

# HTML Forms

- An HTML form is used to collect user input. The user input is most often sent to a server for processing.
- The HTML <form> element is used to create an HTML form for user input.
- The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

## The <input> Element

- The HTML <input> element is the most used form element.
- An <input> element can be displayed in many ways, depending on the type attribute.

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname"  
value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname"  
value="Doe"><br><br>  
  <input type="submit" value="Submit">  
</form>
```

# HTML Form Elements

- Each input field must have a name attribute to be submitted.
- If the name attribute is omitted, the value of the input field will not be sent at all.
- This example will not submit the value of the "First name" input field.

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" value="John"><br><br>  
  <input type="submit" value="Submit">  
</form>
```

# HTML Form Elements

## The <label> Element

- The <label> tag defines a label for many form elements.
- The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focuses on the input element.
- The <label> element also helps users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.
- The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

# HTML Form Elements

## The <select> Element

- The <select> element defines a drop-down list.
- The <option> element defines an option that can be selected.
- By default, the first item in the drop-down list is selected.
- To define a pre-selected option, add the selected attribute to the option.
- Use the size attribute to specify the number of visible values.
- Use the multiple attribute to allow the user to select more than one value:

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars" size="4" multiple>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

# HTML Form Elements

## The <textarea> Element

- The <textarea> element defines a multi-line input field (a text area).
- The rows attribute specifies the visible number of lines in a text area.
- The cols attribute specifies the visible width of a text area.

```
<textarea name="message" rows="10" cols="30">The cat was playing in the garden.</textarea>
```

- You can also define the size of the text area by using CSS.

```
<textarea name="message" style="width:200px; height:600px;">  
The cat was playing in the garden.  
</textarea>
```

# HTML Form Elements

## The <button> Element

- The <button> element defines a clickable button:

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

<h2>The button Element</h2>

<button type="button" onclick="alert('Hello World!')">Click Me!</button>

</body>
</html>
```

# HTML Form Elements

## The <fieldset> and <legend> Elements

- 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>Personalia:</legend>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

# HTML Form Elements

## The <datalist> Element

- The <datalist> element specifies a list of pre-defined options for an <input> element.
- Users will see a drop-down list of the pre-defined options.
- The list attribute of the <input> element, must refer to the id attribute of the <datalist> element.

```
<form action="/action_page.php">
  <input list="browsers">
  <datalist id="browsers">
    <option value="Edge">
    <option value="Firefox">
    <option value="Chrome">
    <option value="Opera">
    <option value="Safari">
  </datalist>
</form>
```



# HTML Form Elements

## The <output> Element

- The <output> element represents the result of a calculation (like one performed by a script).

```
<form action="/action_page.php"
  oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  0
  <input type="range" id="a" name="a" value="50">
  100 +
  <input type="number" id="b" name="b" value="50">
  =
  <output name="x" for="a b"></output>
  <br><br>
  <input type="submit">
</form>
```

# HTML Input Types

- **<input type="hidden">** defines a hidden input field (not visible to a user). A hidden field often stores what database record that needs to be updated when the form is submitted.
- **<input type="number">** defines a numeric input field.
- **<input type="url">** is used for input fields that should contain a URL address.
- **<input type="range">**  
Default Value: 0 to 100. We can change using min, max, and step attributes.
- **<input type="time">** allows the user to select a time (no time zone).

Type	Description
<input type="text">	Displays a single-line text input field
<input type="radio">	Displays a radio button (for selecting one of many choices)
<input type="checkbox">	Displays a checkbox (for selecting zero or more of many choices)
<input type="submit">	Displays a submit button (for submitting the form)
<input type="button">	Displays a clickable button

# HTML Input Types

- **<Input type="password">** defines a password field.
- **<input type="reset">** defines a reset button that will reset all form values to their default values.
- **<input type="color">** is used for input fields that should contain a color.
- **<input type="date">** is used for input fields that should contain a date.
- **<input type="email">** is used for input fields that should contain an e-mail address.
- **<input type="image">** defines an image as a submit button. The path to the image is specified in the src attribute.
- **<input type="file">** defines a file-select field and a "Browse" button for file uploads.

# HTML Input Types

## HTML Radio Buttons

- Radiobuttons let a user select ONE of a limited number of choices.

```
<form>  
  <input type="radio" id="html" name="fav_language" value="HTML">  
  <label for="html">HTML</label><br>  
  <input type="radio" id="css" name="fav_language" value="CSS">  
  <label for="css">CSS</label><br>  
  <input type="radio" id="javascript" name="fav_language" value="JavaScript">  
  <label for="javascript">JavaScript</label>  
</form>
```

# HTML Input Types

## HTML Checkboxes

- Checkboxes let a user select ZERO or MORE options of a limited number of choices.

