SWE 363: Web Engineering & Development

Module 3-2

HTML5 - Semantic Structure & Forms

Objectives

- To learn Page-Structure Elements in HTML5
- To learn HTML5 forms

Outline

- Semantic Structure Elements
  - Header, footer
  - Navigation
  - Section, article
  - Aside
  - Figures and their caption

- Introducing forms
  - Sample HTML form
  - How forms work?
  - Form Elements.

References

- Most of the materials were taken from:
  - Deitel, Harvey, and Abbey Deitel. Internet and World Wide Web How to Program. Prentice Hall Press, 5th Edition. [Chapter 2]
  - Connolly, Randy. *Fundamentals of web development*. Pearson Education, 2015. [Chapter 2,4]

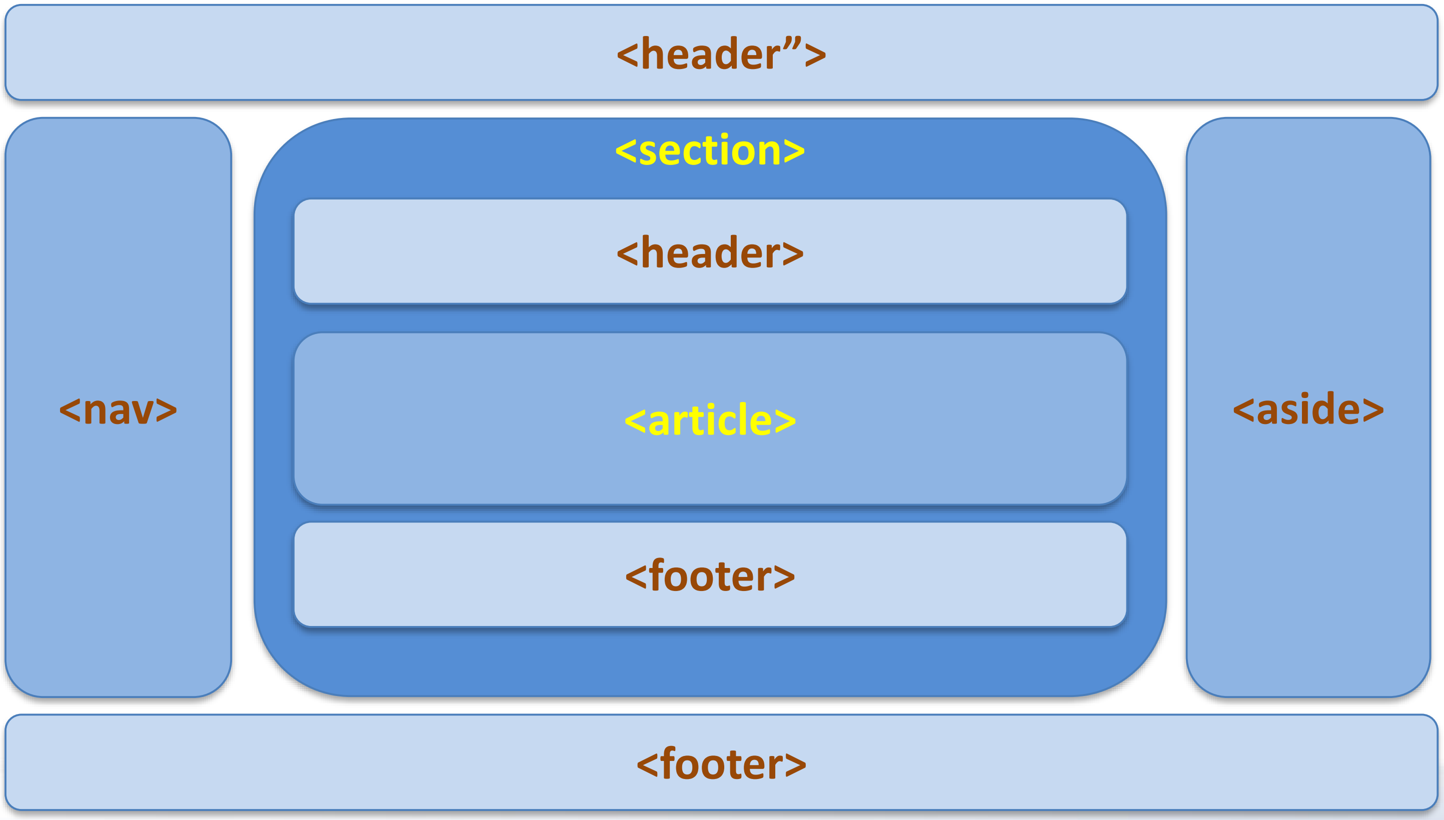
- Some useful links with examples and other resources:
  - Hickson, I. (Eds.). (2011). HTML Living Standard. Retrieved from http://www.whatwg.org/specs/web-apps/current-work/multipage/
  - World Wide Web Consortium. (n.d.). HTML 5 Tutorial. Retrieved from http://www.w3schools.com/html5/default.asp

HTML5 Semantic Structure Elements

Page-Structure Elements

Semantic Structure Elements

- HTML5 introduces several new page-structure elements
  - To identify areas of the page as headers, footers, articles, navigation areas, asides, figures and more.



Semantic Structure Elements

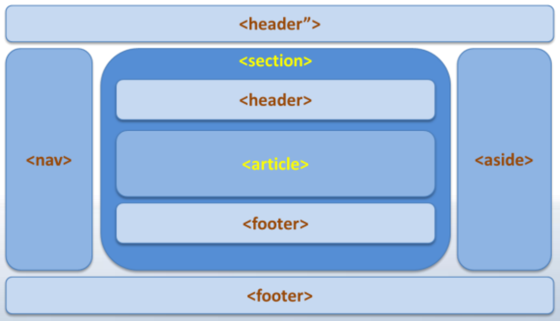
- **section** – Used for grouping together thematically-related content. Sounds like a div element, but its not. The div has **no semantic meaning**. Before replacing all your div's with section elements, always ask yourself, "Is all of the content related?"

- **aside** – used for indirectly related content. Just because some content appears to the left or right of the main content isn't enough reason to use the aside element. Ask yourself if the content within the aside can be removed without reducing the meaning of the main content.

- **header** – There is a crucial difference between the header element and the general accepted usage of header (or masthead). There's usually only one header or 'masthead' in a page. In HTML5 you can have as many as you want. The spec defines it as "a group of introductory or navigational aids". You can use a header in any section on your site. In fact, you probably should use a header within most of your sections.

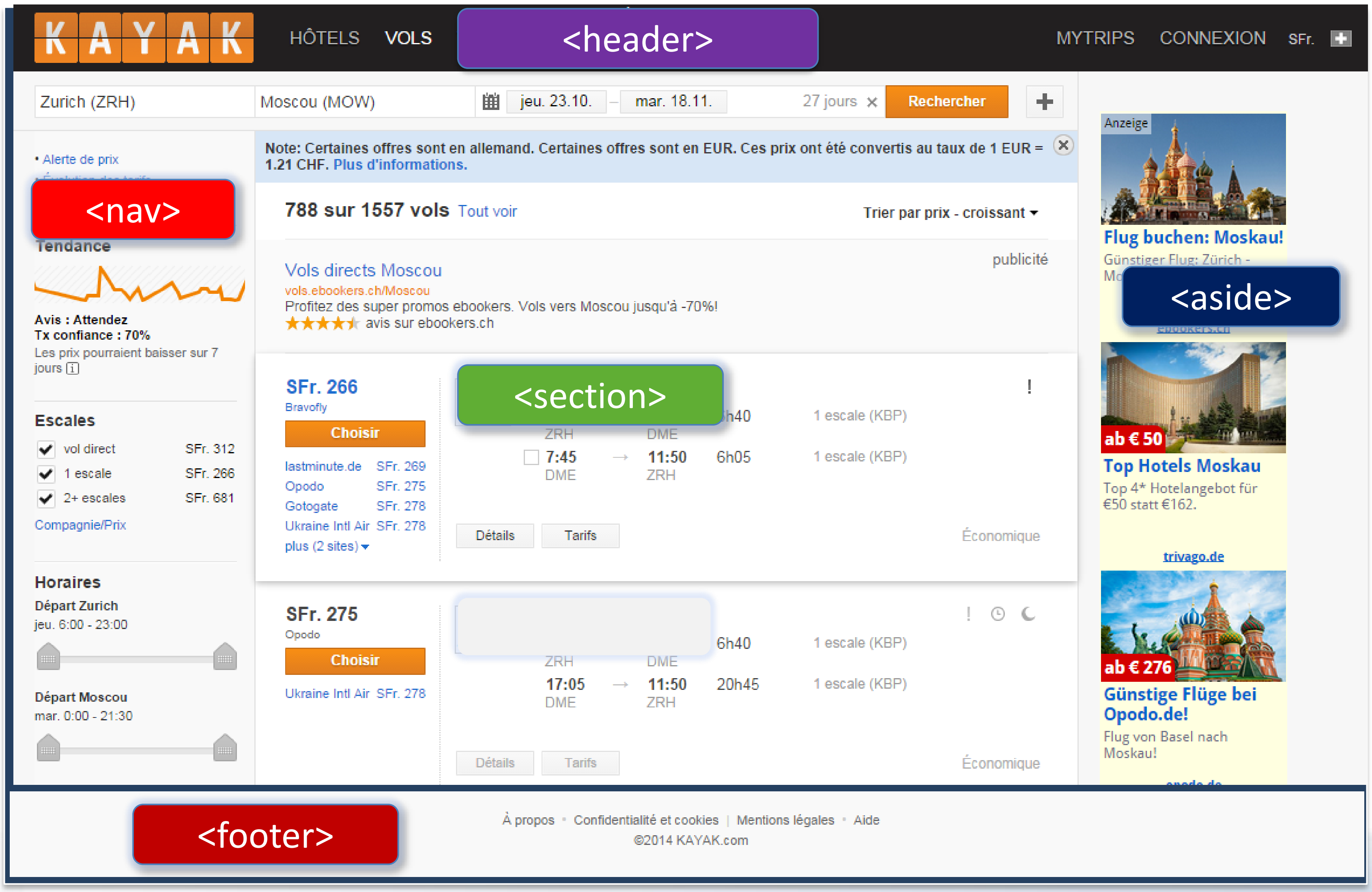
- **nav** – Intended for major navigation information. A group of links grouped together isn't enough reason to use the nav element. Site-wide navigation, on the other hand belongs in a nav element.

- **footer** – Sounds like its a description of the position, but its not. Footer elements contain information about it's containing element: who wrote it, copyright, links to related content, etc. Whereas we usually have one footer for an entire document, HTML5 allows us to also have footer within sections.



Semantic Structure Elements

Example



Note: Certaines offres sont en allemand. Certaines offres sont en EUR. Ces prix ont été convertis au taux de 1 EUR = 1.21 CHF. Plus d'informations.

788 sur 1557 vols Tout voir — Trier par prix - croissant

Vols directs Moscou — publicité
vols.ebookers.ch/Moscou
Profitez des super promos ebookers. Vols vers Moscou jusqu'à -70%!
★★★★★ avis sur ebookers.ch

Avis : Attendez
Tx confiance : 70%
Les prix pourraient baisser sur 7 jours

vol direct	SFr. 312
1 escale	SFr. 266
2+ escales	SFr. 681

Compagnie/Prix

Départ Zurich
jeu. 6:00 - 23:00

Départ Moscou
mar. 0:00 - 21:30

SFr. 266
Bravofly

lastminute.de	SFr. 269
Opodo	SFr. 275
Gotogate	SFr. 278
Ukraine Intl Air	SFr. 278

plus (2 sites)

6h40 — 1 escale (KBP) — ZRH — DME
7:45 DME → 11:50 ZRH — 6h05 — 1 escale (KBP)

SFr. 275
Opodo
Ukraine Intl Air SFr. 278

6h40 — 1 escale (KBP) — ZRH — DME
17:05 DME → 11:50 ZRH — 20h45 — 1 escale (KBP)

Anzeige
Flug buchen: Moskau!
Günstiger Flug: Zürich - Mo
ebookers.ch

ab € 50
Top Hotels Moskau
Top 4* Hotelangebot für €50 statt €162.
trivago.de

ab € 276
Günstige Flüge bei Opodo.de!
Flug von Basel nach Moskau!
opodo.de

À propos • Confidentialité et cookies | Mentions légales • Aide
©2014 KAYAK.com

Semantic Tagging with HTML5

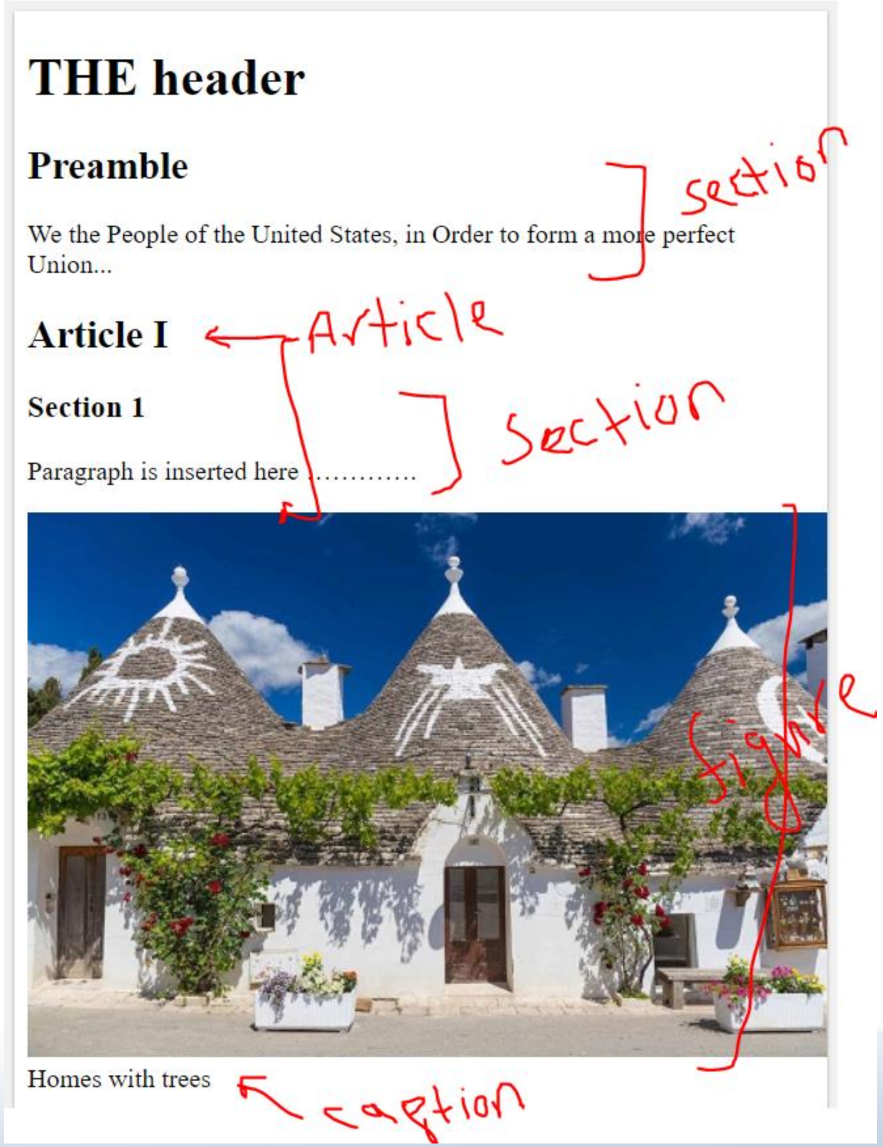
- HTML5 offers a set of new tags that provide the ability to mark up the sections of a document more descriptively than you could in HTML 4.01

HTML 4.01	$\longrightarrow$	HTML5
<div>		<article>
		<aside>
		<div>
		<header>
		<footer>
		<figure>
		<figcaption>
		<nav>
		<section>

Semantic Tagging with HTML5..

