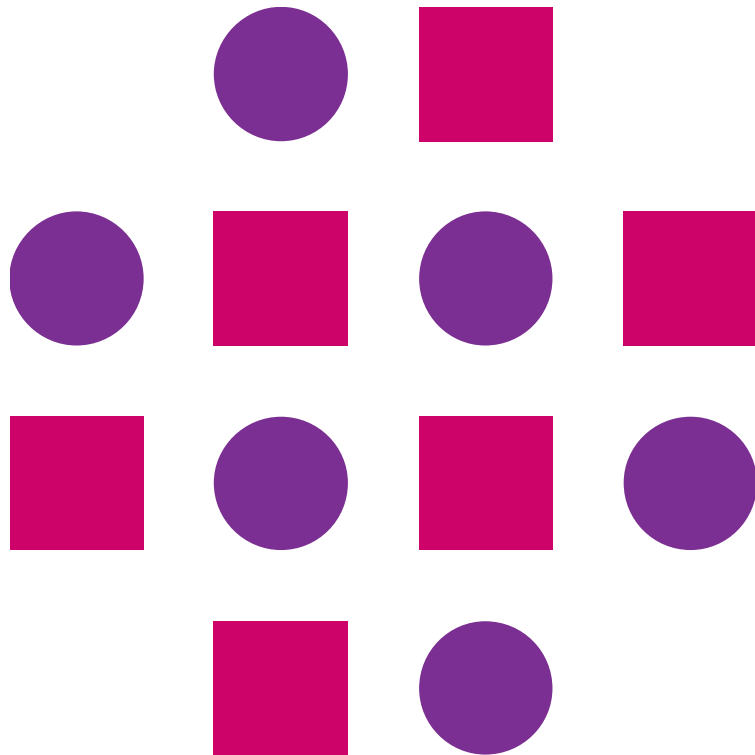


# Forms

## Hypertext Markup Language 5 (HTML5)



# Table of Content

- Forms
- Validation
- Attributes





# Forms

(Hypertext Markup Language 5)

# Forms

- Forms are used to collect information from users
- An HTML form contains form element
- Form elements are different type of input elements like text fields , check box ,radio button,list box and submit buttons etc
- Forms are defined by `<form>.....</form>` tag

# The Action Attribute

- The action attribute tells the browser what to do with the content of the form and the method attribute tells the browser how to handle it
- The method attribute have two values get or post
- Syntax :

`<form action ="url " method="post">`

# The Method Attribute

## Get

- Data is visible to everyone in the URL
- Can be bookmarked
- The parameter data is limited
- Get is less secure compared to the post
- Parameter remain in browser history

## Post

- Data is not displayed in the URL
- Cannot be bookmarked
- Can send parameters including uploading files ,to the server
- Post is little safer than Get
- Parameters are not saved in browser history

# Forms Example

```
<form>
```

```
Firstname :<input type="text" name="firstname"/><br/>
```

```
LastName :<input type="text" name="lastname"/><br/>
```

```
<input type="submit" value="Submit Now"/>
```

```
</form>
```

# The <input> element

- The input fields are defined by the <input> element
- The type attribute defines which input state you are using
- The <input> element is an empty element and therefore it is self-closing / at the end



# The type attribute

- `<input type="text">`
  - Defines a one-line text input field
- `<input type="radio">`
  - Defines a radio button (to select one of many choices)
- `<input type="submit">`
  - Defines a submit button (to submit the form)

# Input Type

- Defining a password field
  - `<input type="password">`
- Defining a reset button that will reset all form values to their default values
  - `<input type="reset">`

# Input Type CheckBox

- CheckBox is suitable choice when a lot of options to the user and user has to choose one or more items.
  - `<input type = "checkbox">`

# Input Type CheckBox

## (Example)

```
<form method="post">
```

```
<fieldset>
```

```
<legend>What is Your Favourite Pet?</legend>
```

```
<input type="checkbox" name="animal" value="Cat" />Cats <br />
```

```
<input type="checkbox" name="animal" value="Dog" />Dogs<br />
```

```
<input type="checkbox" name="animal" value="Bird" />Birds <br />
```

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

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

```
</fieldset>
```

```
</form>
```

# Input Type Radio

## (Example)

```
<form>
```

```
<input type="radio" name="gender" value="male" checked /> Male  
<br/>
```

```
<input type="radio" name="gender" value="female"/> Female  
<br/>
```

```
</form>
```

# The <select> element

- The <select> element defines a drop-down list
- The <option> element defines a option that can be selected
- The selected attribute is added to define a pre-selected option

# The <select> element

## (Example)

```
<select name="Ice Cream Flavors">  
<option value="Butter Scotch" selected>Butter Scotch</option>  
<option value="Vanilla">Vanilla</option>  
<option value="Strawberry">Straw Berry</option>  
</select>
```

