Cascading Style Sheets (CSS)

National Telecommunication Institute - Egypt

Introduction to CSS

- CSS stands for Cascading Style Sheets.
- Styles define how to display HTML elements
- There are three ways to apply CSS to HTML:
  - ✓ In-Line Styles
  - ✓ Internal Styles
  - ✓ External Styles

Introduction to CSS

In-Line Styles

Styles are directly inserted into the HTML tags and applied only on this element

Syntax:

```
<tag style="property: value;">text</tag>
```

Introduction to CSS

Internal Styles

Styles are inserted into the head of an HTML page and applied only on this page.

Syntax:

```
<html><head><title>CSS Example</title>
<style>
        Styles Goes here
</style>
</head>
```

Introduction to CSS

External Styles

Styles are created in a separate.css file and applied to all the website pages.

Syntax:

```
Styles Goes here
```

Creating External CSS Style

Creating External CSS Style

- To create an External CSS Style:
  - ✓ Create new file with CSS extension (.css)
  - ✓ Open the file for editing.
  - ✓ Attach the file to the HTML file.
  - ✓ Add the CSS rules to the file.

CSS Syntax

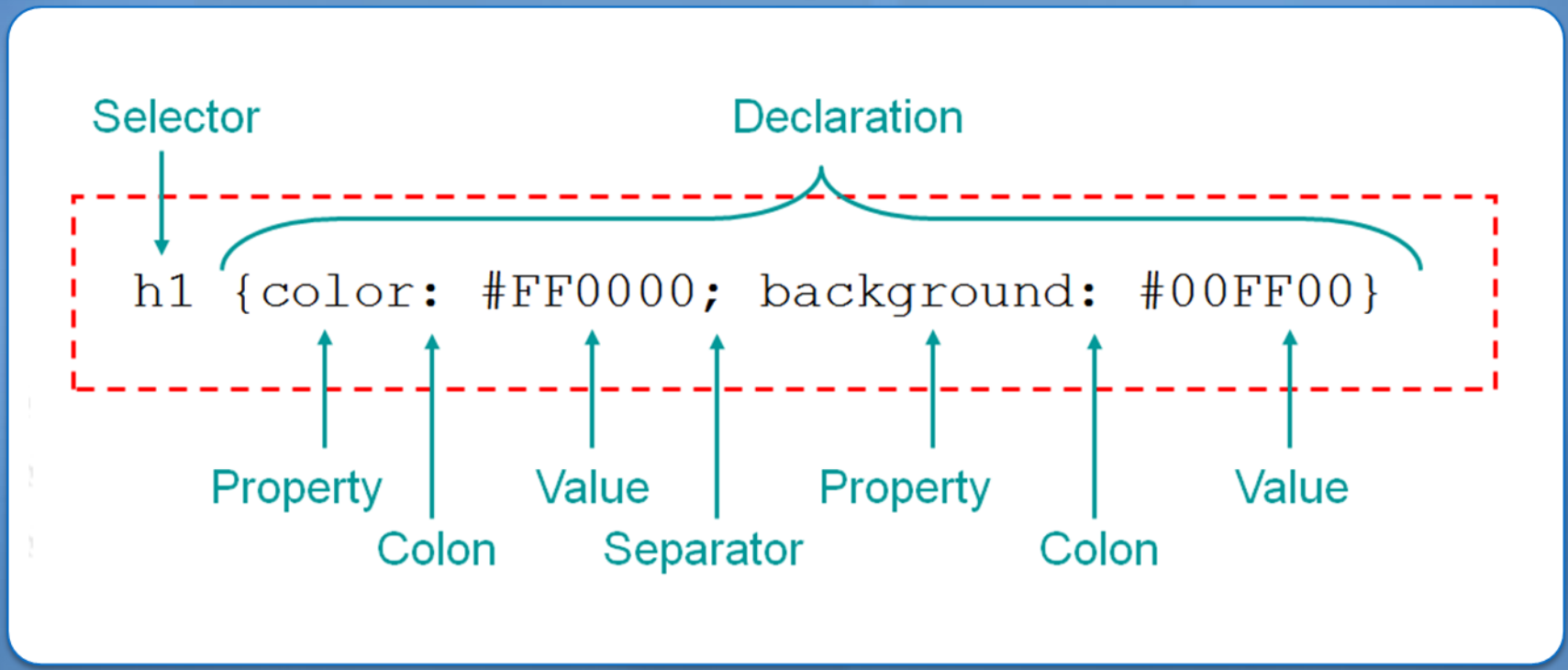
CSS Syntax

A CSS rule (syntax) is composed of:

- ✓ **Selector:** is the HTML tag to be styled.
- ✓ **Property:** is the style attribute.
- ✓ **Value:** is the value of the property.

CSS Syntax

A CSS rule (syntax) is composed of:



Selectors, Properties, and Values

There are 3 types of CSS Selectors

- ✓ Tag Selector
- ✓ Class Selector
- ✓ Id Selector

Selectors, Properties, and Values

Tag Selector

The tag selector selects elements based on the element tag.

Syntax:

```
tag
        {
          property: value;
        }
```

Ex:

```
h1
        {
          color: red;
        }
```

Selectors, Properties, and Values

Class Selector

The class selector selects elements with a specific class attribute

Syntax:

```
.classname
        {
          property: value;
        }
```

Ex:

```
.intro
{
color:red;
}
```

Selectors, Properties, and Values

Id Selector

The id selector uses the id attribute of an HTML element to select an element.