HTML 4.01

```
<body>
<h1>THE header </h1>
<div id="section">
   <h2>Preamble</h2>
   <p>We the People of the United
   States, in Order to form a more
   perfect Union...</p>
</div>
<div id="article">
  <h2>Article I</h2>
  <div id="section">
    <h3>Section 1</h3>
    <p> Paragraph is inserted here
    ………….</p>
  </div>
</div>
<div id="figure">
   <img src="bald_eagle.jpg"/>
   <div id="caption">
      The eagle has landed
   </div>
</div>
</body>
```



THE header

Preamble

We the People of the United States, in Order to form a more perfect Union...

Article I

Section 1

Paragraph is inserted here …………

Homes with trees

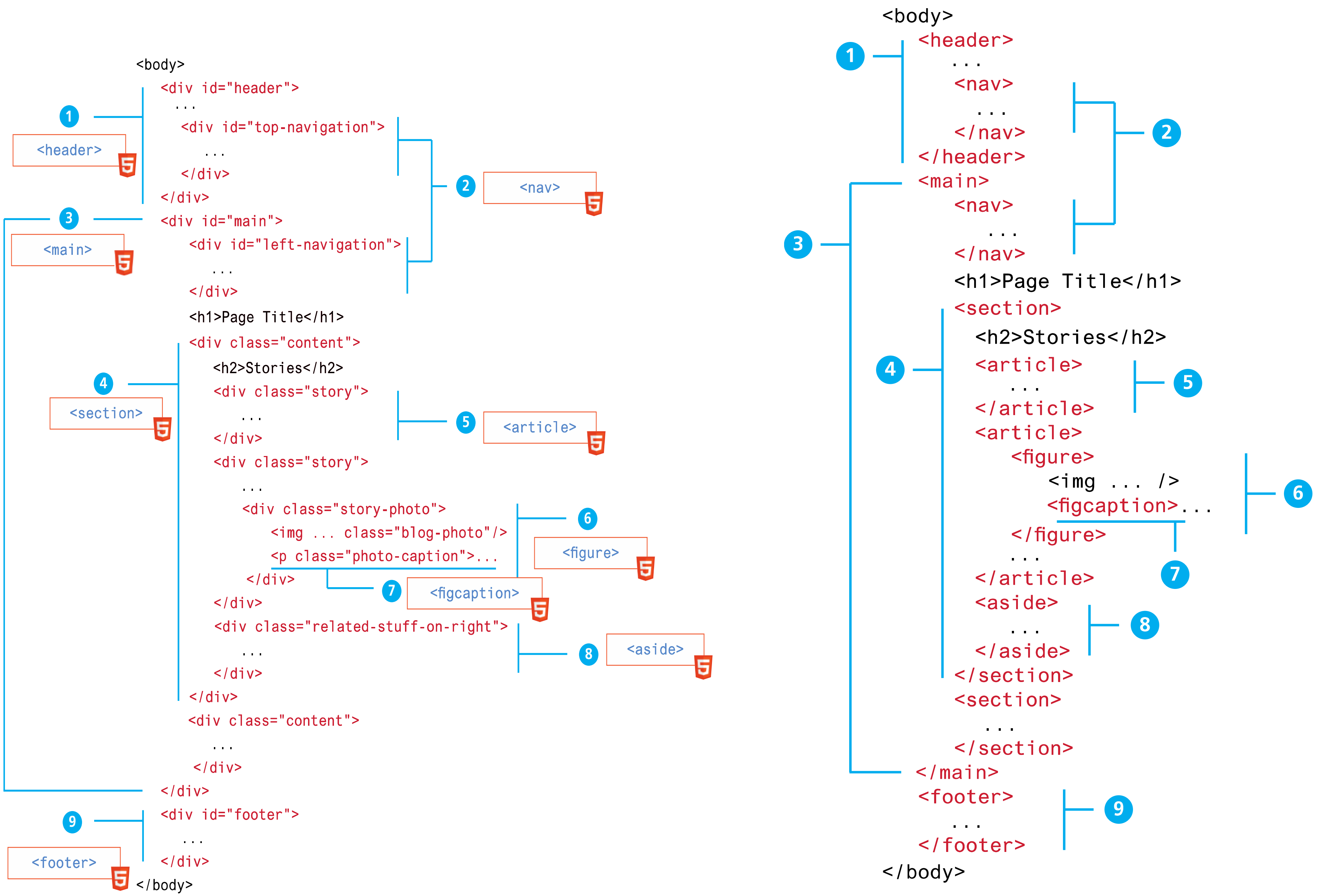
Semantic Tagging with HTML5..

HTML 4.01

```
<body>
<h1>THE header </h1>
<div id="section">
   <h2>Preamble</h2>
   <p>We the People of the United
   States, in Order to form a more
   perfect Union...</p>
</div>
<div id="article">
   <h2>Article I</h2>
  <div id="section">
    <h3>Section 1</h3>
    <p> Paragraph is inserted here
    ………….</p>
  </div>
</div>
<div id="figure">
   <img src="bald_eagle.jpg"/>
    <div id="caption">
      The eagle has landed
    </div>
</div>
</body>
```

HTML5

```
<body>
<h1>THE header </h1>
<section>
   <h2>Preamble</h2>
   <p>We the People of the United
   States, in Order to form a more
   perfect Union...</p>
</section>
<article>
    <h2>Article I</h2>
     <section>
       <h3>Section 1</h3>
       <p> Paragraph is inserted here
       ………….</p>
     </section>
</article>
<figure>
    <img src="bald_eagle.jpg"/>
<figcaption>
  The eagle has landed
</figcaption>
</figure>
</body>
```

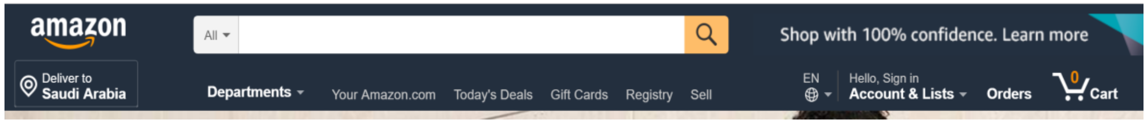


```
<body>
   <div id="header">
    ...
     <div id="top-navigation">
        ...
     </div>
   </div>
   <div id="main">
      <div id="left-navigation">
         ...
      </div>
      <h1>Page Title</h1>
      <div class="content">
         <h2>Stories</h2>
         <div class="story">
            ...
         </div>
         <div class="story">
            ...
            <div class="story-photo">
               <img ... class="blog-photo"/>
               <p class="photo-caption">...
            </div>
         </div>
         <div class="related-stuff-on-right">
            ...
         </div>
      </div>
      <div class="content">
         ...
      </div>
   </div>
   <div id="footer">
     ...
   </div>
</body>
```

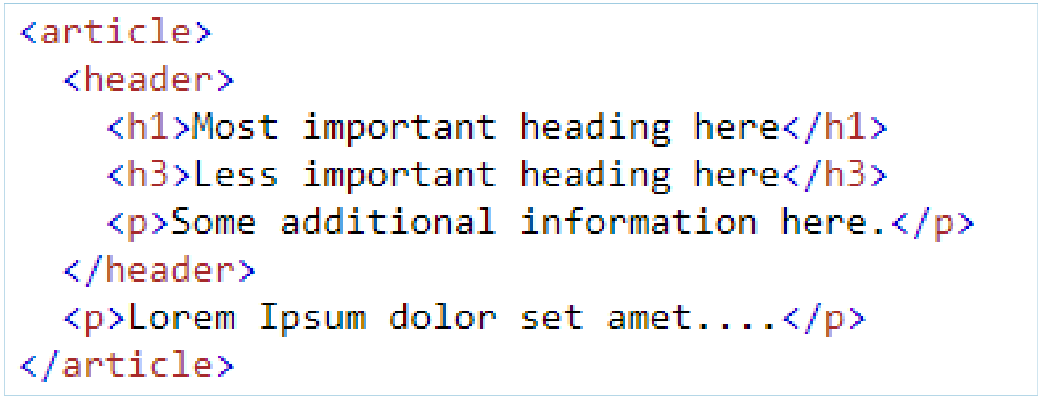
```
<body>
   <header>
      ...
      <nav>
        ...
      </nav>
   </header>
   <main>
      <nav>
        ...
      </nav>
      <h1>Page Title</h1>
      <section>
        <h2>Stories</h2>
        <article>
          ...
        </article>
        <article>
           <figure>
              <img ... />
              <figcaption>...
           </figure>
          ...
        </article>
        <aside>
          ...
        </aside>
      </section>
      <section>
        ...
      </section>
   </main>
   <footer>
      ...
   </footer>
</body>
```

Semantic Structure Elements

Header and Footer

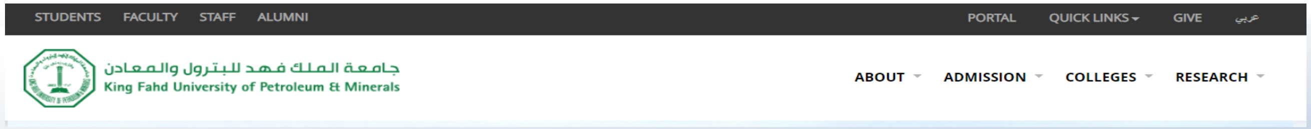


- ❑ Most website pages have a recognizable header and footer section.
- ❑ The <*header*> element represents a container for introductory content or a set of navigational links.
- ❑ Typically the header contains the site logo and title (and additional subtitles), horizontal navigation links, and perhaps one or two horizontal banners.
- ❑ You can have several <header> elements in one document.
  - ▪ But it cannot be placed within a <footer>, <address> or another <header> element.



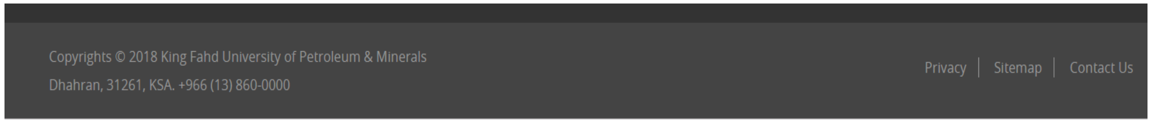
```
<article>
  <header>
    <h1>Most important heading here</h1>
    <h3>Less important heading here</h3>
    <p>Some additional information here.</p>
  </header>
  <p>Lorem Ipsum dolor set amet....</p>
</article>
```

➢ The header element can also be used to wrap **a section's table of contents**, a **search form**, or **any relevant logos**.



Semantic Structure Elements

Header and Footer



Copyrights © 2018 King Fahd University of Petroleum & Minerals
Dhahran, 31261, KSA. +966 (13) 860-0000

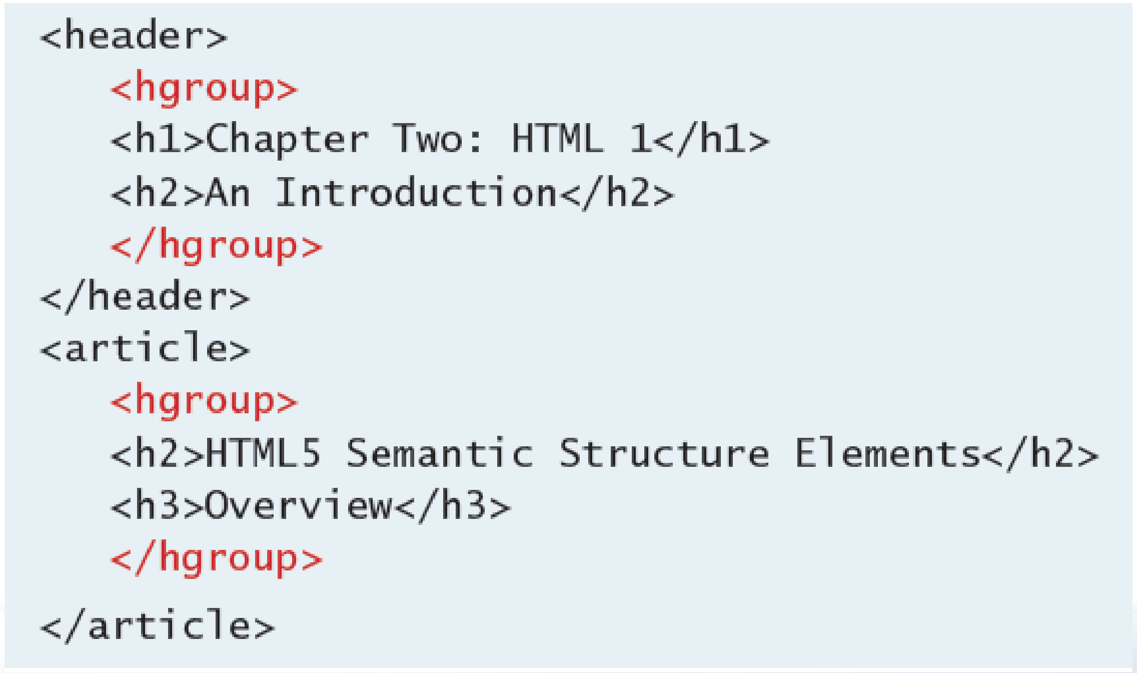
- The footer element describes a footer—content that usually appears at the bottom of the content or section element.
- A <footer> element typically contains:
  - authorship information, copyright information, contact information, sitemap
  - back to top links, related documents

- Both the HTML5 <header> and <footer> element can be used also for header and footer elements within other HTML5 containers, such as <article> or <section>

```
...
</header>
<article>
   <header>
      <h2>HTML5 Semantic Structure Elements</h2>
      <p>By <em>Randy Connolly</em></p>
      <p><time>September 30, 2015</time></p>
   </header>
   ...
</article>
```

Semantic Structure Elements
Heading Groups

- The <hgroup> element can be used to group the headings together within one container.
- The <hgroup> element can be used in contexts other than a header.
  - within an <article> or a <section> element as well.
- The <hgroup> element can *only* contain <h1>, <h2>, etc., elements.