# The <textarea> element

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



# The <textarea> element

```
<textarea cols="20" rows="5">  
</textarea>
```

# Grouping Form Data

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

# Grouping Form Data (Example)

```
<form>
<fieldset>
<legend>User info</legend>
<label for="username">Username:</label>
<input type="text" name="username"><br />
<label for="password">Password:</label>
<input type="password"><br /></fieldset>
<input type="submit" value="Submit now" />
</form>
```

# The <button> element

- The button element defines a clickable button
- **Example :**
  - `<button type="button" onclick="alert('welcome to html5 world')">click </button>`

# Form Elements

- HTML5 added some new Form Elements
  - datalist
  - keygen
  - output

# The <datalist> Element

- The <datalist> element specifies a list of pre-defined option for an <input> element
- These pre-defined option user can see in drop-down list
- The list attribute of the <input> element, must refer to the id attribute of the <datalist> element
- The <datalist> element bind together by using <input> element

# The <datalist> Element Example

```
<input list="states"/>  
<datalist id="states">  
  <option value="Jharkhand">  
  <option value="Karnataka">  
  <option value="UP">  
  <option value="Tamilnadu">  
</datalist>
```

# The <keygen> element

- The main purpose of the <keygen> element is to give the secure or validate way to authenticate users
- It is specified a key-pair generator field in a form
- Two keys (public and private) are generated at the time of form submission
- The private key is stored locally, whereas public key is sent to the server
- The client key generate the client certificate for use in future



# The <keygen> element

## (Example)

```
<form action="default_keygen.asp" method="get">  
Username: <input type="text" name="user_name" />  
Encryption format: <keygen name="security" />  
<input type="submit" />  
</form>
```

# The <output> element

- The <output> element contain the result of any calculation

# The <output> element

## (Example)

```
<form oninput="y.value=parseInt(b.value)
+parseInt(c.value)">0
<input type="range" name="b" value="55" />150
+<input type="number" name="c" value="55" />=<output
name="y" for="b c"></output>
</form>
```

# Validation

(Hypertext Markup Language 5)

# Number

- The `<input type = "number">` defines a numeric input field
- Example

```
<form>
```

Marks (between 50 and 90):

```
<input type = "number" name = "Marks" min = "50" max = "90">
```

```
</form>
```

**\*Here min and max are input restrictions**

# Validating date

- Date element provides a handy drop down calendar to pick the date from
- Date element provides six different ways of defining dates
  - Date
  - Month
  - Week
  - Time
  - date+time
  - date +time-time zone

# Date

- The `<input type="date">` is used for input fields that should contain a date.
- Example

```
<form>  
Enter a date after 2005-01-01  
<input type="date" name="JoinDate" min="2005-02-02">  
</form>
```

# Time

- `<input type = "time">` allows user to select a time (no time zone)
- Example

```
<form>  
Select Time :  
<input type="time" name ="user_time">  
<input type="submit">  
</form>
```



# Color

- The `<input type="color">` is used for input fields that should contain a color.
- Example

```
<form>
```

Select Your favorite Color :

```
<form>
```

```
<input type ="color" name="colorpicker">
```

```
</form>
```

# Email

- Email can be automatically validated at submission
- The `<input type="email">` is used for input fields that should contain an email address
- Example

```
<form>  
Email <input type="email" name="email">  
<input type="submit">  
</form>
```

# Attributes

(Hypertext Markup Language 5)

# The autofocus attribute

- The autofocus attribute specifies the input field should get automatically get focus when the page loads
- Example  
`<input type = "text" name = "search" autofocus />`

# The Required Attribute

- The Required attribute specifies that an input field must be filled out before submitting the form
- Example  
`<input type = "text" name = "search" required />`

# The placeholder Attribute

- The placeholder attribute specifies a hint that describes the expected value of an input field
- The hint is displayed in the input field before the user enters a value

# The placeholder Attribute

```
<form>
```

```
First name : <input type="text" name="firstname"  
placeholder="Jane"/> <br>
```

```
Surname : <input type="text" name="surname"  
placeholder="Doe"/> <br>
```

```
<input type="submit" value="Submit now" />
```

```
</form>
```

# The autocomplete Attribute

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



# The autocomplete Attribute

```
<form method="post" autocomplete="on">
```

```
User Name : <input type="text" name="username"> <br>
```

```
Password : <input type="password" name="passwd"  
autocomplete="off"><br>
```

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

```
</form>
```

Web Stack Academy (P) Ltd

#83, Farah Towers,  
1st floor, MG Road,  
Bangalore - 560001

M: +91-80-4128 9576

T: +91-98862 69112

E: [info@www.webstackacademy.com](mailto:info@www.webstackacademy.com)

*Thank  
you*