Syntax:

```
#id
        {
          property: value;
        }
```

Ex:

```
#firstname
{
color:red;
}
```

Cascading Order

Cascading Order

- ✓ What style will be used when there is more than one style specified for the same HTML element?
  1. Inline style (inside an HTML tag).
  2. Internal style sheets (inside the head).
  3. External style sheets (CSS separate file).
  4. HTML Attribute.
  5. Browser default.

CSS Background

CSS Background

Background Color

Description: specifies the background color of an element

Property: { background-color:#b0c4de; }

Values: #ff0000 | rgb(255,0,0) | red | transparent

Background Image

Description: specifies the background image of an element

Property: { background-image:url("imge1.gif"); }

Values: image path | none

CSS Background

Background Repeat

Description: specifies to repeat the background or not

Property: { background-repeat:repeat; }

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

Background attachment

Description: sets the background image fixed or scrolls

Property: { background-attachment:scroll; }

Values: scroll | fixed

CSS Background

Background position

Description: sets the starting position of a background image

Property: { background-position:center center; }

Values: left top | left center | left bottom |

right top | right center | right bottom |

center top | center center | center bottom |

0% 0% | 10px 200px

CSS Background

Background ➔ Shorthand Property

Description: sets all the properties of a background in one declaration

Property: { background: color image position repeat attachment ; **}**

Ex: background: #000 url(1.jpg) repeat left bottom fixed;

CSS Colors

CSS Colors

- Colors in CSS are specified as:
  - Color Names. (supports 140 standard color)
    Ex: Red
  - RGB values. Ex: rgb(255, 99, 71)
  - HEX values. Ex: #ff6347
  - HSL values. Ex: hsl(9, 100%, 64%)
  - RGBA values. Ex: rgba(255, 99, 71, 0.5)
  - HSLA values. Ex: hsla(9, 100%, 64%, 0.5)

CSS Text

CSS Text

Text Color

Description: used to set the color of the text

Property: { color:#b0c4de; }

Values: #ff0000 | rgb(255,0,0)

Text Align

Description: used to set the horizontal alignment of a text

Property: {text-align:right; }

Values: center | left | right | justify

CSS Text

Text Indentation

Description: specify the indentation of the first line of a text

Property: {text-indent:50px; }

Values: in pixels | 0

Text Transformation

Description: used to specify case of letters in a text

Property: {text-transform:uppercase; }

Values: capitalize | lowercase | uppercase | none

CSS Text

Text decoration

Description: used to set or remove decorations from text

Property: {text-decoration:none; }

Values: none | underline | overline | line-through

Text Direction

Description: specifies the text direction/writing direction

Property: {direction:rtl; }

Values: ltr | rtl

CSS Text

Letter Spacing

Description: Defines an extra space between characters

Property: {letter-spacing:2px; }

Values: in pixels | normal

Line Height

Description: specifies the line height

Property: {line-height:30px; }

Values: In Pixels | normal

CSS Text

Word Spacing

Description: Increases or decreases the space between words

Property: {word-spacing:30px; }

Values: In Pixels | normal

CSS Font

CSS Font

Font Family

Description: Specifies the font family for text

Property: { font-family:"Times New Roman",Georgia,Serif; }

Values: family-name

Font Size

Description: Specifies the font size of text

Property: {font-size:25px; }

Values: percent| pixels

CSS Font

Font Style

Description: Specifies the font style for text

Property: { font-style:italic; }

Values: normal | italic | oblique

Font weight

Description: Specifies the weight of a font

Property: { font-weight:bold; }

Values: normal | bold | bolder | lighter

CSS Font

Font ➔ Shorthand Property

Description: sets all the properties of a font in one declaration

Property: { font: size weight style family; }

Ex: { font: 13px bold italic Arial, Helvetica, sans-serif; }

CSS Comments

CSS Comments

- Comments are used to explain the code, and may help when you edit the source code at a later time.
- Comments are ignored by browsers.
- A CSS comment starts with /* and ends with */.

CSS Links

CSS Links

- Links can be styled with any CSS property (e.g. color, font-family, background, etc.)
- links can be styled differently depending on their state.

CSS Links

- The four links states are:
  - ✓ a:link - a normal, unvisited link
  - ✓ a:visited - a link the user has visited
  - ✓ a:hover - a link when the user mouses over it
  - ✓ a:active - a link the moment it is clicked
- The order of the four state is case sensitive

CSS Links

Example

```
a:link {color:#FF0000;}

a:visited {color:#00FF00;}

a:hover {color:#FF00FF;}

a:active {color:#0000FF;}
```

CSS Lists

CSS Lists

list-style-image

Description: Specifies an image as the list-item marker

Property: { list-style-image:url('img1.gif'); }

Values: image path | none

list-style-type

Description: Specifies the type of list-item marker

Property: { list-style-type:circle; }

Values: disc | circle | square | decimal | none |
decimal-leading-zero | lower-alpha | lower-roman |
upper-alpha | upper-roman

CSS Lists

list-style-position

Description: specifies the position of bullets.