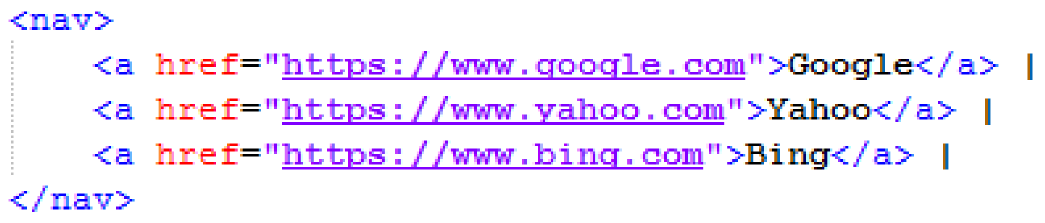


```
<header>
   <hgroup>
   <h1>Chapter Two: HTML 1</h1>
   <h2>An Introduction</h2>
   </hgroup>
</header>
<article>
   <hgroup>
   <h2>HTML5 Semantic Structure Elements</h2>
   <h3>Overview</h3>
   </hgroup>
</article>
```

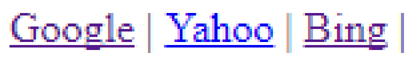
Semantic Structure Elements

Navigation

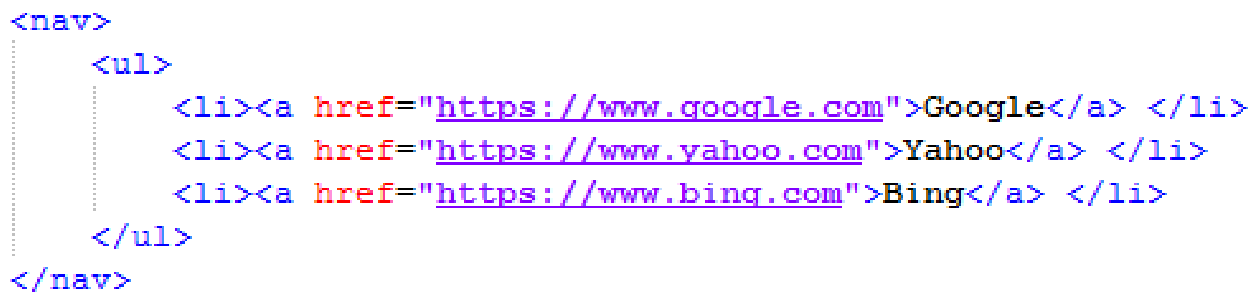
- The <nav> element represents a section of a page that contains links to other pages or to other parts within the same page.
- Common examples of navigation sections are menus, tables of contents, and indexes.
- Its purpose is to group navigation links.



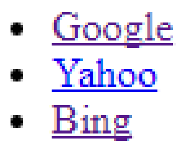
```
<nav>
    <a href="https://www.google.com">Google</a> |
    <a href="https://www.yahoo.com">Yahoo</a> |
    <a href="https://www.bing.com">Bing</a> |
</nav>
```



Google | Yahoo | Bing |



```
<nav>
    <ul>
        <li><a href="https://www.google.com">Google</a> </li>
        <li><a href="https://www.yahoo.com">Yahoo</a> </li>
        <li><a href="https://www.bing.com">Bing</a> </li>
    </ul>
</nav>
```



- Google
- Yahoo
- Bing

Semantic Structure Elements
Main

- <main> is meant to contain the main unique content of the document.
- <main> provides a semantic replacement for markup such as <div id="main"> or <div id="main-content">

Semantic Structure Elements
Articles and Sections

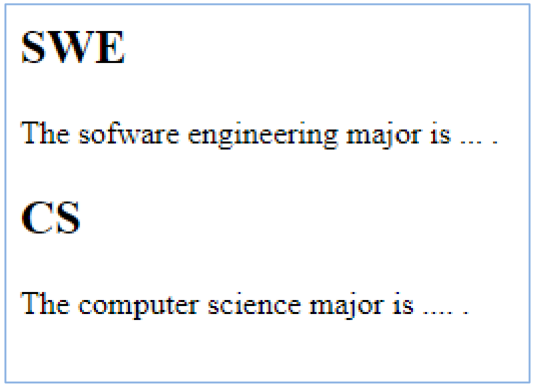
- <section> is a much broader element, while the
- <article> element is to be used for blocks of content that could potentially be read or consumed independently of the other content on the page

Semantic Structure Elements

Articles and Sections

- The <section> element defines sections in a document, such as chapters, headers, footers, or any other sections of the document.
  - The section element may also be nested in an article.

```
<section>
  <h1>SWE </h1>
  <p>The sofware engineering major is ... .</p>
</section>

<section>
  <h1>CS</h1>
  <p>The computer science major is .... .</p>
</section>
```



SWE

The sofware engineering major is ... .

CS

The computer science major is .... .

Semantic Structure Elements
Articles and Sections

- The <article> element describes standalone content that could potentially be used or distributed elsewhere
  - such as a news article, forum post or blog entry.
- You can nest article elements.
  - For example, reader comments about a magazine nested as an article within the magazine article.

```
  <article>
    <h1>Google Chrome</h1>
    <p>Google Chrome is a free, open-source web browser
developed by Google, released in 2008.</p>
  </article>
```

- You can use the *<section> element* to split the article into logical groups of content with headings:

Semantic Structure Elements
Figure and Figure Captions

- In traditional printed material like books and magazines, an image, chart, or code example would be accompanied by a caption.
  - HTML5 introduces → <figure> and <figcaption> elements.

- The figure element describes self-contained content, like illustrations, diagrams, photos, code listings, etc.

- While the content of the <figure> element is related to the main flow, its position is independent of the main flow, and if removed it should not affect the flow of the document.

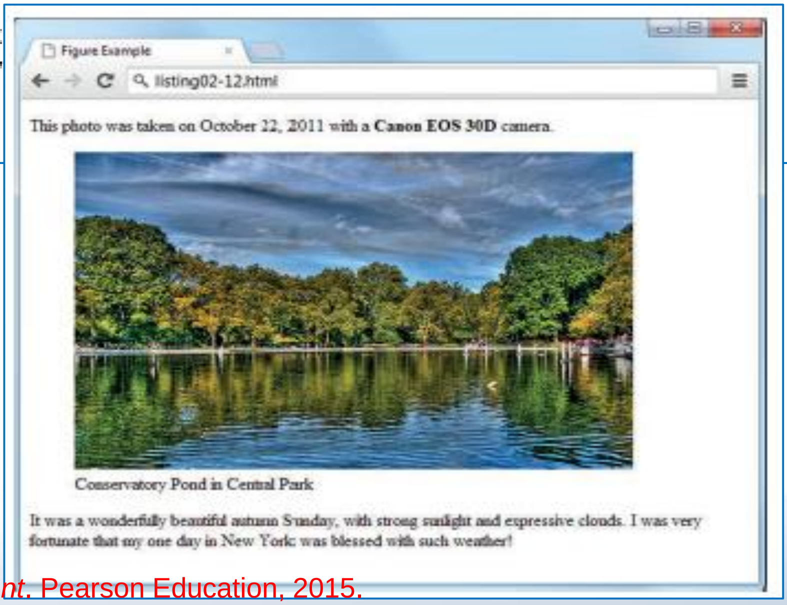
```
<figure>
  <img src="everest.jpg" alt="Mount Everest" width="304" height="228">
</figure>
```

Semantic Structure Elements

Figure and Figure Captions

- The <figcaption> tag defines a caption for a <figure> element.
- The <figcaption> element can be placed as the first or last child of the <figure> element.

```
<p>This photo was taken on October 22, 2011 with a Canon EOS 30D camera.</p>
<figure>
  <img src="images/central-park.jpg" alt="Central Park" /><br/>
  <figcaption>Conservatory Pond in Central Park</figcaption>
</figure>
<p>
It was a wonderfully beaut
expressive clouds. I was v
blessed with such weather!
</p>
```



This photo was taken on October 22, 2011 with a **Canon EOS 30D** camera.

Conservatory Pond in Central Park

It was a wonderfully beautiful autumn Sunday, with strong sunlight and expressive clouds. I was very fortunate that my one day in New York was blessed with such weather!

Semantic Structure Elements
Aside

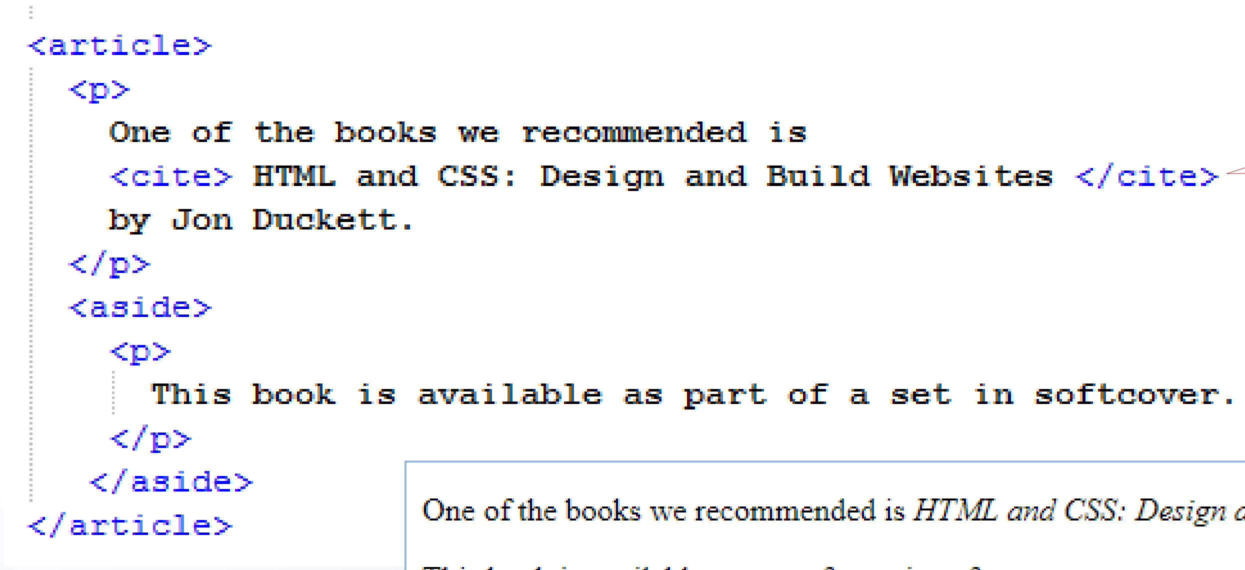
- The **<aside>** element is similar to the <figure> element in that it is used for marking up content that is separate from the main content on the page.

- But the <figure> element is used to indicate important information whose location on the page is somewhat unimportant, >> while the <aside> element “represents a section of a page that consists of content that is tangentially related to the content around the aside element”

- This **<aside>** element can be used for sidebars, pull quotes, groups of advertising images, or any other grouping of nonessential elements

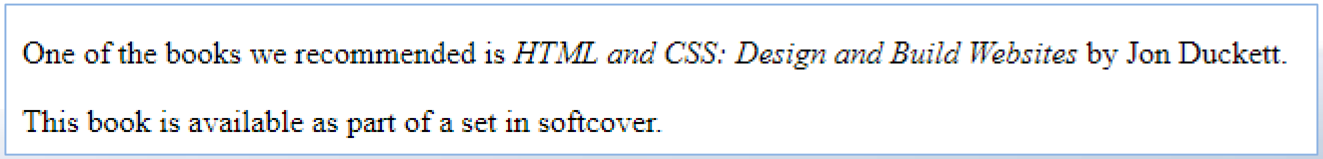
Semantic Structure Elements
Aside

- HTML 5 offers a new element to mark additional information that can enhance an article but isn't necessarily key to understand it.
- The aside element describes some content aside from the content it is placed in, but it should be related to the surrounding content.
  - The aside element can be used within or outside the article element
  - The aside element is a block level element



```
<article>
  <p>
    One of the books we recommended is
    <cite> HTML and CSS: Design and Build Websites </cite>
    by Jon Duckett.
  </p>
  <aside>
    <p>
      This book is available as part of a set in softcover.
    </p>
   </aside>
</article>
```

<cite> used to define titles etc.



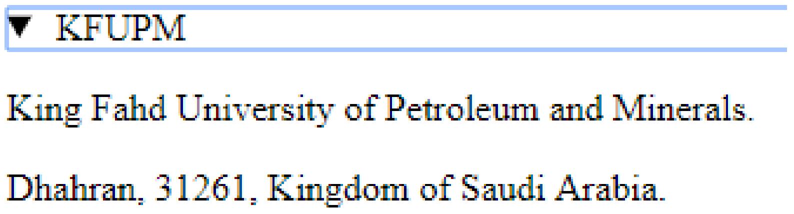
One of the books we recommended is *HTML and CSS: Design and Build Websites* by Jon Duckett.

This book is available as part of a set in softcover.

Semantic Structure Elements
Details and Summary

- The *summary* displays a right-pointing arrow next to a summary or caption when the document is rendered in a browser
  - When clicked, the arrow points downward and reveals the content in the details element.

```
<details>
  <summary> KFUPM </summary>
  <p> King Fahd University of Petroleum and Minerals.</p>
  <p> Dhahran, 31261, Kingdom of Saudi Arabia.</p>
</details>
```



▼ KFUPM

King Fahd University of Petroleum and Minerals.

Dhahran, 31261, Kingdom of Saudi Arabia.

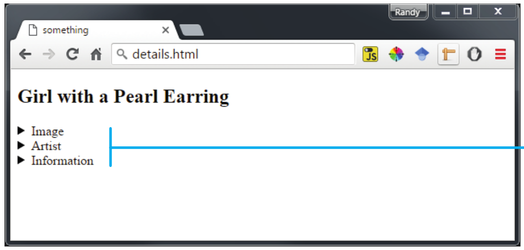
- The <summary> tag is new in HTML5.

https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_summary

Not supported

Semantic Structure Elements

Details and Summary

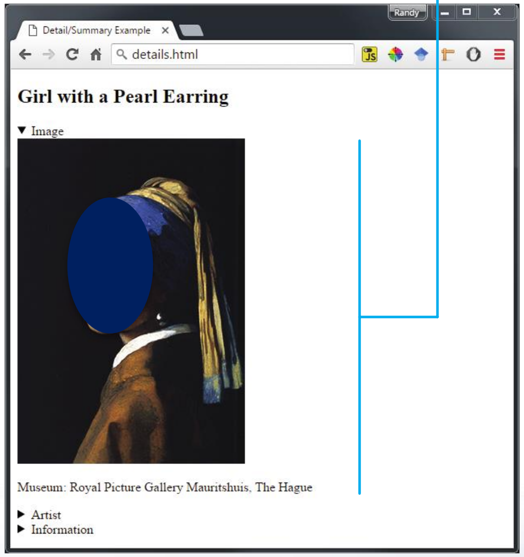

something
details.html

Girl with a Pearl Earring

▶ Image
▶ Artist
▶ Information

```
<body>
   <h2>Girl with a Pearl Earring</h2>
   <details>
     <summary>Image</summary>
     <img src="images/106020.jpg"><br>
     <p>Museum: Royal Picture Gallery Mauritshuis ...
   </details>
   <details>
     <summary>Artist</summary>
     <p><strong>Jan Vermeer</strong> was a Dutch ...
   </details>
   <details>
     <summary>Information</summary>
     <p>
        Date: 1665<br>
        Medium: Oil on Canvas
     </p>
  </details>
</body>
```

Clicking on the summary label reveals the rest of the content with the `<details>` container


