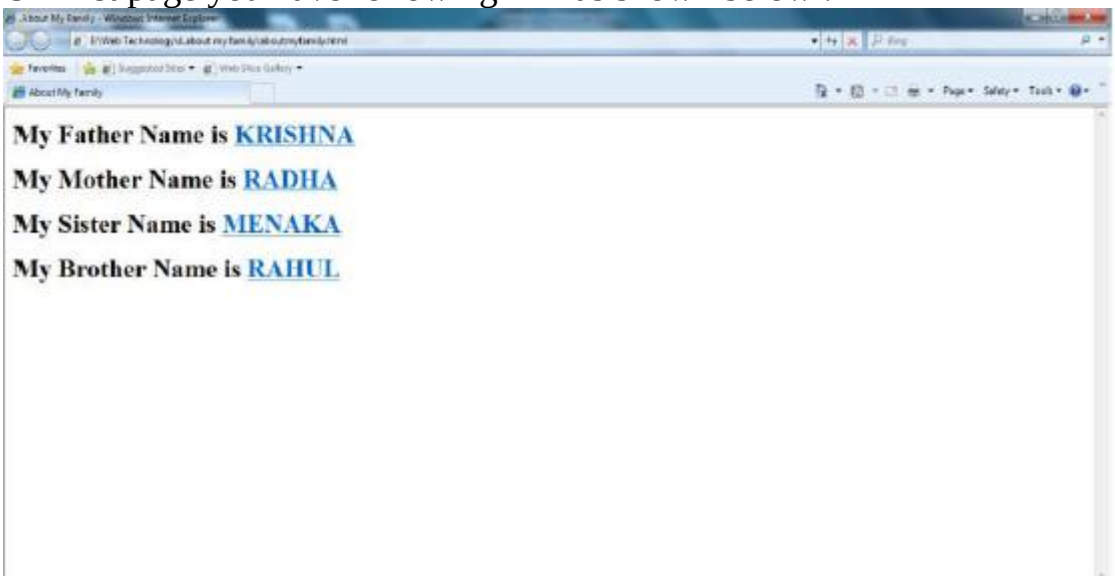
Practical -4

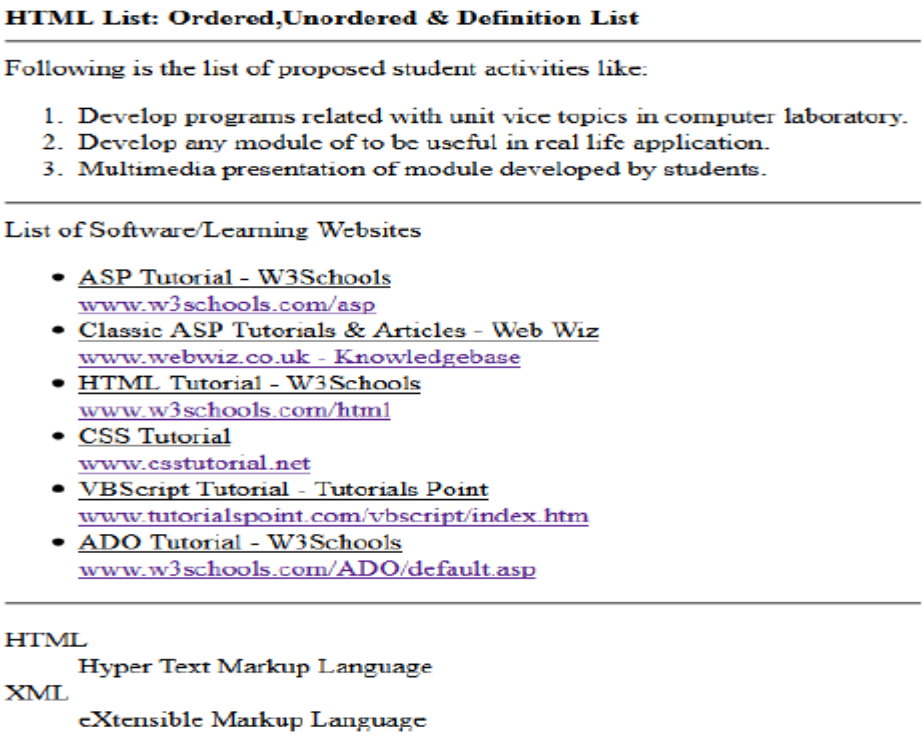
Implementing Link and Form in HTML

1. Design a simple webpage about your college. Create another webpage named address.html containing the college address. Give links from college page to address.htm.
2. Design an attractive webpage about India. Provide details about the Indian freedom movement at the bottom of the webpage. Also create another webpage containing the list of states in India, named 'states.html'. Create two links in the main webpage – one to link to the bottom of the webpage where details about freedom movement is given and another to the webpage 'states.html'.
3. Create a HTML webpage about my family using hyperlink.
   On first page you have following link as shown below :