Property: { list-style-position:inside; }

Values: outside | inside

list-style ➔ Shorthand Property

Description: Sets all the list properties in one declaration

Property: {

list-style:list-style-typelist-style-position list-style-image; }

Ex: list-style: square inside url("sqpurple.gif")

CSS Tables

CSS Tables

caption-side

Description: Specifies the position of a table caption

Property: { caption-side:bottom; }

Values: top | bottom

border-collapse

Description: Sets whether the table borders are collapsed into a single border or detached as in standard HTML

Property: {border-collapse:collapse; }

Values: collapse | separate

CSS Tables

border-spacing

Description: sets the distance between the borders of adjacent cells. (needs border-collapse:separate)

Property: { **border-spacing**:h-space v-space; }

Values: h-sapace: horizontal spacing in pixels.

v-space: vertical spacing in pixels.

Single value: for both in pixel ➔ default: 2px

Ex: { **border-spacing**:5px 5px; }

{ **border-spacing**:3px; }

CSS Tables

table-layout

Description: Specify that the column width should increase or not to fit the content size.

Property: { table-layout:fixed; }

Values: auto: Column width fits with the content size.

Fixed: Column width will not change (fixed).

Default Value: auto

CSS Tables

empty-cells

Description: hides the borders and background of empty cells in a table. (needs border-collapse:separate)

Property: { empty-cells:hide; }

Values: Show | hide

Default Value: show

CSS Dimensions

CSS Dimensions

Width

Description: Sets the width of an element

Property: { width:500px; }

Values: value in pixels | percentage | auto

Height

Description: Sets the height of an element

Property: { height:500px; }

Values: value in pixels | percentage | auto

CSS Dimentions

max-height

Description: used to set the maximum height of an element.

Property: { max-height:500px; }

Values: value in pixels | percentage | none

min-height

Description: defines the minimum height of an element

Property: { min-height:500px; }

Values: value in pixels | percentage | 0

CSS Dimentions

max-width

Description: used to set the maximum width of an element.

Property: { max-width:500px; }

Values: value in pixels | percentage | none

min-width

Description: defines the minimum width of an element

Property: { min-width:500px; }

Values: value in pixels | percentage | 0

CSS Overflow

CSS Overflow

overflow

Description: specifies what happens if content overflows an element's box.

Property: { overflow:hidden; }

Values: visible | hidden | scroll | auto

Default Value: visible

Ex:

```
div {
    overflow: auto;
}
```

CSS Opacity

Opacity

Description: The opacity property sets the transparency level for an element, it accepts values from 0.0 to 1.0. The lower value makes the element more transparent.

Property: { opacity:0.5; }

Values: Number between 0.1 and 1.0

Default Value: 1

CSS Opacity

Ex: Image Transparency - Hover Effect

```
img
        {
        opacity:0.4;
        }
img:hover
        {
        opacity:1.0;
        }
```