Detail/Summary Example
details.html

Girl with a Pearl Earring

▼ Image

Museum: Royal Picture Gallery Mauritshuis, The Hague

▶ Artist
▶ Information

More Semantic/Structural Elements

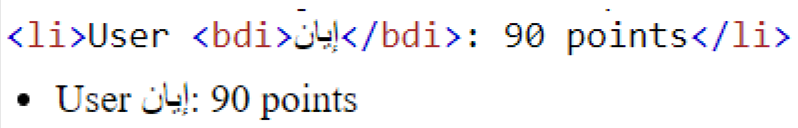
- HTML5 offers new elements for better document structure:

Tag	Description
<bdi>	Isolates a part of text that might be formatted in a different direction from other text outside it
<dialog>	Defines a dialog box or window. Example 01, Example 02
<main>	Defines the main content of a document
<mark>	Defines marked/highlighted text
<menu>	To define a list/menu of commands (only supported in Firefox)
<menuitem>	To define a command/menu item that the user can invoke from a popup menu (only supported in Firefox)
<rp>	Defines what to show in browsers that do not support ruby annotations
<rt>	Defines an explanation/pronunciation of characters (for East Asian typography)
<ruby>	Defines a ruby annotation (for East Asian typography)
<wbr>	Defines a possible line-break

https://www.w3schools.com/html/html5_new_elements.asp

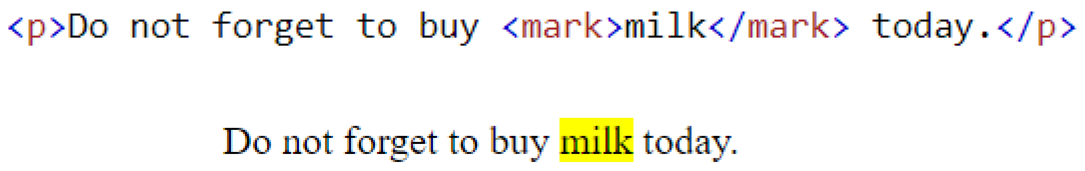
More Semantic/Structural Elements

Tag	Description
<bdi>	Isolates a part of text that might be formatted in a different direction from other text outside it
<dialog>	Defines a dialog box or window. [Example 01](), [Example 02]()
<main>	Defines the main content of a document
<mark>	Defines marked/highlighted text



```
<li>User <bdi>إيان</bdi>: 90 points</li>
```

- User إيان: 90 points



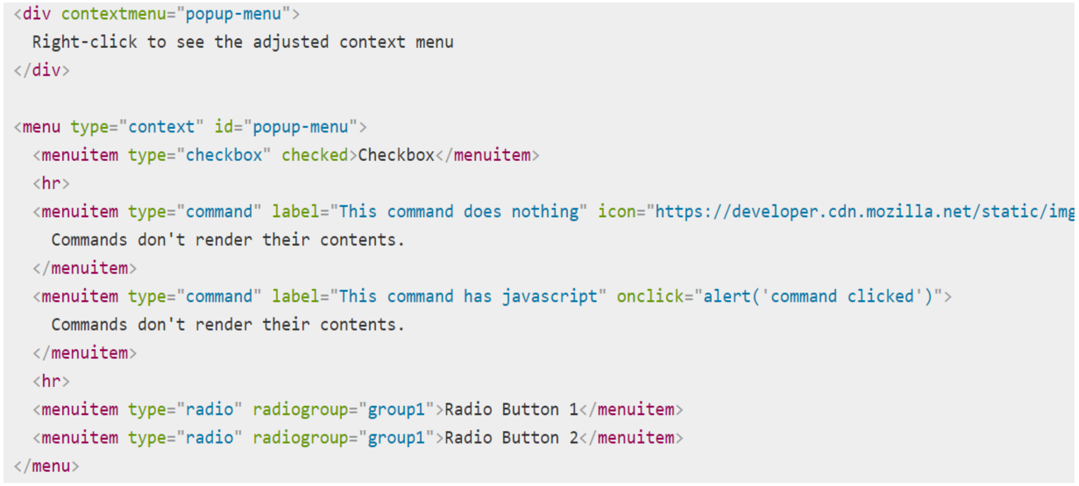
```
<p>Do not forget to buy <mark>milk</mark> today.</p>
```

Do not forget to buy milk today.

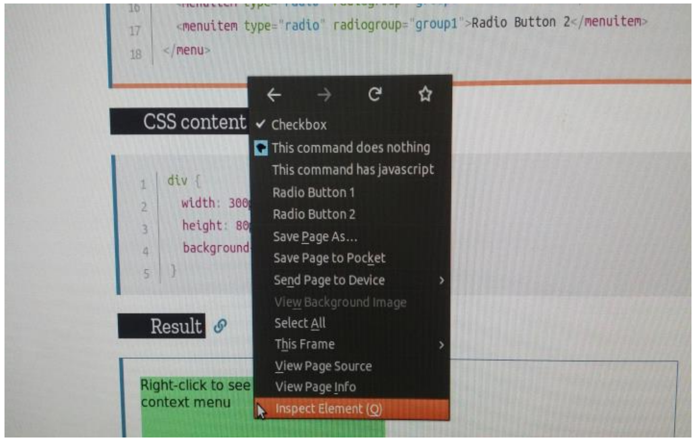
https://www.w3schools.com/html/html5_new_elements.asp

More Semantic/Structural Elements

Tag	Description
<menu>	To define a list/menu of commands (only supported in Firefox)
<menuitem>	To define a command/menu item that the user can invoke from a popup menu (only supported in Firefox)



```
<div contextmenu="popup-menu">
  Right-click to see the adjusted context menu
</div>

<menu type="context" id="popup-menu">
  <menuitem type="checkbox" checked>Checkbox</menuitem>
  <hr>
  <menuitem type="command" label="This command does nothing" icon="https://developer.cdn.mozilla.net/static/img
    Commands don't render their contents.
  </menuitem>
  <menuitem type="command" label="This command has javascript" onclick="alert('command clicked')">
    Commands don't render their contents.
  </menuitem>
  <hr>
  <menuitem type="radio" radiogroup="group1">Radio Button 1</menuitem>
  <menuitem type="radio" radiogroup="group1">Radio Button 2</menuitem>
</menu>
```

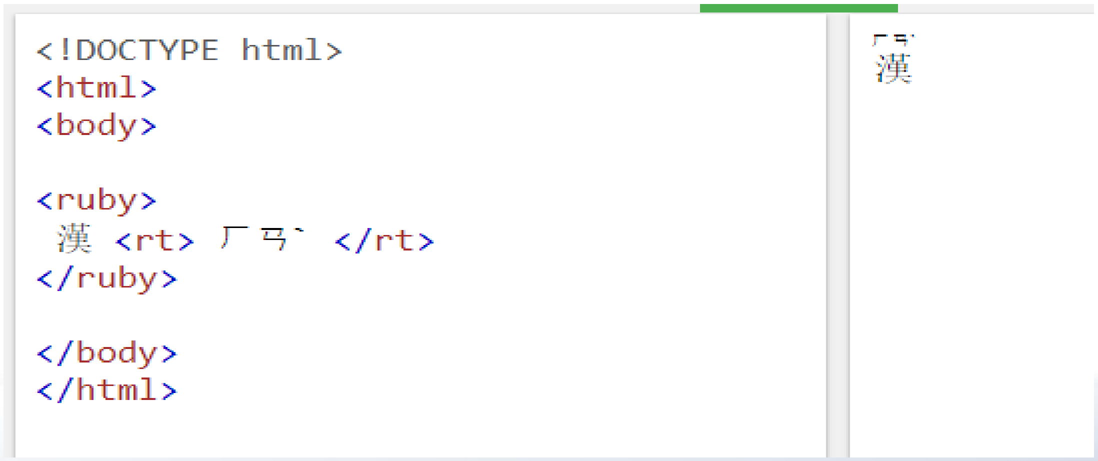


https://www.w3schools.com/html/html5_new_elements.asp

More Semantic/Structural Elements…

Tag	Description
<rp>	Defines what to show in browsers that do not support ruby annotations
<rt>	Defines an explanation/pronunciation of characters (for East Asian typography)
<ruby>	Defines a ruby annotation (for East Asian typography)

- The <ruby> tag specifies a ruby annotation.
- A ruby annotation is a small extra text, attached to the main text to indicate the pronunciation or meaning of the corresponding characters. This kind of annotation is often used in Japanese publications.

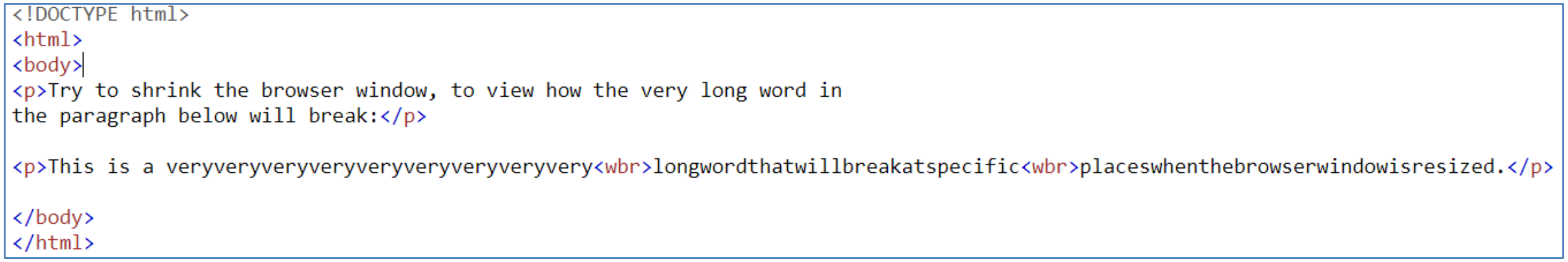


```
<!DOCTYPE html>
<html>
<body>

<ruby>
 漢 <rt> ㄏㄢˋ </rt>
</ruby>

</body>
</html>
```

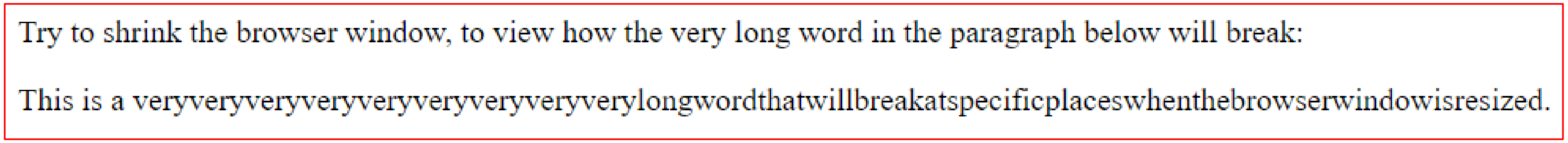
More Semantic/Structural Elements…

Tag	Description
<wbr>	Defines a possible line-break

- The <wbr> (Word Break Opportunity) tag specifies where in a text it would be ok to add a line-break.



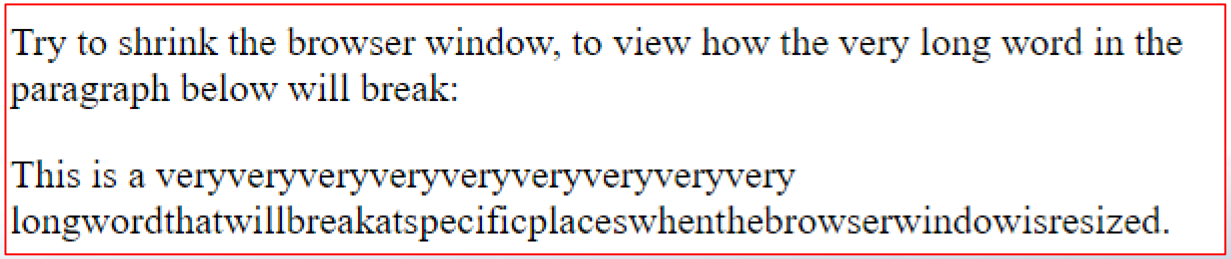
```
<!DOCTYPE html>
<html>
<body>
<p>Try to shrink the browser window, to view how the very long word in
the paragraph below will break:</p>

<p>This is a veryveryveryveryveryveryveryveryvery<wbr>longwordthatwillbreakatspecific<wbr>placeswhenthebrowserwindowisresized.</p>

</body>
</html>
```



Try to shrink the browser window, to view how the very long word in the paragraph below will break:

This is a veryveryveryveryveryveryveryveryverylongwordthatwillbreakatspecificplaceswhenthebrowserwindowisresized.

After page resizing



Try to shrink the browser window, to view how the very long word in the paragraph below will break:

This is a veryveryveryveryveryveryveryveryvery
longwordthatwillbreakatspecificplaceswhenthebrowserwindowisresized.

Introducing Forms

Forms provide the user with a way to interact with a web server.

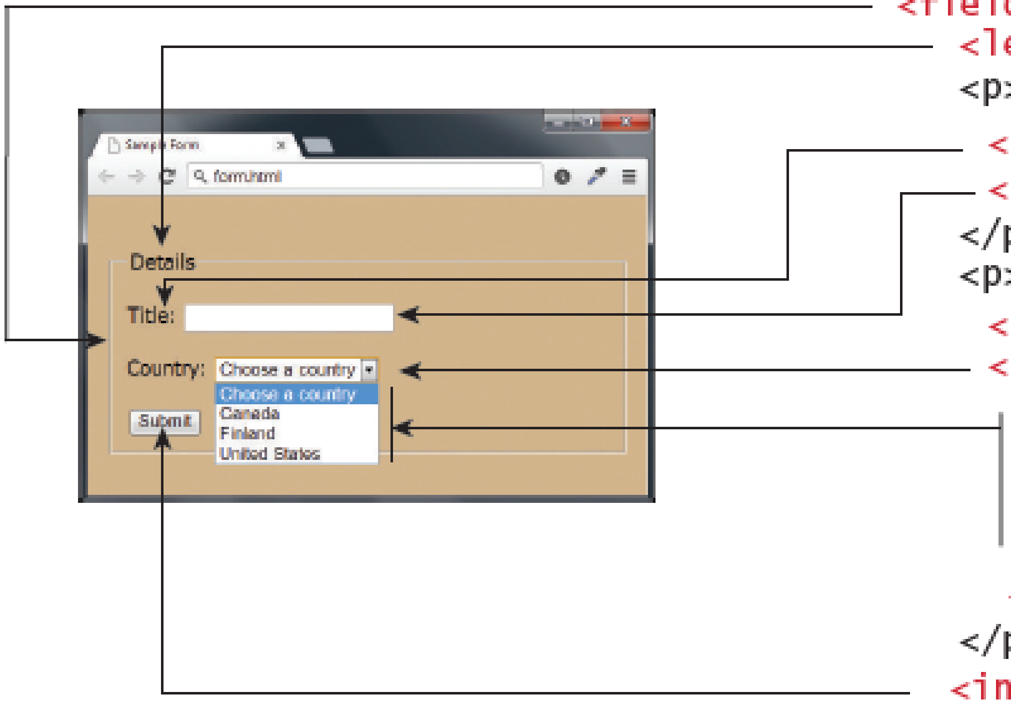
Forms

- HTML web forms let websites become interactive by allowing user input
  - Forms are used to collect different input data from a user
- Forms have front-end and back-end processing.
  - HTML takes care of the front end
  - Another language (such as PHP or ASP) takes care of the back-end

- Forms usually include a series of html input tags and other elements
- HTML can create forms , but HTML alone is not enough to process forms

- Prior to HTML5, there were a limited number of data-entry controls available in HTML forms.
  - There were controls for entering text, controls for choosing from a list, buttons, checkboxes, and radio buttons.
  - HTML5 has added a number of new controls as well as more customization options for the existing controls.

