##,##,##)`

Default: `ease`

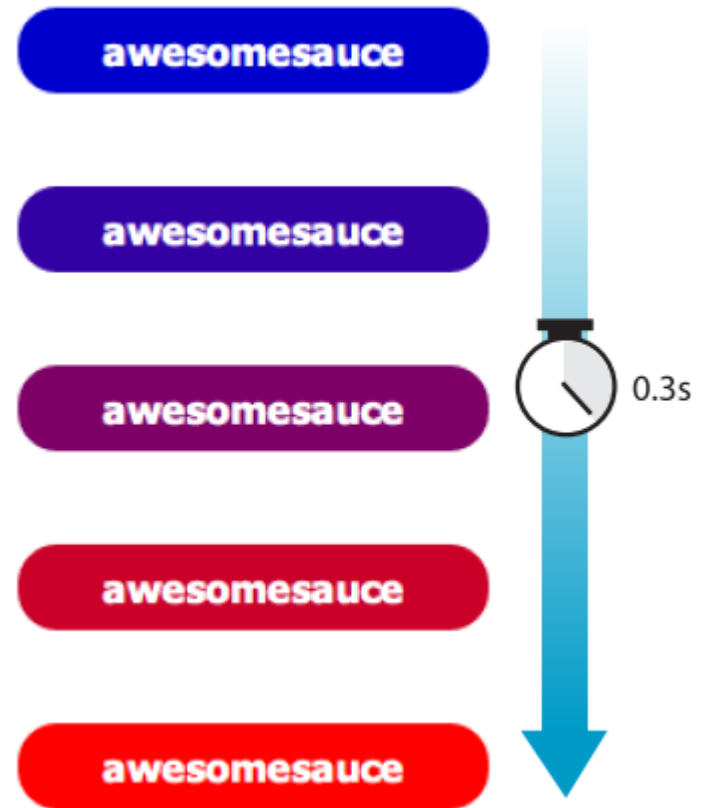
THE MARKUP

```
<a href="..."  
  class="smooth">awesomesauce</a>
```

THE STYLES

```
.smooth {  
  display: block;  
  text-decoration:none;  
  text-align: center;  
  padding: 1em 2em;  
  width: 10em;  
  border-radius: 1.5em;  
  color: #fff;  
  background-color: mediumblue;  
  transition-property: background-color;  
  transition-duration: 0.3s;  
}
```

```
.smooth:hover, .smooth:focus {  
  background-color: red;  
}
```



EXERCISE 18-1. Trying out transitions

(Learning Web Design 5th 2018, p.525)

Normal state.



:hover, :focus

The background and border colors change.



:active

Link appears to be pressed down.



Перетворення (Transforms)

- CSS3 Transforms Module (www.w3.org/TR/css-transforms-1) дає авторам можливість обертати, переміщувати, змінювати розміри та перекосувати елементи HTML у дво- та тривимірному просторі.



rotate()



translate()



scale()



skew()

Transforming the Angle (rotate)

```
img {  
  width: 400px;  
  height: 300px;  
  transform: rotate(-10deg);  
}
```



```
transform: rotate(-10deg);
```

transform-origin

Values: *percentage* | *length* | left | center | right | top | bottom

Default: 50% 50%

Applies to: transformable elements

transform-origin: center top;

transform-origin: 50%, 0%;

transform-origin: 200px, 0;



transform-origin: center top;



transform-origin: 100% 100%;



transform-origin: 400px 0;

Transforming the Position (translate)

```
transform: translateX(50px);
```

```
transform: translateY(25px);
```

```
transform: translate(50px, 25px);
```



```
transform: translate(90px, 60px);
```



```
transform: translate(-5%, -25%);
```

Transforming the Size (scale)

```
a img {  
  transform: scaleX(1.5);  
}
```



```
a img {  
  transform: scale(2, .5);  
}
```

```
transform: scale(.75);
```



```
transform: scale(1.25);
```

```
transform: scale(1.5, .5);
```