Box Model

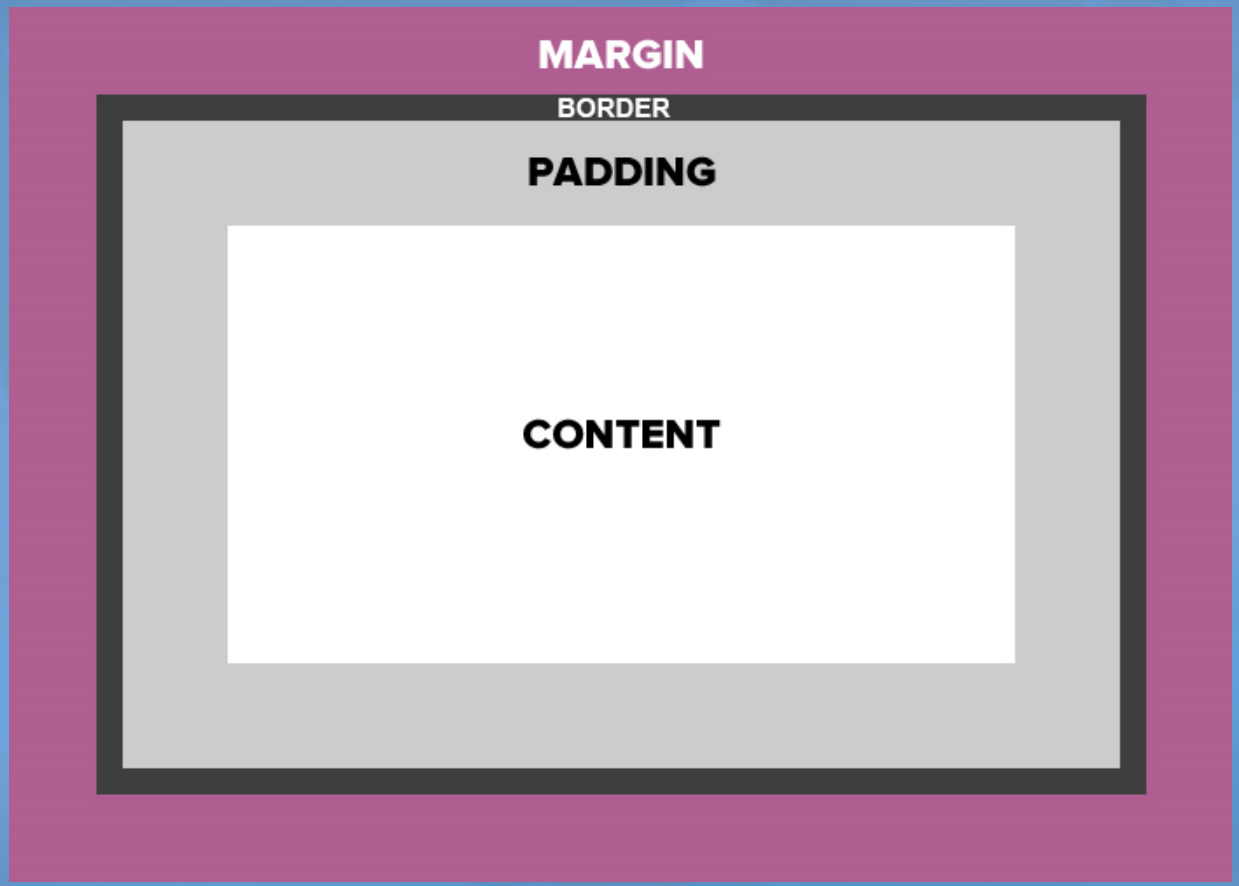
Box Model

- ✓ In CSS, the term "box model" is used when talking about design and layout.
- ✓ All HTML elements is considered as boxes.
- ✓ Box Model consists of: margins, borders, padding, and the content.
- ✓ The box model allows to place borders and spaces between elements.

Box Model

- ✓ **Margin:** The area between the border of an element and the parent element's border. it is completely transparent.
- ✓ **Border:** A border that goes around the padding and the content.
- ✓ **Padding:** The area between the content and the border. The padding is affected by the background color of the box
- ✓ **Content:** The content of the box, where text and images appear.

Box Model



CSS Borders

CSS Borders

Border Style

Description: used to set the style of the four borders

Property: { border-style:solid; }

Values: none| dotted | dashed | solid | double | hidden

Border Color

Description: used to set the color of the four borders

Property: { border-color:#ff0000; }

Values: #ff0000 | rgb(255,0,0) | red

CSS Borders

border-width

Description: used to set the width of the four borders

Property: { border-width:3px; }

Values: medium | thin | thick | pixels

border

Description: Sets all the border properties in one declaration

Property: { border:5px solid red; }

Values: border-width border-style border-color

CSS Individual Border

Individual Border

Description: used to set Individual Border of an element

Properties:

{border-left:6px solid red; }

{border-bottom:6px solid red; }

{border-top:6px solid red; }

{border-right:6px solid red; }

CSS Rounded Borders

border-radius

Description: used to add rounded borders to an element

Property:

{border-radius:6px;}

Default Value: 0

CSS Rounded Borders

Individual border-radius

Description: used to add rounded border to one corner of an element

Property:

{border-bottom-left-radius:6px; }

{border-bottom-right-radius:6px; }

{border-top-left-radius:6px; }

{border-top-right-radius:6px; }

CSS Margins

CSS Margins

Margin Top

Description: Sets the top margin of an element

Property: { margin-top:5px; }

Values: value in pixels | 0

Margin Bottom

Description: Sets the bottom margin of an element

Property: { margin-bottom:5px; }

Values: value in pixels | 0

CSS Margins

Margin Left

Description: Sets the left margin of an element

Property: { margin-left:5px; }

Values: value in pixels | 0

Margin Right

Description: Sets the right margin of an element

Property: { margin-right:5px; }

Values: value in pixels | 0

CSS Margins

Margin

Description: Sets the 4 margins in one declaration

Property: { margin:top right bottom left; }

Ex: { margin:10px 5px 15px 20px; }

Default Value: 0

Ex: margin:10px 5px 15px 20px; ➔ (T,R,B,L)

margin:10px 5px 15px; ➔ (T,R and L,B)

margin:10px 5px; ➔ (T and B,R and L)

margin:10px; ➔ (All)

CSS Margins

Margin Collapse:

- ✓ Top and bottom margins of elements are collapsed into a single margin that is equal to the largest of the two margins.
- ✓ This does not happen on left and right margins. Only top and bottom margins.

CSS Padding

CSS Padding

Padding Top

Description: Sets the top padding of an element

Property: { padding-top:5px; }

Values: value in pixels | 0

Padding Bottom

Description: Sets the bottom padding of an element

Property: { padding-bottom:5px; }

Values: value in pixels | 0

CSS Padding

Padding Left

Description: Sets the left padding of an element

Property: { padding-left:5px; }

Values: value in pixels | 0

Padding Right

Description: Sets the right padding of an element

Property: { padding-right:5px; }

Values: value in pixels | 0

CSS Padding

Padding

Description: Sets the 4 paddings in one declaration

Property: { padding:top right bottom left; }

Ex: { padding:10px 5px 15px 20px; }

Default Value: 0

Ex: padding:10px 5px 15px 20px; ➔ (T,R,B,L)

padding:10px 5px 15px; ➔ (T,R and L,B)

padding:10px 5px; ➔ (T and B,R and L)

padding:10px; ➔ (All)

CSS Outline

CSS Outline

- ✓ An outline is a line that is drawn around elements, outside the borders.
- ✓ The outline is drawn around the element's border, and may overlap other content.
- ✓ The outline is NOT a part of the element's dimensions; the element's total width and height is not affected by the width of the outline.

CSS Outline

Outline Properties

- **outline-style:** specifies the style of the outline
- **outline-color:** used to set the color of the outline
- **outline-width:** specifies the width of the outline
- **outline-offset:** adds space between the outline and the border of an element
- **Outline:** is a shorthand property for outline-width, outline-style and outline-color.