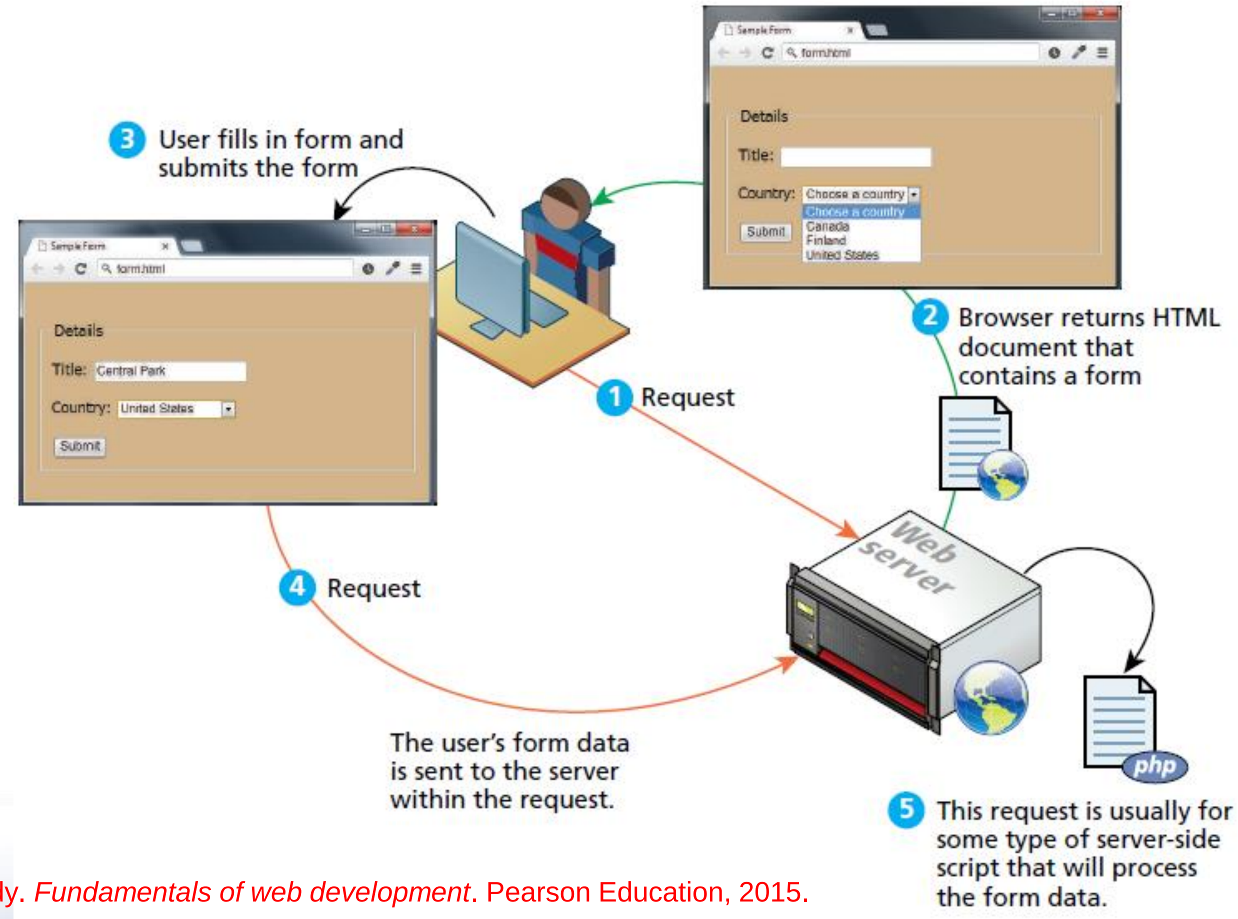
Sample HTML form



```
<form method="get" action="process.php">
  <fieldset>
    <legend>Details</legend>
    <p>
      <label>Title: </label>
      <input type="text" name="title" />
    </p>
    <p>
      <label>Country: </label>
      <select name="where">
        <option>Choose a country</option>
        <option>Canada</option>
        <option>Finland</option>
        <option>United States</option>
      </select>
    </p>
    <input type="submit" />
  </fieldset>
</form>
```

How Forms Work

- While forms are constructed with HTML elements, a form also requires some type of server-side resource that processes the user's form input

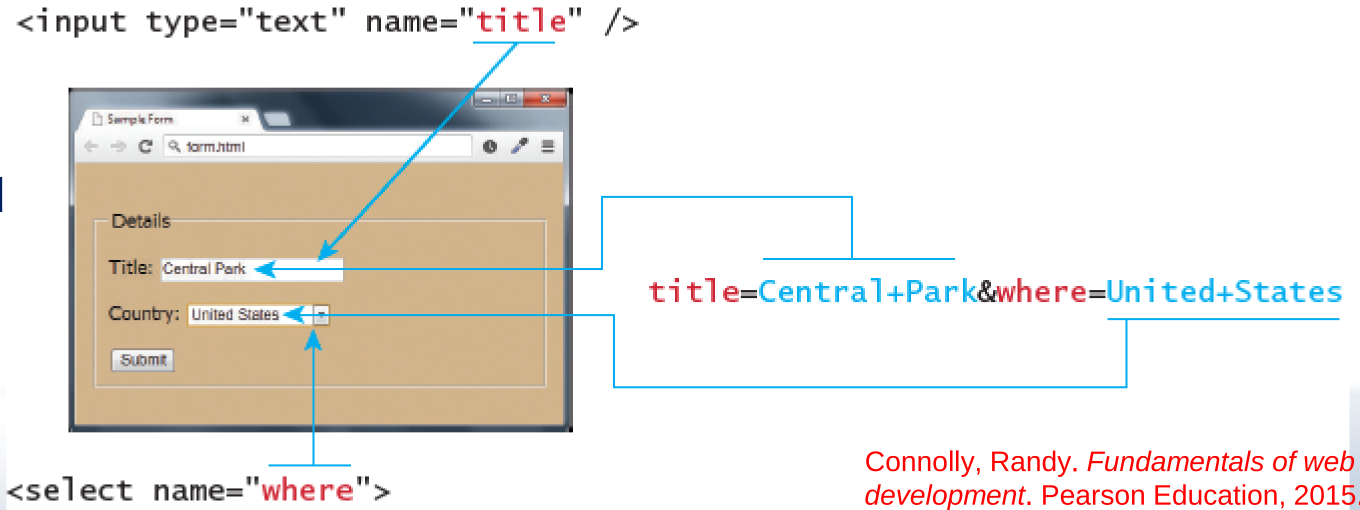


3 User fills in form and submits the form
Sample Form
form.html
Details
Title: Central Park
Country: United States
Submit
Sample Form
form.html
Details
Title:
Country: Choose a country
Choose a country
Canada
Finland
United States
Submit
2 Browser returns HTML document that contains a form
1 Request
4 Request
Web server
php
The user's form data is sent to the server within the request.
5 This request is usually for some type of server-side script that will process the form data.

Query Strings

- ❑ How the browser “sends” the data to the server?
  - via an HTTP request
- ❑ How is the data packaged in a request?
  - Query Strings

- ❑ A **query string** is a series of name=value pairs separated by ampersands (the **&** character).
  - the names in the query string were defined by the HTML form



```
<input type="text" name="title" />
```

```
title=Central+Park&where=United+States
```

```
<select name="where">
```

Query string data and its connection to the form elements

Connolly, Randy. *Fundamentals of web development*. Pearson Education, 2015.

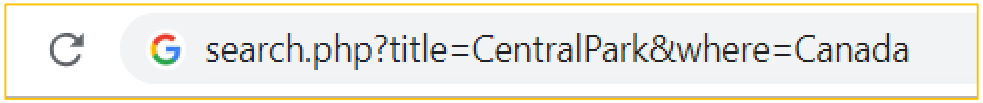
The <form> Element

- A form is defined with the <form> tag (wraps around the entire form)

  <form> … … … … … </form>

- Attributes:
  - *Action*- specifies the page or script used to process the form on the server side
  - *method*:- specifies how the data will be transmitted from the browser to the server. There are two possibilities: **GET** and **POST**.

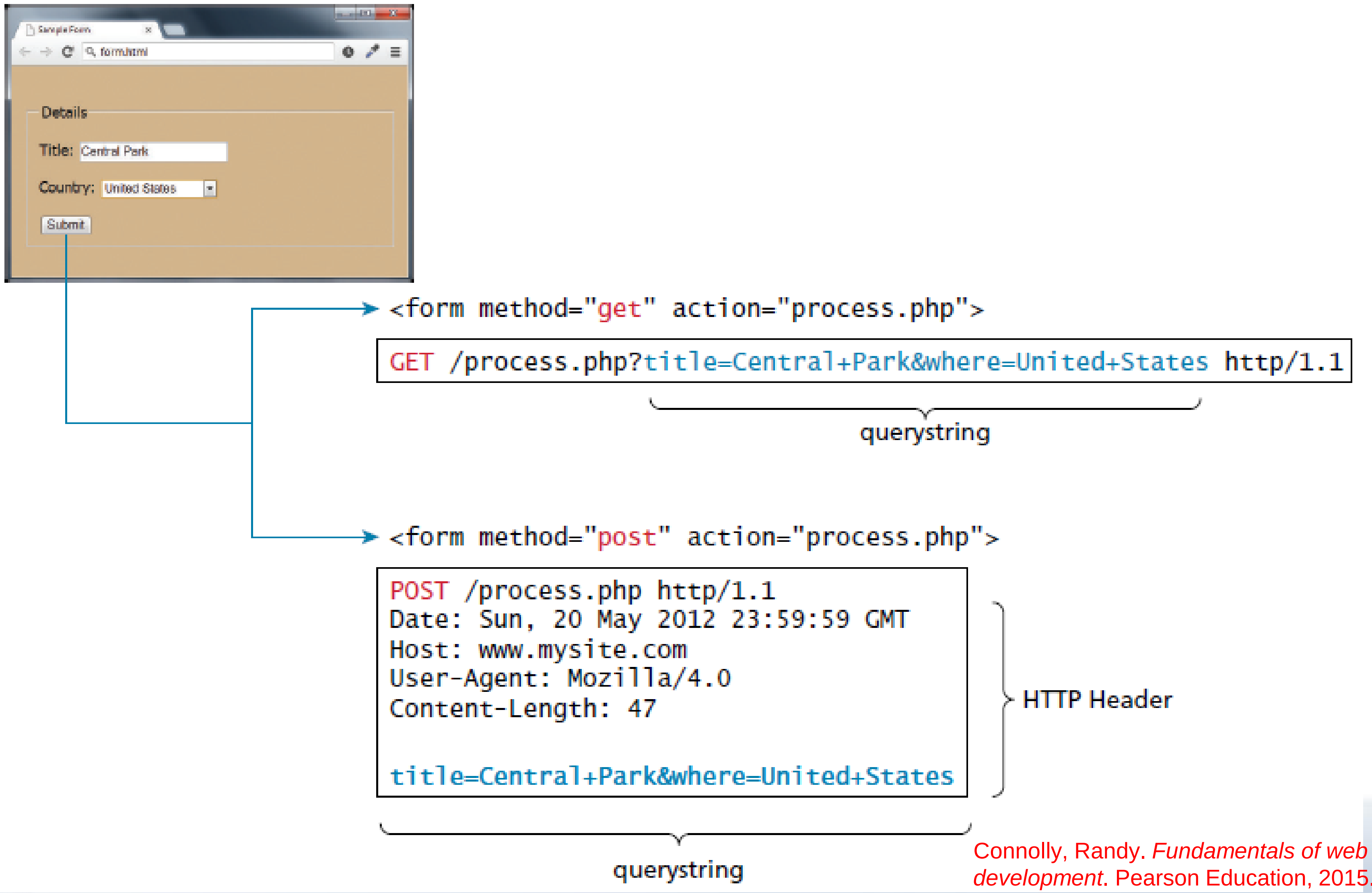
- With using **method = "get"**, the browser locates the data in the URL of the request;
  - appends the form data directly to the end of the URL of the script, where it's visible in the browser's Address field.



search.php?title=CentralPark&where=Canada

- With using **method = "post"**, the form data is located in the HTTP header after the HTTP variables

GET versus POST



```
<form method="get" action="process.php">
```

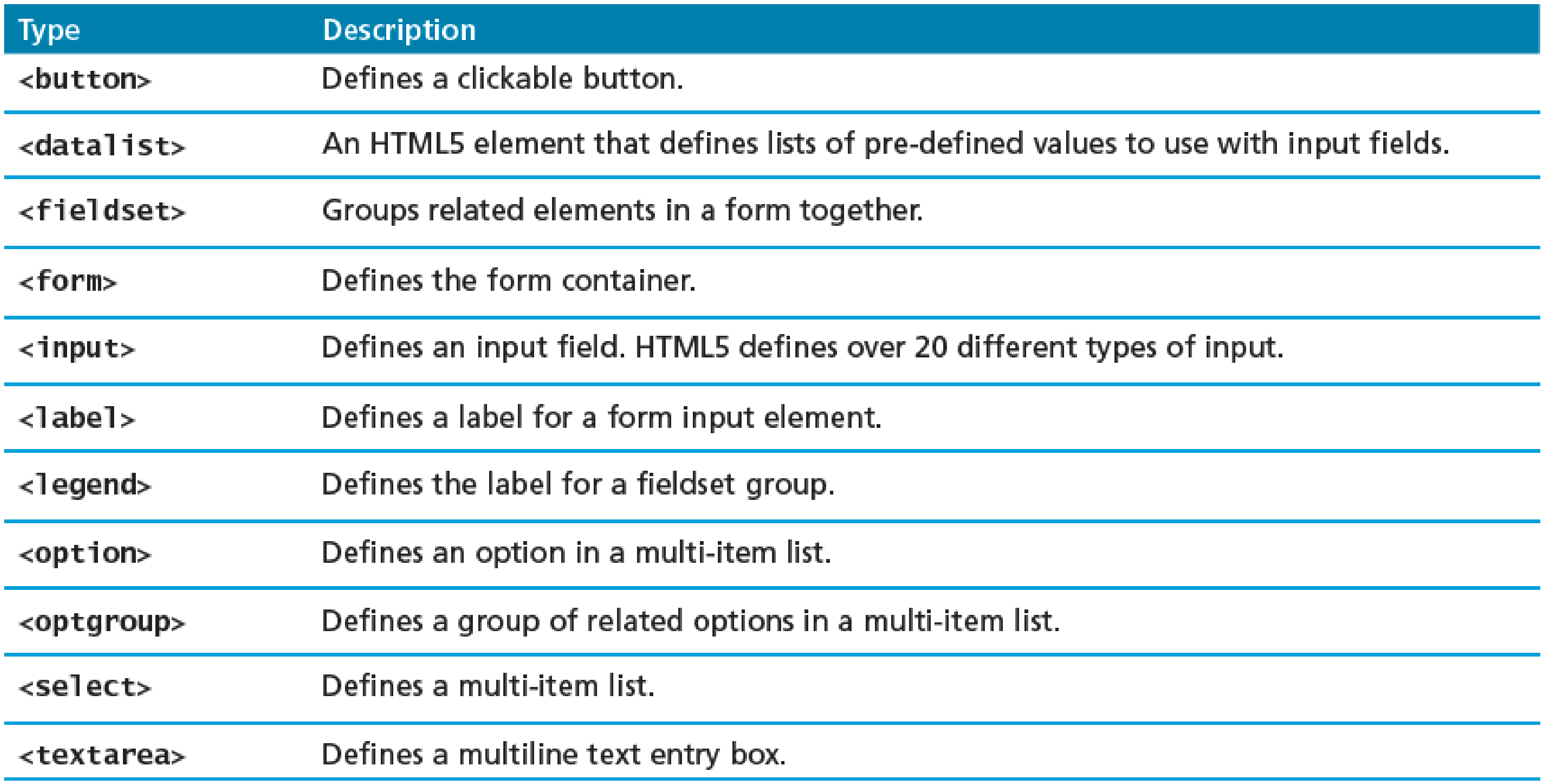
```
GET /process.php?title=Central+Park&where=United+States http/1.1
```

```
<form method="post" action="process.php">
```

```
POST /process.php http/1.1
Date: Sun, 20 May 2012 23:59:59 GMT
Host: www.mysite.com
User-Agent: Mozilla/4.0
Content-Length: 47

title=Central+Park&where=United+States
```

Connolly, Randy. *Fundamentals of web development*. Pearson Education, 2015.