Making It Slanty (skew)

```
a img {  
  transform: skewX(15deg);  
}
```

```
a img {  
  transform: skewY(30deg);  
}
```

```
a img {  
  transform: skew(15deg,  
    30deg);  
}
```



transform: skewX(15deg);



transform: skew(15deg, 30deg);



Запоріжж **transform: skewY(30deg);**
кафедра ПЗАС, доц. Попівций В.І., 2020

Applying Multiple Transforms

```
img:hover, img:focus {  
  transform: scale(1.5) rotate(-5deg) translate(50px,30px);  
}
```

Normal state



:hover, :focus

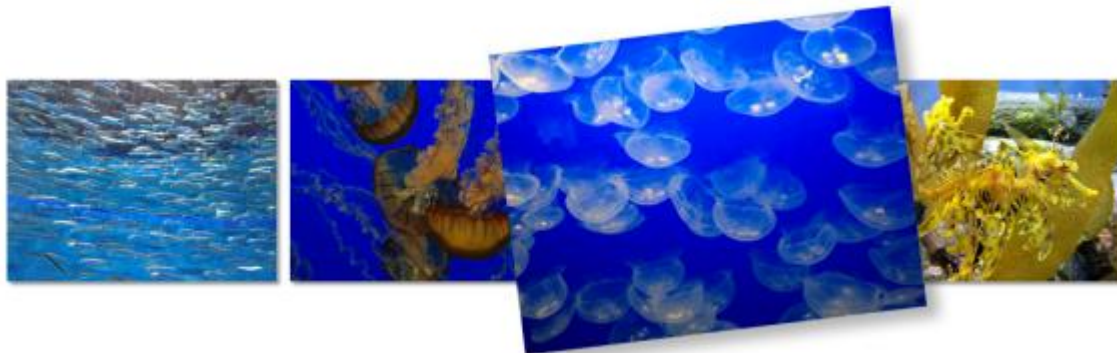
rotate(), translate(), and scale() applied



EXERCISE 18-2. Transitioning transforms

(Learning Web Design 5th 2018, p.532)

```
a:hover #img1, a:focus #img1 {  
  transform: rotate(-3deg);  
}  
a:hover #img2, a:focus #img2 {  
  transform: rotate(5deg);  
}  
a:hover #img3, a:focus #img3 {  
  transform: rotate(-7deg);  
}  
a:hover #img4, a:focus #img4 {  
  transform: rotate(2deg);  
}
```



KEYFRAME ANIMATION

Приклади.

1) How I Learned to Walk by Andrew Wang-Hoyer
<http://andrew.wang-hoyer.com/experiments/walking/>



2) Animated Web Banner by Caleb Jacob
<http://tympanus.net/codrops/2012/01/10/animated-web-banners-with-css3/>

3) Adorable animated submarine by Alberto Jerez
<https://codepen.io/ajerez/pen/EaEEOW>



Встановлення ключових кадрів

Процес анімації складається з двох частин:

1. Встановіть ключові кадри за допомогою правила **@keyframes**.
2. Додайте властивості анімації до елементів, які будуть анімовані.

```
@keyframes colors {  
  0% { background-color: red; }  
  20% { background-color: orange; }  
  40% { background-color: yellow; }  
  60% { background-color: green; }  
  80% { background-color: blue; }  
  100% { background-color: purple; }  
}  
  
@keyframes animation-name {  
  keyframe { property: value; }  
  /* additional keyframes */  
}
```



Додавання властивостей анімації

```
<div id="magic">Magic!</div>
```

```
#magic {
```

```
...
```

```
animation-name: colors;
```

```
animation-duration: 5s;
```

```
animation-timing-function: linear;
```

```
animation-iteration-count: infinite;
```

```
animation-direction: alternate;
```

```
}
```

Або

```
#magic {
```

```
animation: colors 5s linear infinite alternate;
```

```
}
```