CSS Outline

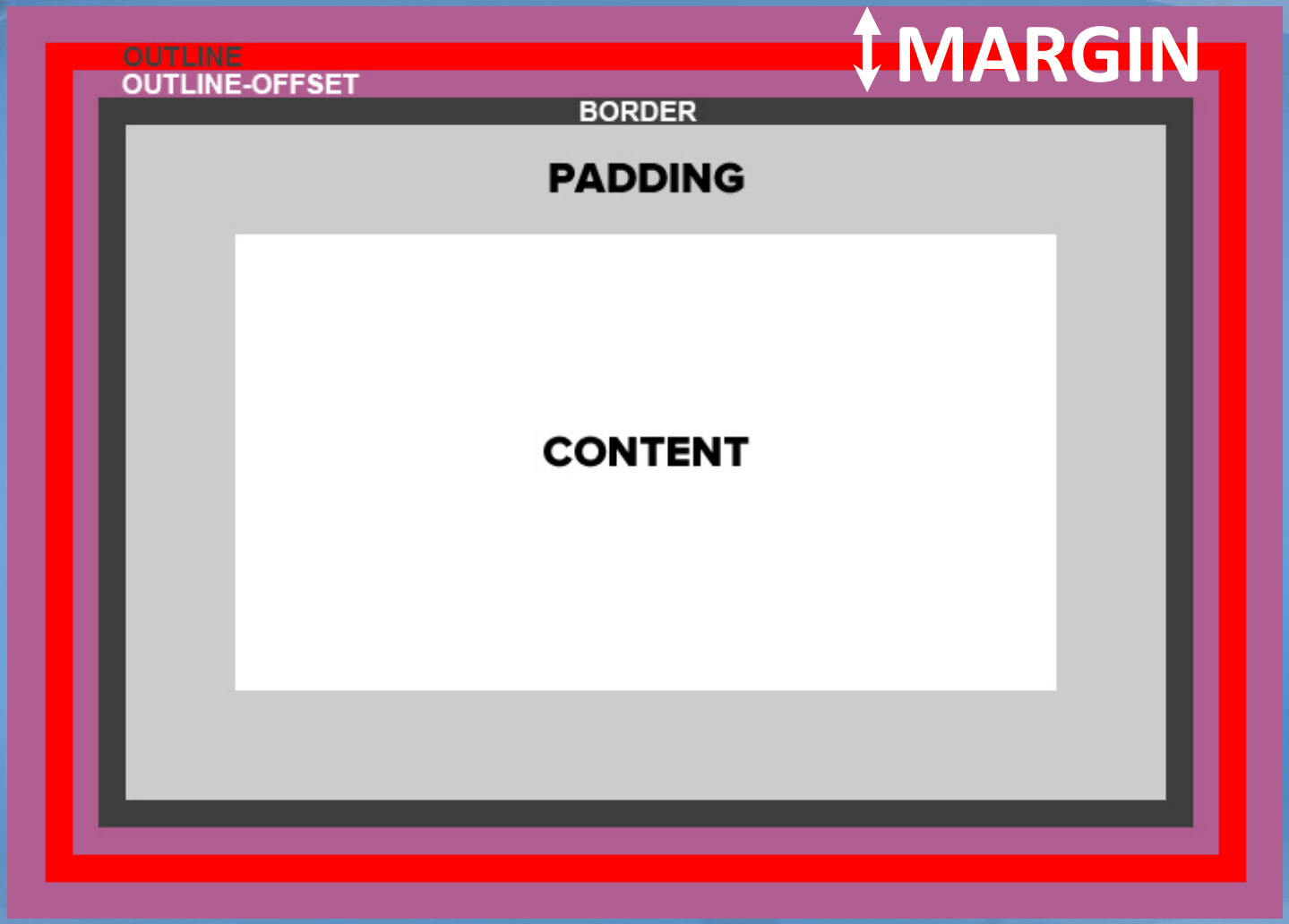
Ex:

```
{       outline: 10px dashed blue;
        outline-offset: 15px;
}
```

Same as:

```
{       outline-width: 10px;
        outline-style: dashed;
        outline-color: blue;
        outline-offset: 15px;
}
```

CSS Outline



CSS Float

CSS Float

- ✓ With CSS float, an element can be fly to the left or right corner of the parent.
- ✓ The other elements will wrap around it.
- ✓ Float is used specially with images and layouts.
- ✓ Elements are floated horizontally
- ✓ The elements before the floating element will not be affected.

CSS Float

Example

```
img
        {
                float:right;
        }
```

Float Clearing

Float Clearing

- ✓ Elements after the floating element will flow around it.
- ✓ To avoid this, use the clear property.
- ✓ The clear property specifies which sides of an element not to flow.

Float Clearing

Clear Property

Property: { clear:right; }

Values:

- **right:** don't allow other element to flow to right
- **left:** don't allow other element to flow to left
- **both:** don't allow other element to flow to left or right

Block and Inline Elements

Block and Inline Elements

A block element

- ✓ is an element that takes up the full width available, and has a line break before and after it.
- ✓ Examples of block elements:
  - ✓ <h1>
  - ✓ <p>
  - ✓
  - ✓

Block and Inline Elements

An inline element

- ✓ Is an element that only takes up as much width as necessary, and does not force line breaks.
- ✓ Does not allow to set a width and height on the element.
- ✓ Top and Bottom margins/paddings are not respected.
- ✓ Examples of inline elements:
  - <a>

Block and Inline Elements

Changing How an Element is Displayed

- ✓ Changing an **inline element** to a **block element**, or **vice versa**, is useful for creating new layouts.