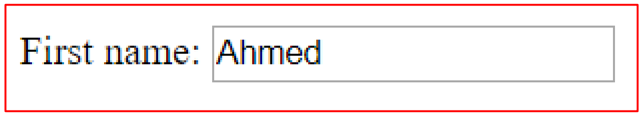
Form-Related HTML Elements



Type	Description
`<button>`	Defines a clickable button.
`<datalist>`	An HTML5 element that defines lists of pre-defined values to use with input fields.
`<fieldset>`	Groups related elements in a form together.
`<form>`	Defines the form container.
`<input>`	Defines an input field. HTML5 defines over 20 different types of input.
`<label>`	Defines a label for a form input element.
`<legend>`	Defines the label for a fieldset group.
`<option>`	Defines an option in a multi-item list.
`<optgroup>`	Defines a group of related options in a multi-item list.
`<select>`	Defines a multi-item list.
`<textarea>`	Defines a multiline text entry box.

Text Input Controls

- Most forms need to gather text information from the user.

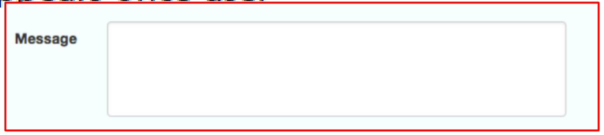


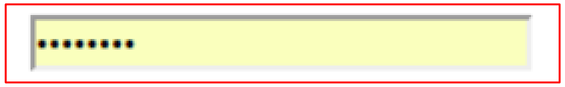
- The <input> element is the most used element.
  - It can be displayed in several ways, depending on the type attribute.
  - The type of input is specified with the type attribute: text fields, radio button, etc.

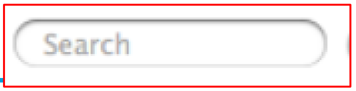
- The <input> specifies an input field where a user can enter data.
  - type: This specifies the type of input (text, password, etc)
  - name: Specifies a name for the element. Used in server-side and/or client-side processing
  - size: Specifies the width in characters
  - max: The maximum amount of characters
  - and more..

Text Input Controls..

Type	Description
`text`	Creates a single-line text entry box. `<input type="text" name="title" />`
`textarea`	Creates a multiline text entry box. You can add content text or if using an HTML5 browser, placeholder text (hint text that disappears once user begins typing into the field). `<textarea rows="3" ... />`
`password`	Creates a single-line text entry box for a password (which masks the user entry as bullets or some other character) `<input type="password" ... />`
`search`	Creates a single-line text entry box suitable for a search string. This is an HTML5 element. Some browsers on some platforms will style search elements differently or will provide a clear field icon within the text box. `<input type="search" ... />`
`email`	Creates a single-line text entry box suitable for entering an email address. This is an HTML5 element. Some devices (such as the iPhone) will provide a specialized keyboard for this element. Some browsers will perform validation when form is submitted. `<input type="email" ... />`

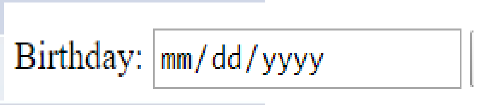


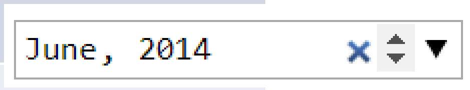




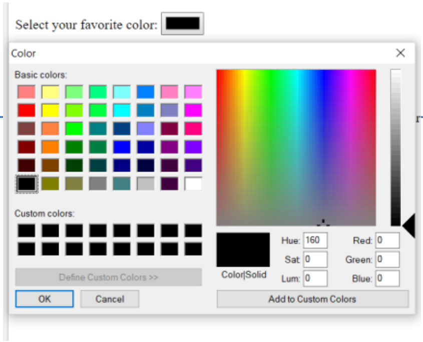
Text Input Controls..

Type	Description
tel	The input value is of type telephone number
search	The input field is a search field
url	The input value is a URL
email	The input value is one or more email addresses
datetime	The input value is a date and/or time
date	The input value is a date
month	The input value is a month
week	The input value is a week
time	The input value is of type time
datetime-local	The input value is a local date/time
number	The input value is a number
range	The input value is a number in a given range
color	The input value is a hexadecimal color, like #FF8800
placeholder	Specifies a short hint that describes the expected value of an input field


Birthday: mm/dd/yyyy


June, 2014

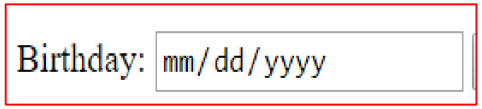
Text Input Controls..


Select your favorite color:

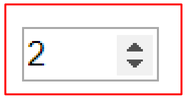
Hue: 160 Sat: 0 Lum: 0
Red: 0 Green: 0 Blue: 0

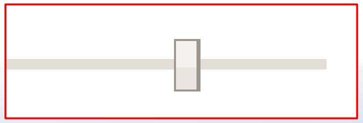
- These are not yet universally supported by all browsers
  - Examples:

```
<p>
   <label>Date:
      <input type = "date" />
        (yyyy-mm-dd)
   </label>
</p>
<p>
   <label>Datetime:
      <input type = "datetime" />
        (yyyy-mm-ddThh:mm+ff:gg, such as 2012-01-27T03:15)
   </label>
</p>
<p>
   <label>Datetime-local:
      <input type = "datetime-local" />
        (yyyy-mm-ddThh:mm, such as 2012-01-27T03:15)
   </label>
</p>
<p>
   <label>Email:
      <input type = "email" placeholder = "name@domain.com"
        required /> (name@domain.com)
   </label>
</p>
```


Birthday: mm/dd/yyyy

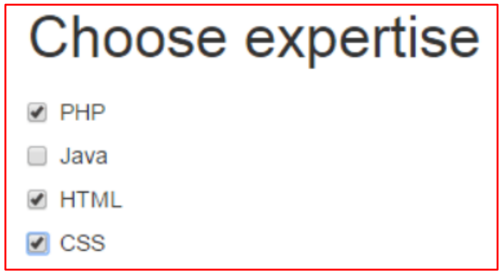
```
<label>Color:
   <input type = "color" autofocus />
     (Hexadecimal code such as #ADD8E6)
</label>


<p>
   <label>Number:
      <input type = "number"
        min = "0"
        max = "7"
        step = "1"
        value = "4" />
   </label> (Enter a number between 0 and 7)
</p>
<p>
   <label>Range:
      0 <input type = "range"
        min = "0"
        max = "20"
        value = "10" /> 20
   </label>
</p>
```


2

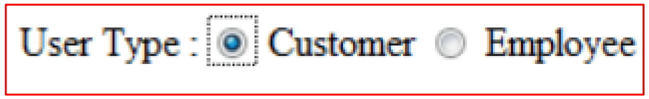


Additional Elements

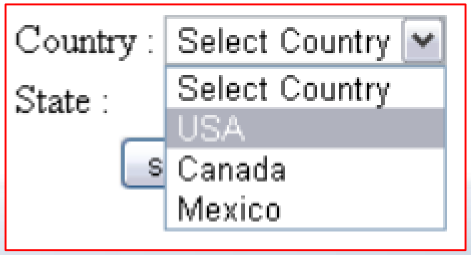
- **checkbox** input element enables users to select an option.
  - When the checkbox is selected, a check mark appears in the checkbox . Otherwise, the checkbox is empty
  - checkboxes can be used individually and in groups. checkboxes that are part of the same group have the same name



- **radio** buttons are similar to checkboxes, except that only one radio button in a group can be selected at any time.
  - All radio buttons in a group have the same name attribute but different value attributes.



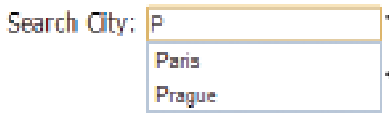
- **select** input provides a drop-down list of items.
  - The name attribute identifies the drop-down list.
  - The option element adds items to the drop-down list.



Choice Controls
Select Lists

- The <select> element is used to create a multiline box for selecting one or more items.
- The options (defined using the <option> element) can be hidden in a dropdown list or multiple rows of the list can be visible.

- Example: Using the <datalist> element

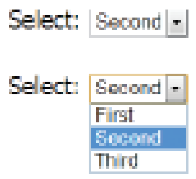


```
<input type="text" name="city" list="cities"  />

<datalist id="cities">
   <option>Calcutta</option>
   <option>Calgary</option>
   <option>London</option>
   <option>Los Angeles</option>
   <option>Paris</option>
   <option>Prague</option>
</datalist>
```

Not supported

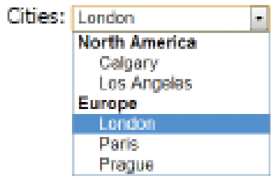
Choice Controls
Select Lists

- The selected attribute in the <option> makes it a default value.



```
<select name="choices">
  <option>First</option>
  <option selected>Second</option>
  <option>Third</option>
</select>
```

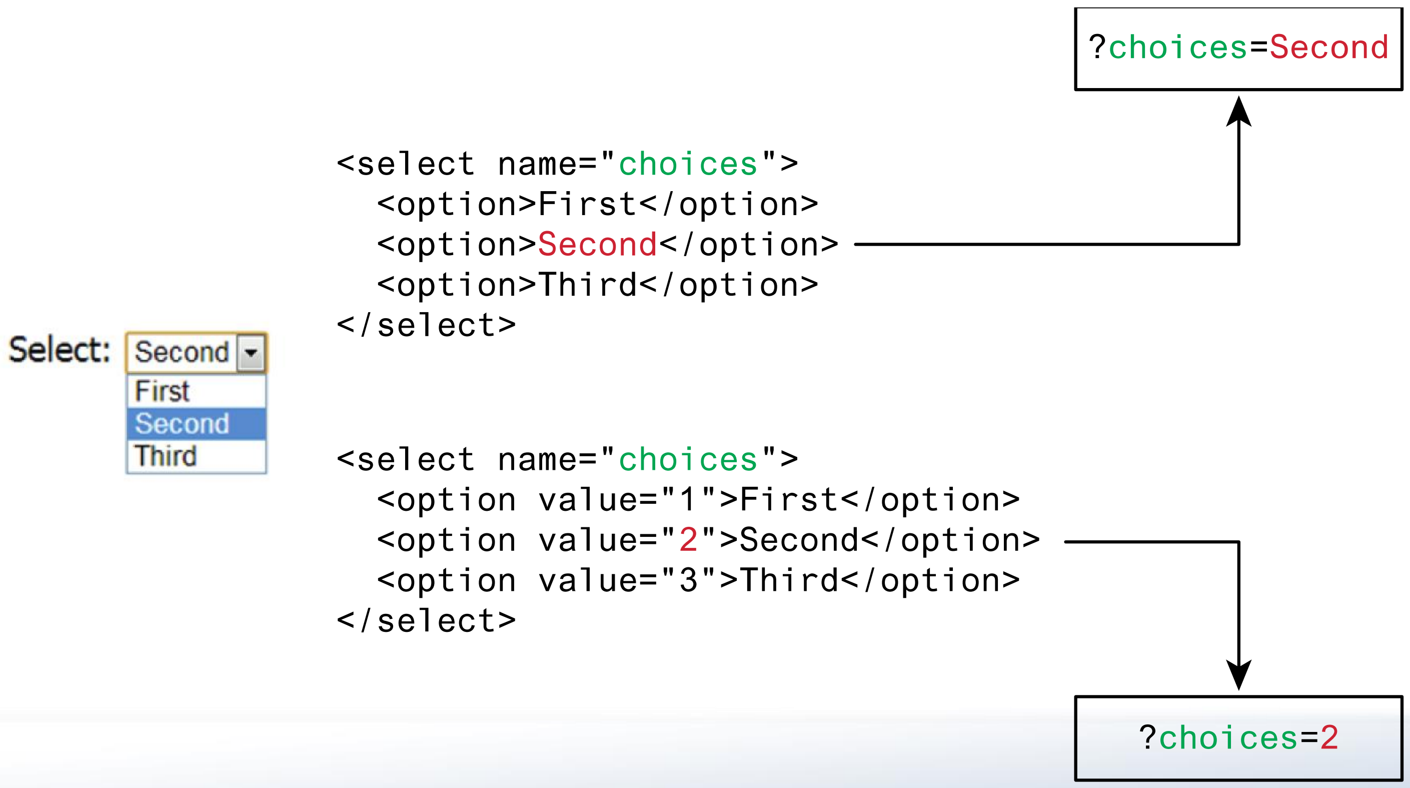
- Option items can be grouped together via the <optgroup> element.



```
<select … >
 <optgroup label="North America">
   <option>Calgary</option>
   <option>Los Angeles</option>
 </optgroup>
 <optgroup label="Europe">
   <option>London</option>
   <option>Paris</option>
   <option>Prague</option>
 </optgroup>
</select>
```

Choice Controls
Select Lists

- The value attribute of the <option> element is used to specify what value will be sent back to the server in the query string when that option is selected.
- The value attribute is optional; if it is not specified, then the text within the container is sent instead



```
?choices=Second
```

```
<select name="choices">
  <option>First</option>
  <option>Second</option>
  <option>Third</option>
</select>
```

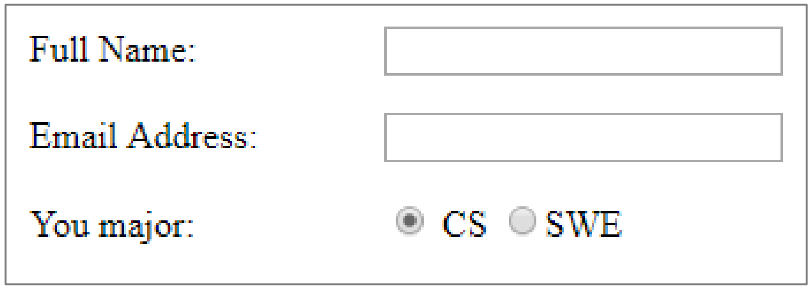
```
<select name="choices">
  <option value="1">First</option>
  <option value="2">Second</option>
  <option value="3">Third</option>
</select>
```

```
?choices=2
```

Choice Controls
Radio Buttons

- Radio buttons are useful when you want the user to select a single item from a small list of choices and you want all the choices to be visible.
- The checked attribute is used to indicate the default choice, while the value attribute works in the same manner as with the <option> element.