```
<form>  
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">  
  <label for="vehicle1"> I have a bike</label><br>  
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">  
  <label for="vehicle2"> I have a car</label><br>  
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">  
  <label for="vehicle3"> I have a boat</label>  
</form>
```

# HTML Input Types

## HTML Submit Button

- The `<input type="submit">` defines a button for submitting the form data to a form-handler.
- The form-handler is typically a file on the server with a script for processing input data.
- The form-handler is specified in the form's action attribute.

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe"><br><br>  
  <input type="submit" value="Submit">  
</form>
```

# HTML Forms Attributes

- The **Action Attribute** defines the action to be performed when the form is submitted.
- Usually, the form data is sent to a file on the server when the user clicks the submit button.
- In the example below, the form data is sent to a file called "action\_page.php". This file contains a server-side script that handles the form data:
- If the action attribute is omitted, the action is set to the current page.

```
<form action="/action_page.php">  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John"><br>
```

# HTML Forms Attributes

- The **Target Attribute** specifies where to display the response that is received after submitting the form.
- The target attribute can have one of the following values:

Value	Description
<code>_blank</code>	The response is displayed in a new window or tab
<code>_self</code>	The response is displayed in the current window
<code>_parent</code>	The response is displayed in the parent frame
<code>_top</code>	The response is displayed in the full body of the window
<code>framename</code>	The response is displayed in a named iframe

```
<form action="/action_page.php" target="_blank">
```



# HTML Forms Attributes

- The **Method Attribute** specifies the HTTP method to be used when submitting the form data.
- The form-data can be sent as URL variables (with method="get") or as HTTP post transaction (with method="post").
- The default HTTP method when submitting form data is GET.
- **Get**
  - Appends the form data to the URL in name/value pairs
  - NEVER use GET to send sensitive data (the submitted form data is visible in the URL!)
  - The length of a URL is limited (2048 characters)
  - Useful for form submissions where a user wants to bookmark the result

```
<form action="/action_page.php" method="get">
```

# HTML Forms Attributes

- **Post**

- Appends the form data inside the body of the HTTP request (the submitted form data is not shown in the URL)
- POST has no size limitations, and can be used to send large amounts of data.

```
<form action="/action_page.php" method="post">
```

- **Autocomplete**

- The autocomplete attribute specifies whether a form should have autocomplete on or off.
- When autocomplete is on, the browser automatically complete values based on values that the user has entered before.

```
<form action="/action_page.php" autocomplete="on">
```

# HTML Forms Attributes

- **Novalidate**

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

Attribute	Description
<u>accept-charset</u>	Specifies the character encodings used for form submission
<u>action</u>	Specifies where to send the form-data when a form is submitted
<u>autocomplete</u>	Specifies whether a form should have autocomplete on or off
<u>enctype</u>	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
<u>method</u>	Specifies the HTTP method to use when sending form-data
<u>name</u>	Specifies the name of the form
<u>novalidate</u>	Specifies that the form should not be validated when submitted
<u>rel</u>	Specifies the relationship between a linked resource and the current document
<u>target</u>	Specifies where to display the response that is received after submitting the form

# HTML Input Attributes

- The input “**value**” attribute specifies an initial value for an input field.
- The input “**readonly**” attribute specifies that an input field is read-only.
- The input “**disabled**” attribute specifies that an input field should be disabled.
- The input “**size**” attribute specifies the visible width, in characters, of an input field (default value for size is 20).
- The input “**maxlength**” attribute specifies the maximum number of characters allowed in an input field.
- The input “**min and max**” attributes specify the minimum and maximum values for an input field.
- The input “**multiple**” attribute specifies that the user is allowed to enter more than one value in an input field.

# HTML Input Attributes

- The input “**pattern**” attribute specifies a regular expression that the input field's value is checked against, when the form is submitted.
- The input “**placeholder**” attribute specifies a short hint that describes the expected value of an input field (a sample value or a short description of the expected format).
- The input “**required**” attribute specifies that an input field must be filled out before submitting the form.
- The input step attribute specifies the legal number intervals for an input field. This attribute can be used together with the max and min attributes to create a range of legal values.
- The input autofocus attribute specifies that an input field should automatically get focus when the page loads.

# HTML Input Attributes

- The input “**height and width**” attributes specify the height and width of an `<input type="image">` element.
- The input “**list**” attribute refers to a `<datalist>` element that contains pre-defined options for an `<input>` element.
- The input “**autocomplete**” attribute specifies whether a form or an input field should have autocomplete on or off.
- Autocomplete allows the browser to predict the value. When a user starts to type in a field, the browser should display options to fill in the field, based on earlier typed values.

**Tools for Building Web Page Front Page:** Study on your own