Ex

```
li {display:inline;}
```

Ex

```
span {display:block;}
```

The inline-block Elements

An inline-block element

- ✓ Inline-block allows to set a width and height on the element.
- ✓ Inline-block respects Top/Bottom margins and paddings.
- ✓ Inline-block does not add line-break after the element, so one element can sit next to another.

The inline-block Elements

Ex

```
div {
        display: inline-block;
        width: 70px;
        height: 70px;
        padding: 15px;
        border: 1px solid blue;
        background-color: yellow;
      }
```

CSS Display and Visibility

CSS Display and Visibility

Hiding an element

- ✓ Hiding an element can be done by setting the **display** property to "**none**" or the **visibility** property to "**hidden**". However, notice that these two methods produce different results.

CSS Display and Visibility

- ✓ **visibility:hidden** → hides an element, but it will still take up the same space as before. The element will be hidden, but still affect the layout.

Example

```
h1 {
visibility:hidden;
}
```

CSS Display and Visibility

- **display:none** hides an element, and it will not take up any space. The element will be hidden, and no effect on the layout

Example

```
h1 {
display:none;
}
```

CSS Positioning

CSS Positioning

- The position property specifies the type of positioning method used for an element. There are four different position values:
  - Static Positioning
  - Fixed Positioning
  - Relative Positioning
  - Absolute Positioning

CSS Positioning

- Depending on four other properties:
  - **Top:** sets the top edge of an element to a unit above/below (uses pixels and percent, negative values are allowed, default is auto)
  - **Bottom:** sets the bottom edge of an element to a unit above/below.

CSS Positioning

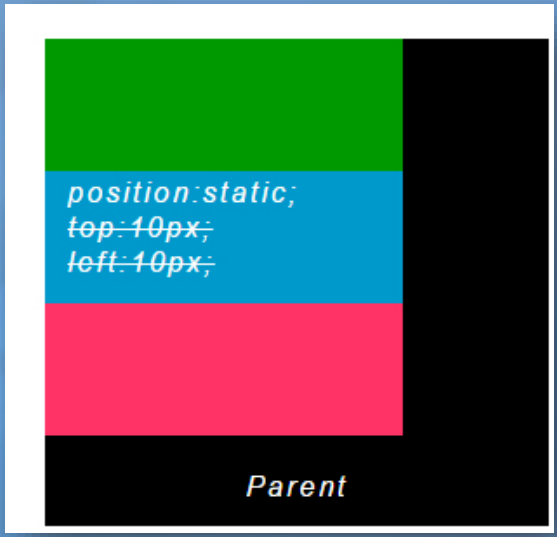
- **Left:** sets the left edge of an element to a unit to the left/right
- **right:** sets the right edge of an element to a unit to the left/right

✓ These properties will not work unless the position property is set first, and also will not work with the static position.

CSS Positioning

Static Positioning

- An element with static positioning is always positioned according to the normal flow of the page.
- **Property:** {position:static;}
- **Ex:**
  - { position: static;
    top:10px;
    left:10px; }



CSS Positioning

Fixed Positioning

- ✓ An element with fixed positioning is positioned relative to the window, which means it always stays in the same place even if the page is scrolled.

CSS Positioning

Fixed Positioning

Property: { position:fixed; }

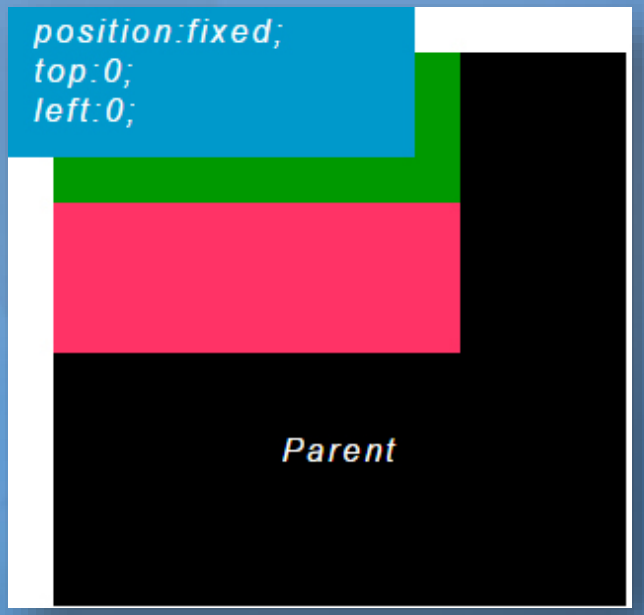
Ex: {

position: fixed;

top:0;

left:0;

}

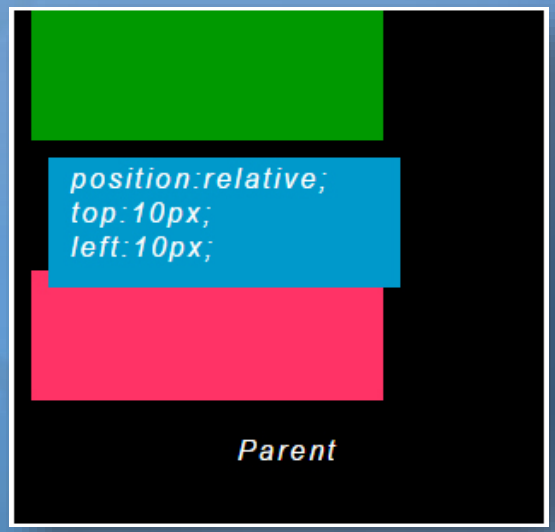


CSS Positioning

Relative Positioning

- ✓ An element with relative positioning is positioned relative to its normal position.
- ✓ Property: {position:relative;}
- ✓ Ex: {

```
position: relative;
top:10px;
left:10px;
}
```