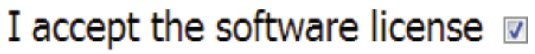
```
<label for="area"> You major: </label>
<input type="radio" name="major" value="cs" checked> CS
<input type="radio" name="major" value="swe">SWE
```



Full Name:	
Email Address:	
You major:	● CS ○ SWE

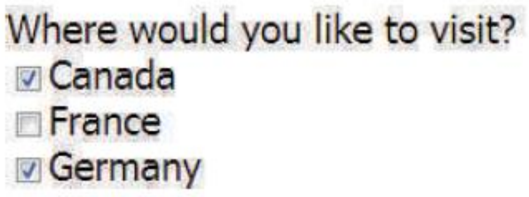
Choice Controls
Checkboxes

- Checkboxes are used for getting yes/no or on/off responses from the user.



```
<label>I accept the software license</label>
<input type="checkbox" name="accept" >
```

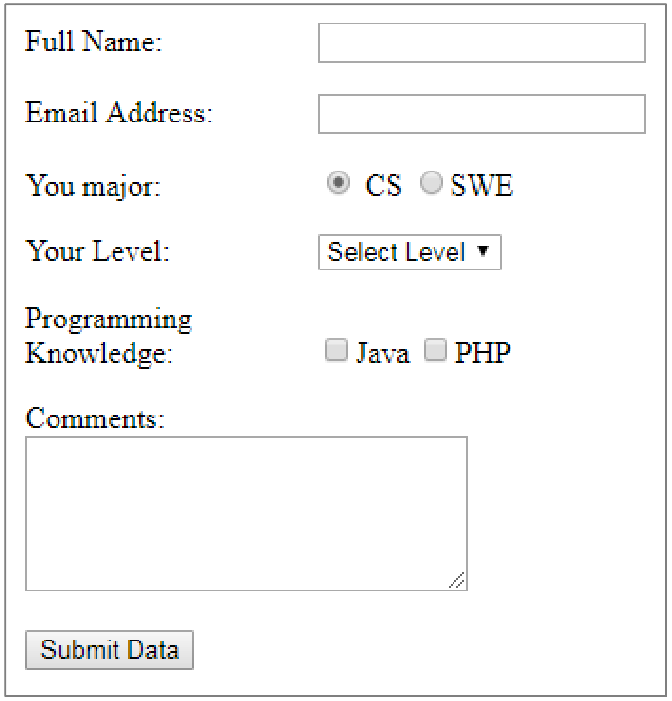
- You can also group checkboxes together by having them share the same name attribute.
  - Each checked checkbox will have its value sent to the server.



```
<label>Where would you like to visit? </label><br/>
<input type="checkbox" name="visit" value="canada">Canada<br/>
<input type="checkbox" name="visit" value="france">France<br/>
<input type="checkbox" name="visit" value="germany">Germany
```

Forms..
Example

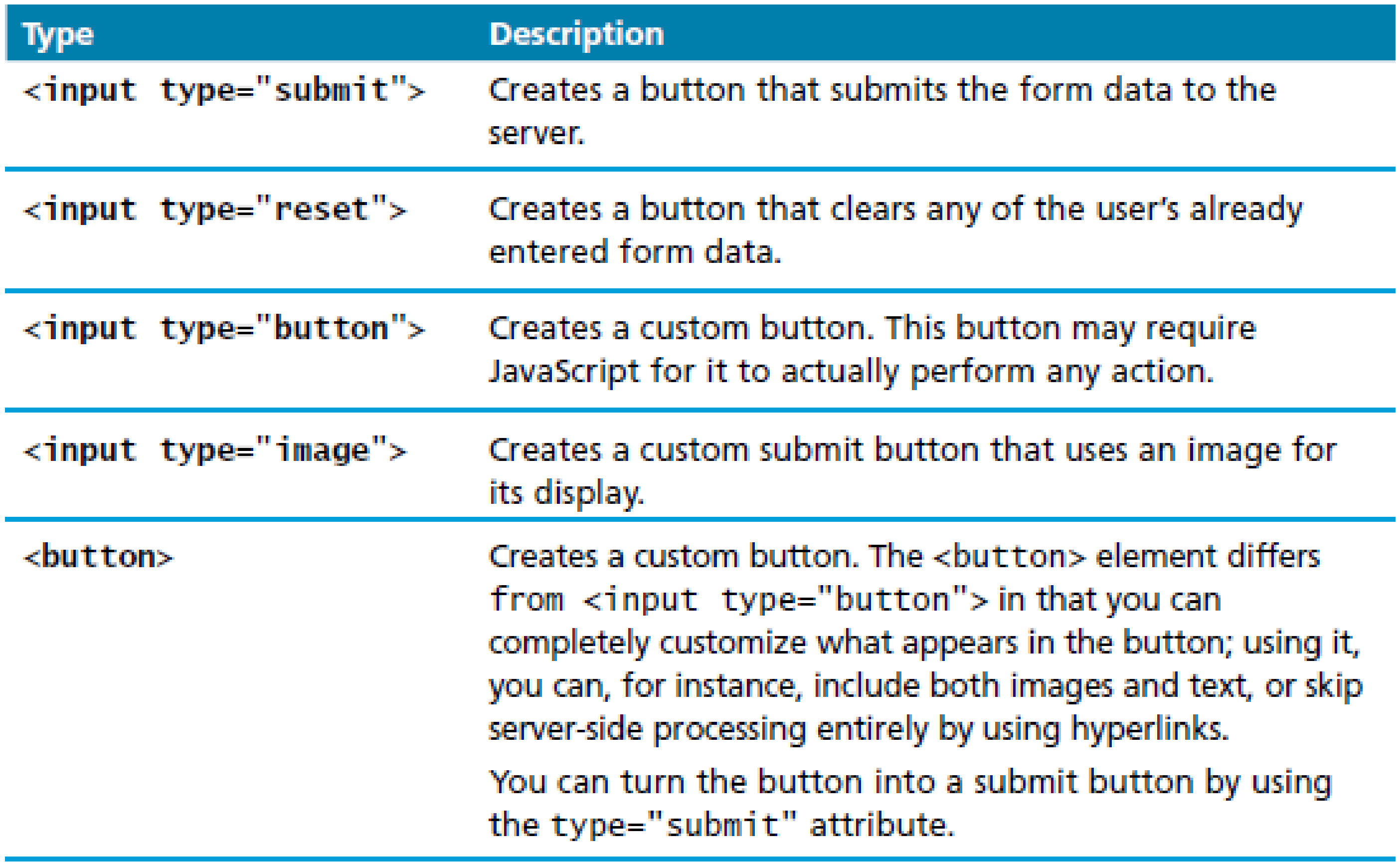
```
<form method="post" action="somescript.php">
<p>
        <label for="name">Full Name:</label>
        <input type="text" name="name">
    </p>
    <p>
        <label for="email">Email Address: </label>
        <input type="text" name="email">
    </p>
    <p>
        <label for="area"> You major: </label>
        <input type="radio" name="major" value="cs" checked> CS
        <input type="radio" name="major" value="swe">SWE
    </p>
    <p>
        <label for="level"> Your Level: </label>
        <select type="list" name="level">
            <option> Select Level  </option>
            <option value="junior">Junior</option>
            <option value="senior">Senior</option>
        </select>
    </p>
    <p>
        <label for="call">Programming Knowledge: </label>
        <input type="checkbox" name="languages" value="java">Java
        <input type="checkbox" name="languages" value="php">PHP
    </p>

    <p> <label for="comment">Comments:</label> <br>
        <textarea rows="5" cols="30"> </textarea> </p>
    <p> <input type="submit" value="Submit Data"> </p>
</form>
```



When your *from* has several checkboxes with the **same name**, make sure that they have **different values**, or the web server scripts will not be able to distinguish them.

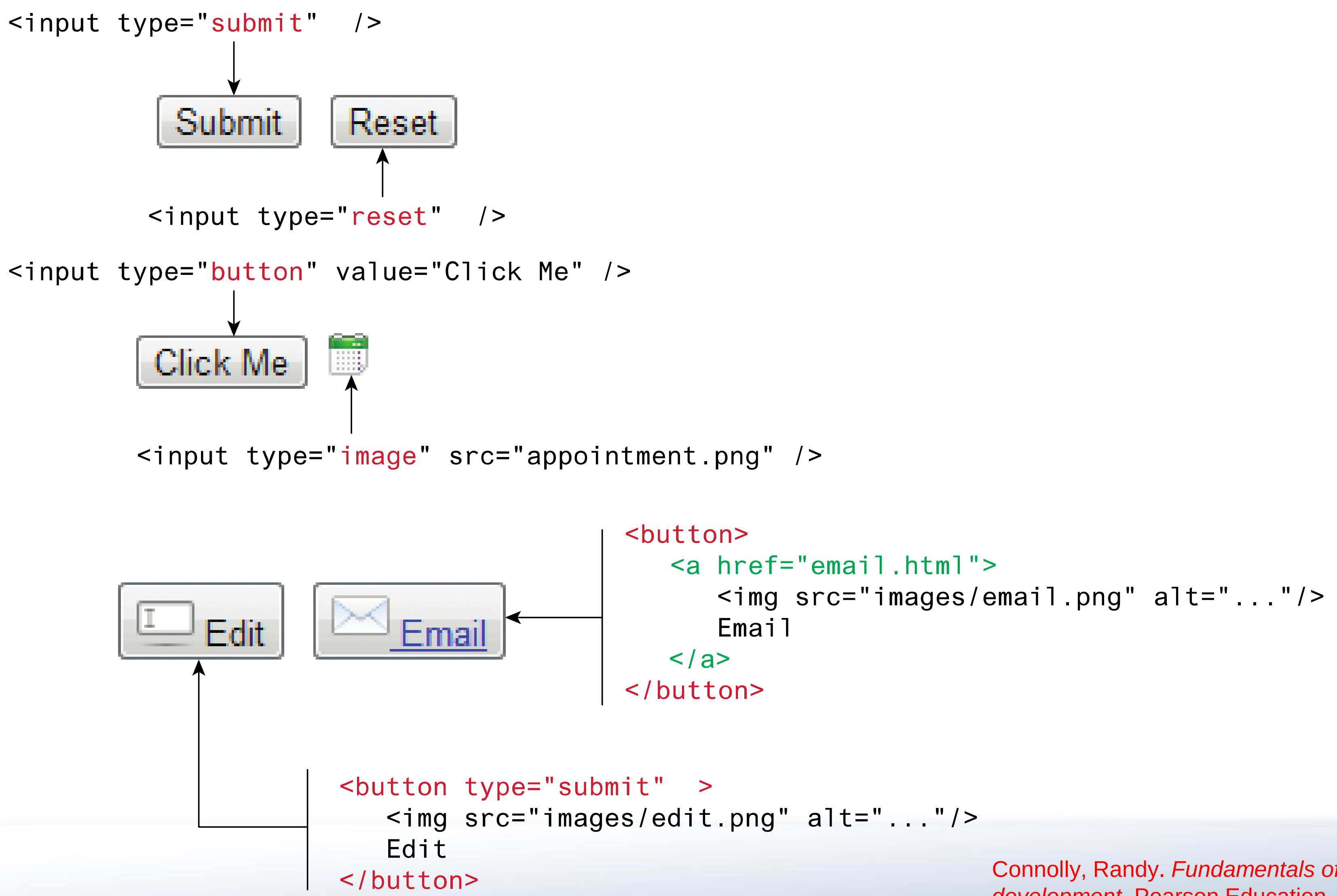
Button Controls

❑ HTML defines several different types of buttons



Type	Description
`<input type="submit">`	Creates a button that submits the form data to the server.
`<input type="reset">`	Creates a button that clears any of the user's already entered form data.
`<input type="button">`	Creates a custom button. This button may require JavaScript for it to actually perform any action.
`<input type="image">`	Creates a custom submit button that uses an image for its display.
`<button>`	Creates a custom button. The `<button>` element differs from `<input type="button">` in that you can completely customize what appears in the button; using it, you can, for instance, include both images and text, or skip server-side processing entirely by using hyperlinks. You can turn the button into a submit button by using the `type="submit"` attribute.

Button Controls



```
<input type="submit"  />
```

```
<input type="reset"  />
```

```
<input type="button" value="Click Me" />
```

```
<input type="image" src="appointment.png" />
```

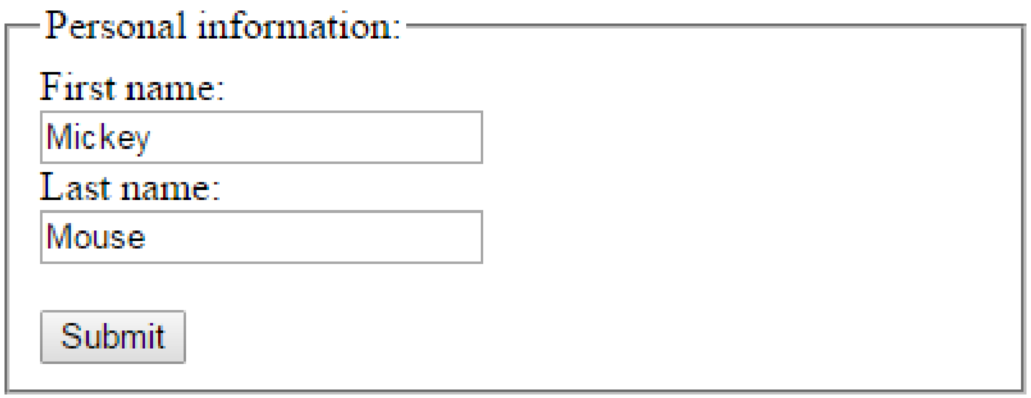
```
<button>
   <a href="email.html">
      <img src="images/email.png" alt="..."/>
      Email
   </a>
</button>
```

```
<button type="submit"  >
   <img src="images/edit.png" alt="..."/>
   Edit
</button>
```

Grouping Form Data with <fieldset>

- The <fieldset> element is used to group related data in a form.
- The <legend> element defines a caption for the <fieldset> element.