CSS Positioning

Absolute Positioning

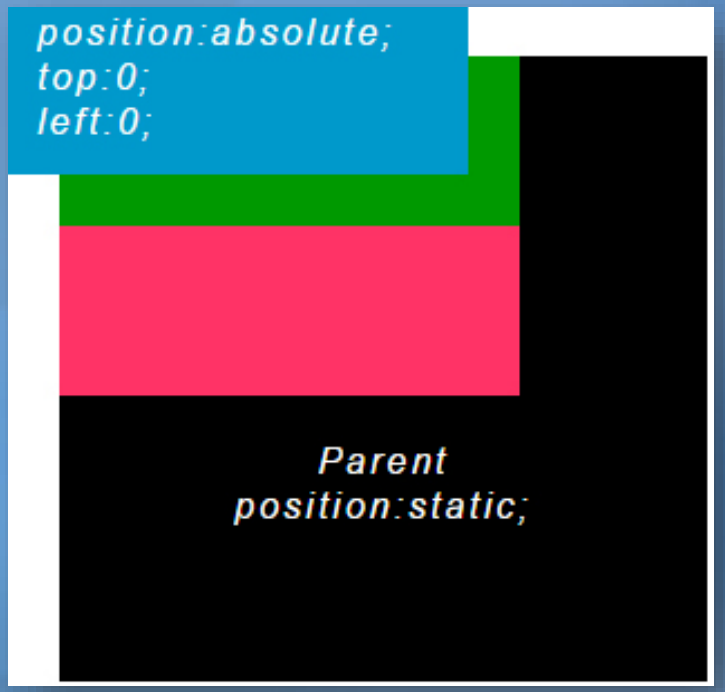
- ✓ An element with absolute positioning is positioned relative to the nearest parent that has “relative position”. If this parent is not found it will use the body.

CSS Positioning

Absolute Positioning

- ✓ Property: {position: absolute;}
- ✓ Ex 1:

```
.child   {
    position: absolute;
    left:0;
    top:0;
}
.parent {
    position: static;
}
```

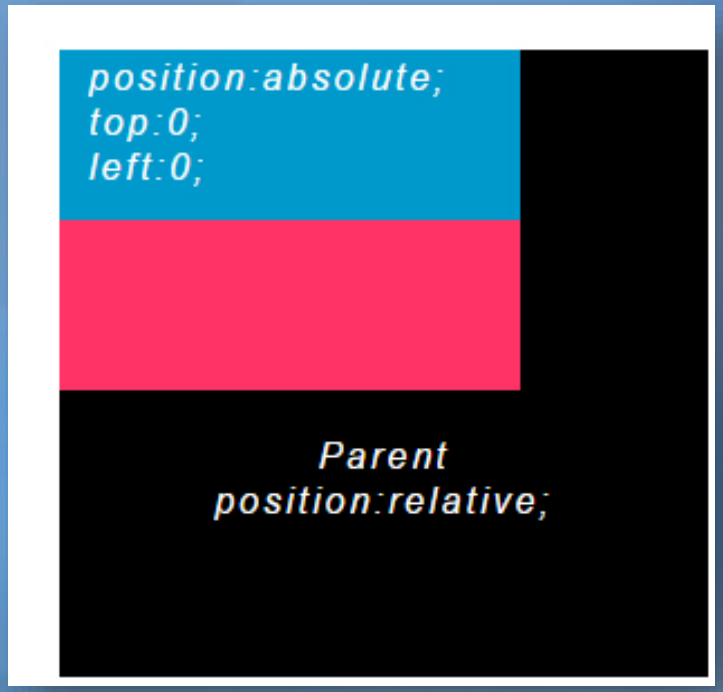


CSS Positioning

Absolute Positioning

- ✓ Property: {position: absolute;}
- ✓ Ex 2:

```
.child   {
    position: absolute;
    left:0;
    top:0;
}
.parent {
    position: relative;
}
```



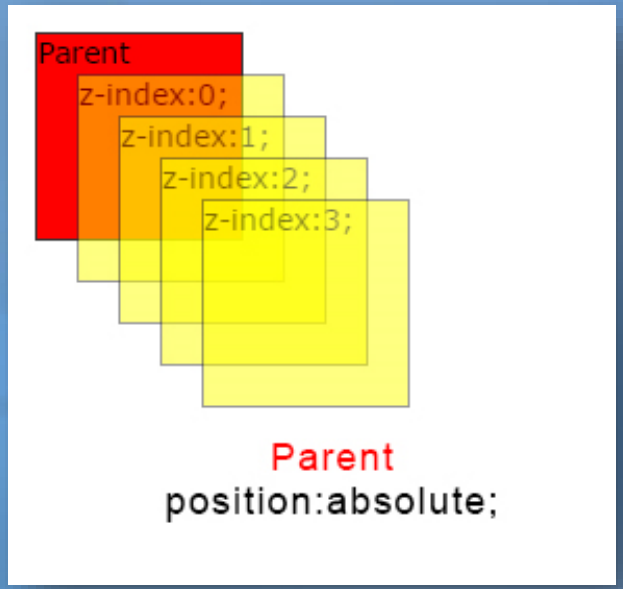
CSS Z-index

- ✓ Description: Sets the stack order of an element.

  Element with greater order is always in front.

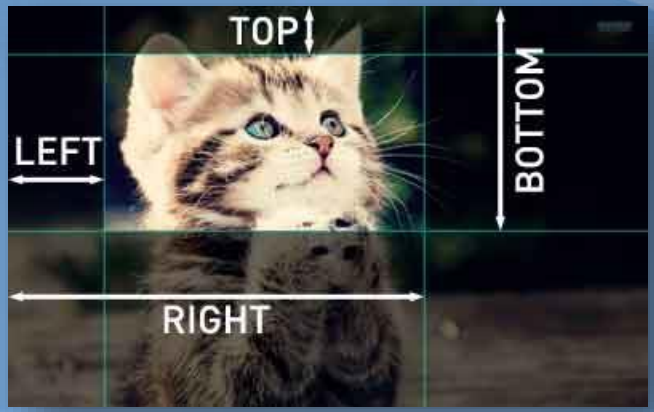
  z-index will not work with

  static positioning.
- ✓ Property: {z-index:5;}
- ✓ Values: auto | number (+/-)
- ✓ Default Value: auto



CSS Clip property

- ✓ specify a rectangle to crop an absolutely positioned element. This rectangle is specified as four coordinates.



- ✓ Property: auto | rect (top, right, bottom, left)
- ✓ Ex: {clip: rect(0px,25px,25px,0px);}
- ✓ Default Value: auto

Aligning Block Elements

Aligning Block Elements

Center Aligning

- ✓ Block elements can be center-aligned by setting the left and right margins to "auto".
- ✓ Setting the left and right margins to auto specifies that they should split the available margin equally.

Aligning Block Elements

Ex

```
.center
    {
        background-color:#b0e0e6;
        width:70%;
        margin-left:auto;
        margin-right:auto;
    }
```

Aligning Block Elements

Left and Right Aligning

- ✓ Block elements can be left and right-aligned by setting the absolute positioning
- ✓ Block elements also can be left and right-aligned by setting the float properties.

Aligning Block Elements

Ex

```
.right
    {
        width:300px;
        background-color:#b0e0e6;
        position:absolute;
        right:0px;
    }
```

Aligning Block Elements

Ex

```
.right
    {
        width:300px;
        background-color:#b0e0e6;
        float:right;
    }
```

Grouping Selectors

Grouping Selectors

- In style sheets there are often elements with the same style.

```
h1 {color:green;}
h2 {color:green;}
p {color:green;}
```

- To minimize the code, you can group selectors and separate each selector with a comma.

```
h1,h2,p {color:green;}
```

CSS Combinators

CSS Combinators

- ✓ Among the CSS selectors, we can include a combinator.
- ✓ A combinator is something that explains the relationship between the selectors.
- ✓ There are four different combinators in CSS: descendant selector, child selector, adjacent sibling selector, general sibling selector.

CSS Combinators

Descendant Selector (Space)

Selects all elements that are descendants of a specified element.

```
<style>
    div p
        {
            color: red;
        }
</style>
```

```
<div>
  <p>Para 1</p>
  <p>Para 2.</p>
  <span><p>Para3</p></span>
</div>
  <p>Para 4</p>
  <p>Para 5</p>
```

CSS Combinators

Child Selector (>)

Selects all elements that are the immediate (Direct) children of a specified element.

```
<style>
    div > p
        {
            color: red;
        }
</style>
```

```
<div>
  <p>Para 1</p>
  <p>Para 2.</p>
  <span><p>Para3</p></span>
</div>
  <p>Para 4</p>
  <p>Para 5</p>
```

CSS Combinators

adjacent sibling selector (+)

Selects an element that immediately follows a specified element.

```
<style>
    div + p
        {
            color: red;
        }
</style>
```

```
<div>
  <p>Para 1</p>
  <p>Para 2.</p>
  <span><p>Para3</p></span>
</div>
  <p>Para 4</p>
  <p>Para 5</p>
```

CSS Combinators

General Sibling Selector (~)

Selects all elements that are siblings of a specified element.

```
<style>
    div ~ p
        {
            color: red;
        }
</style>
```

```
<p>Para 0</p>
<div>
 <p>Para 1</p>
 <p>Para 2.</p>
 <span><p>Para3</p></span>
</div>
 <p>Para 4</p>
 <p>Para 5</p>
```

CSS Combinators

- ✓ Used to apply a style for a selector inside a selector.

```
.x1 p
{ color:white;}
```

- ✓ This code selects the p elements inside elements with class="x1":

Nesting Selectors

```
#x1 p
{ color:white;}
```

- ✓ **This code selects the p elements inside elements with class="x1":**

CSS cursor

- ✓ **cursor:** Specifies the type of cursor to be displayed.

  Values → Crosshair | default | pointer | move | e-resize | ne-resize | nw-resize | n-resize | se-resize | sw-resize | s-resize | w-resize | text | wait help | auto

CSS Pseudo Classes

Aligning Block Elements

CSS Pseudo Classes

- ✓ CSS pseudo-classes are used to add special effects to some selectors.

Aligning Block Elements

CSS Pseudo Classes

Selector	Example	Example description
:link	a:link	Selects all unvisited links
:visited	a:visited	Selects all visited links
:active	a:active	Selects the active link
:hover	a:hover	Selects links on mouse over
:focus	input:focus	Selects the input element which has focus
:first-letter	p:first-letter	Selects the first letter of every <p> element
:first-line	p:first-line	Selects the first line of every <p> element