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personal information:</legend>
    First name:<br>
    <input type="text" name="firstname" value="Mickey"><br>
    Last name:<br>
    <input type="text" name="lastname" value="Mouse"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

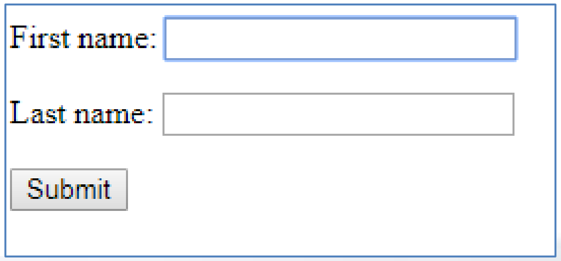


https://www.w3schools.com/html/tryit.asp?filename=tryhtml_form_legend

Specialized Controls
autofocus

- The autofocus attribute —
  - an optional attribute that can be used in only one input element on a form—
  - it automatically gives the focus to the input element, allowing the user to begin typing in that element immediately.

```
<!DOCTYPE html>
<html>
  <body>

    <form action="/action_page.php">

      <p> First name: <input type="text" name="fname" autofocus></p>
      <p> Last name: <input type="text" name="lname"> </p>

      <input type="submit">

    </form>

  </body>
</html>
```


First name:

Last name:

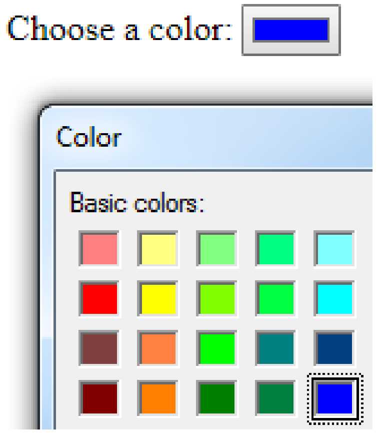
Specialized Controls
color

- ❑ The color input type enables the user to enter a color.

- ❑ In HTML5, we can have color input with simply <input type="color">.
  - ▪ The textbox should only carry value of so called "simple color" string in lowercase such as #ff0000

- ❑ With Color input type, you no longer need a complex Javascript color picker, a simple line of code below will do the work.

```
<label for="bg-color">Choose a color:</label>
<input id="background-color" type="color" />
```



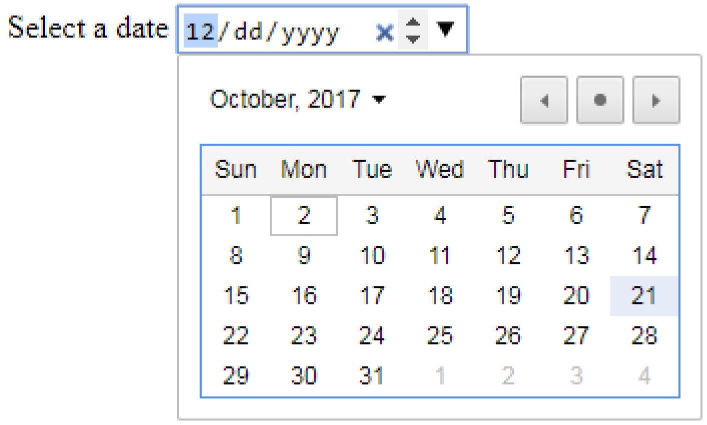
- ❑ Some browsers render the color input type as a text field in which the user can enter a hexadecamal code or a color name.
  - ▪ Other browsers like Chrome, Opera11 when you click a color input, browsers will likely display a color picker.

Specialized Controls
date

- The date input type enables the user to enter a date in the form yyyy-mm-dd.
- In HTML5, it is the job of web browser to ensure user can only enter a valid date time string into the input textbox.

```
<input id="date" type="date">
```

- Firefox and Internet Explorer display a text field in which a user can enter a date such as 2012-01-27.
- Chrome display a date control (from which you can choose a date) and a spinner control—a text field with an up-down arrow () on the right side—allowing the user to select a date by clicking the up or down arrow.
  - The start date is the current date.


Select a date 12/dd/yyyy

October, 2017

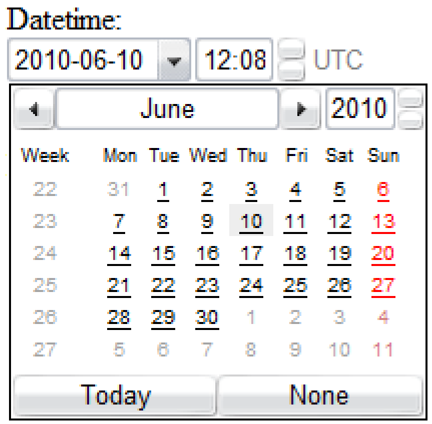
Sun	Mon	Tue	Wed	Thu	Fri	Sat
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Specialized Controls
dateTime

- We can combine the date and time by using type="datetime" for specifying a precise time on a given day

- The *datetime input type* enables the user to enter a date (year, month, day), time (hour, minute, second, fraction of a second) and the time zone set to UTC (Coordinated Universal Time or Universal Time, Coordinated).

```
<input id="entry-day-time" name="entry-day-time" type="datetime">
```



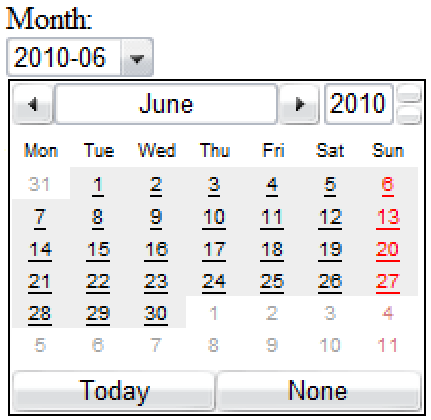
- type="datetime" is displayed in Opera 10.5
- Internet Explorer, Firefox and Chrome all display a text field.

Specialized Controls
month

- The *month input type* enables the user to enter a year and month in the format yyyy-mm, such as 2017-10.
  - for example, be used for a credit card expiry date.

- If the user enters the data in an improper format (e.g., January 2012) and submits the form, a callout stating that an invalid value was entered appears.

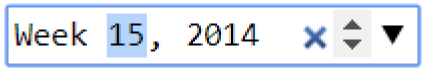
```
<input id="expiry" name="expiry" type="month" required>
```


Month:
2010-06

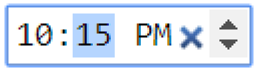
Mon	Tue	Wed	Thu	Fri	Sat	Sun
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	1	2	3	4
5	6	7	8	9	10	11

Specialized Controls
week, time

- The week input type enables the user to select a year and week number in the format yyyy-Wnn, where nn is 01–53.For example, 2012-W01 represents the first week of 2012.
  - Internet Explorer, Firefox and Safari render a text field.
  - Chrome renders an up-down control.
  - Opera renders week control with a down arrow that, when clicked, brings up a calendar for the current month with the corresponding week numbers listed down the left side.



- The *time input type* enables the user to enter an hour, minute, seconds and fraction of second
  - The HTML5 specification indicates that a time must have two digits representing the hour, followed by a colon (:) and two digits representing the minute.



Specialized Controls
autoComplete

- The *autocomplete attribute* can be used on input types to allow the browser to predict automatically the value in the user's information
  - When a user starts to type in a field, the browser should display options to fill in the field, based on earlier typed values.

```
<form action="/action_page.php" autocomplete="on">
  First name:<input type="text" name="fname"><br>
  Last name: <input type="text" name="lname"><br>
  E-mail: <input type="email" name="email" autocomplete="off"><br>
  <input type="submit">
</form>
```

- The autocomplete attribute works only if you specify a name or id attribute for the input element.
- You can enable *autocomplete* for an entire form or just for specific elements.
  - The *autocomplete attribute* works with the following <input> types: text, search, url, tel, email, password, datepickers, range, and color.

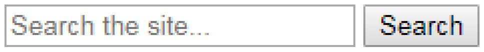
Specialized Controls
search

- The *search input type* provides a search field for entering a query.
  - This input element is functionally equivalent to an input of type text.

```
<form>
  <div>
    <input type="search" id="mySearch" name="q"
     placeholder="Search the site...">
    <button>Search</button>
  </div>
</form>
```

The <div> tag defines a division or a section in an HTML document.

It is used to group block-elements to format them with CSS.

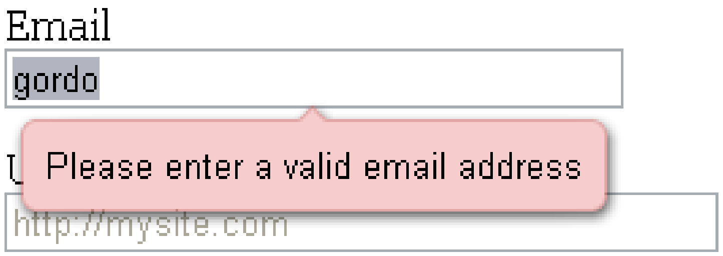


  - When the user begins to type in the search field, Chrome and Safari display an X that can be clicked to clear the field

Specialized Controls
email

- The *email input type* enables the user to enter an e-mail address or a list of e-mail addresses separated by commas (if the multiple attribute is specified).
  - Currently, all of the browsers display a text field.

```
<input type="email" name="email" required>
```


Email
gordo
Please enter a valid email address
U
http://mysite.com

- The *required attribute* forces the user to enter a value before submitting the form.
  - You can add `required` to any of the `input` types.

Specialized Controls
tel, url

- The *tel input type* enables the user to enter a telephone number—
  - Phone numbers differ around the world, to ensure that the user enters a phone number in a proper format, a *pattern attribute* was added to determine whether the required format

- The *url input type* enables the user to enter a URL.
  - The element is rendered as a text field, and the proper format is http://www.google.com
  - If the user enters an improperly formatted URL (e.g., www.google.com), the URL will not validate

- HTML5 does not check whether the URL entered is valid; rather it validates that the URL entered is in the proper format.

Specialized Controls
placeholder attribute

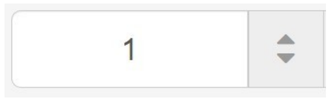
- The placeholder attribute allows you to place temporary text in a text field.
  - Placeholder provides an example of the required format
  - When the focus is placed in the text field, the placeholder text disappears
  - HTML5 supports placeholder text for only six input types— text, search, url, tel, email and password.

- HTML5 does not check whether an e-mail address entered by the user actually exists
  - it just validates that the e-mail address is in the proper format.

Specialized Controls
number

- The *number input type* enables the user to enter a numerical value— mobile browsers typically display a numeric keypad for this input type.
  - The min attribute sets the minimum valid number.
  - The max attribute sets the maximum valid number.
  - The step attribute determines the increment in which the numbers increase.
  - The value attribute sets the default value displayed in the form

```
<input type="number" min="1" max="12" step="1" value="9"
                    name="school-level">
```



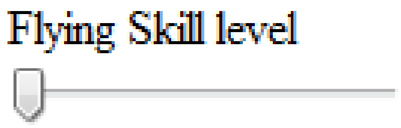
- The spinner control includes only the valid numbers.
  - Internet Explorer, Firefox and Safari display a text field in which the user can enter a number.
  - Chrome and Opera render a spinner control for adjusting the number.

Specialized Controls
range

- The range input type is similar to number but <u>more specific</u>.
  - It differs in that, the exact value isn't important. It allows browsers to offer a simpler control than for number.

- The *range input type* appears as a slider control in most current browsers.
  - You can set the minimum and maximum and specify a value.
  - The range input type is inherently self-validating when it is rendered by the browser as a slider control, because the user is unable to move the slider outside the bounds of the minimum or maximum value.

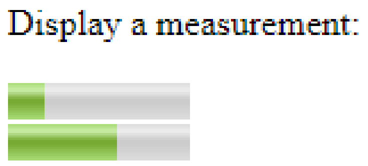
```
<input id="skill" type="range" min="1" max="100“
value="0">
```



Specialized Controls
meter , progress

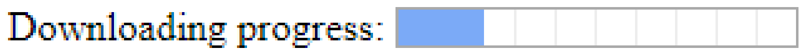
- The *meter element* renders a visual representation of a measure within a range

```
<p>Display a measurement:</p>
<meter value="2" min="0" max="10"></meter><br>
<meter value="0.6"></meter>
```



- The <progress> tag represents the progress of a task.

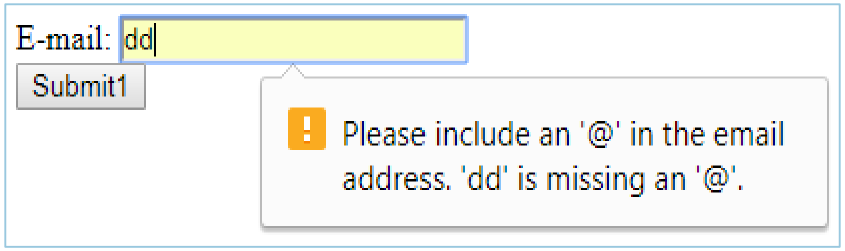
```
Downloading progress:
<progress value="22" max="100">
</progress>
```



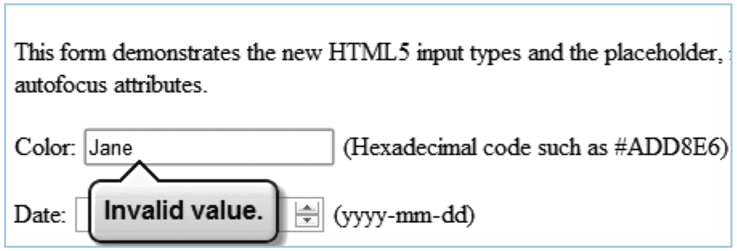
Input Self validating

- The new HTML 5 input types are self validating on the client side,
  - eliminating the need to add complicated JavaScript code to your web pages to validate user input,
  - reducing the amount of invalid data submitted and
  - consequently reducing Internet traffic between the server and the client to correct invalid input.

- When a user enters data into a form then submits the form the browser immediately checks the self-validating elements to ensure that the data is correct



Please include an '@' in the email address. 'dd' is missing an '@'.



This form demonstrates the new HTML5 input types and the placeholder, autofocus attributes.

Color: Jane (Hexadecimal code such as #ADD8E6)

Date: (yyyy-mm-dd)

- <u>The server should still validate all user input.</u>

novalidate Attribute

If you want to bypass validation, you can use the following attributes:

- The *novalidate attribute* is a boolean attribute.
  - When present, it specifies that the form-data (input) should not be validated when submitted.

```
<form action="/action_page.php" novalidate>
    E-mail: <input type="email" name="user_email">
    <input type="submit">
</form>
```

- The *formnovalidate attribute* can be used with type="submit".

```
<form action="/action_page.php">
  E-mail: <input type="email" name="userid"><br>
  <input type="submit" formnovalidate="formnovalidate" value="Submit">
</form>
```

- Safari, Safari iOS and Anroid accept the **attributes** but it don't actually work (because they *don't have form validation anyway*).

Example: https://www.w3schools.com/TAGs/tryit.asp?filename=tryhtml5_input_formnovalidate

Input Restrictions

❑ Here is a list of some common input restrictions (some are new in HTML5):

Attribute	**Description**
disabled	Specifies that an input field should be disabled
Max	Specifies the maximum value for an input field (new in HTML5)
maxlength	Specifies the maximum number of character for an input field
min	Specifies the minimum value for an input field (new in HTML5)
pattern	Specifies a regular expression to check the input value against (new in HTML5)
readonly	Specifies that an input field is read only (cannot be changed)
required	Specifies that an input field is required (must be filled out) (new in HTML5)
size	Specifies the width (in characters) of an input field
step	Specifies the legal number intervals for an input field (new in HTML5)
value	Specifies the default value for an input field

https://www.w3schools.com/html/html_form_input_types.asp