   Now create father.html, mother.html, sister.html and brother.html.
   Each of these page would have image of family member with one line showing your family member's quality. For example "He is my father, a philospher and my mentor".
   When the appropiate link is clicked the corresponding family member page whould be open.
4. Create a web page with an appropriate image towards the left hand side of the page, when user clicks on the image another web page should open.
5. Write a HTML program for demonstrating Hyperlinks. Navigation within the page.
6. Write a HTML program to demonstrate all target attribute of <a> tag.[for example target : _self, _blank, _top, _parent, iframe]

7. Create a web page for internal links; when the user clicks on different links on the web page it should go to the appropriate locations/sections in the same page. [Hint:
    1. First give name for the each section of the page by using name attribute of the <a>tag (i.e)
    2. To provide link for the sections use href="#name of section" in<a> tag.
    3. The ouput is shown below:



HTML List: Ordered,Unordered & Definition List

Following is the list of proposed student activities like:

1. Develop programs related with unit vice topics in computer laboratory.
2. Develop any module of to be useful in real life application.
3. Multimedia presentation of module developed by students.

List of Software/Learning Websites

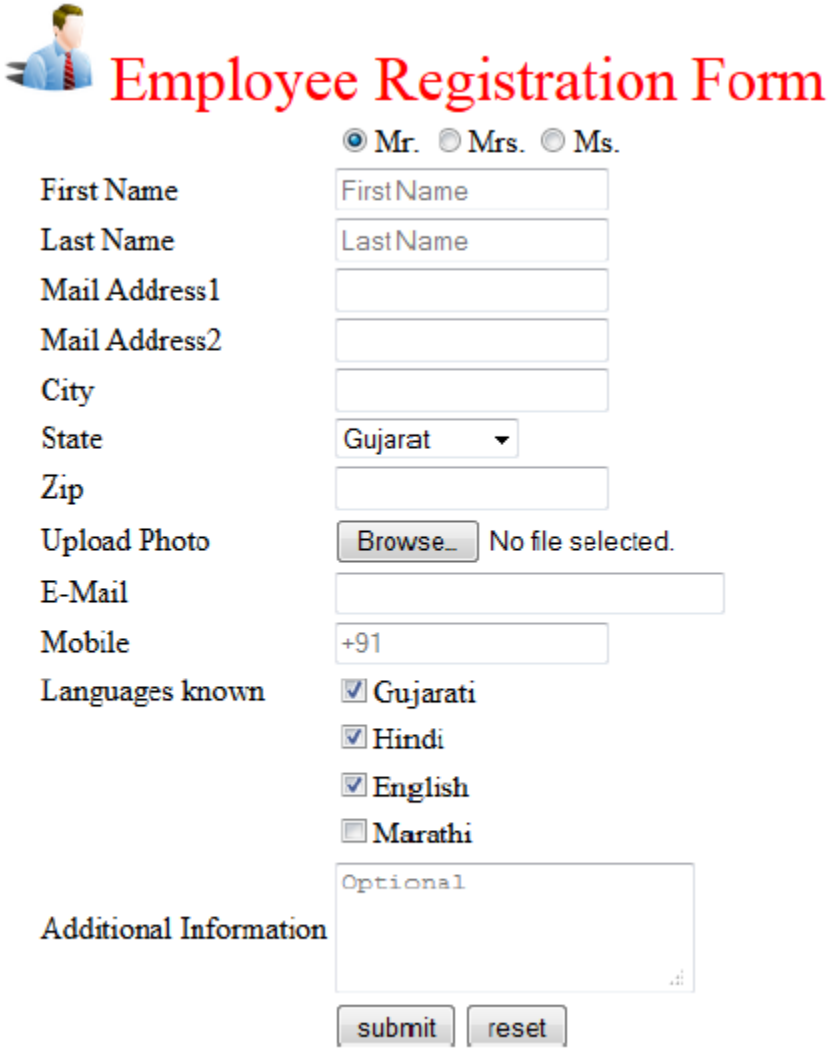
- ASP Tutorial - W3Schools
  www.w3schools.com/asp
- Classic ASP Tutorials & Articles - Web Wiz
  www.webwiz.co.uk - Knowledgebase
- HTML Tutorial - W3Schools
  www.w3schools.com/html
- CSS Tutorial
  www.csstutorial.net
- VBScript Tutorial - Tutorials Point
  www.tutorialspoint.com/vbscript/index.htm
- ADO Tutorial - W3Schools
  www.w3schools.com/ADO/default.asp

HTML
Hyper Text Markup Language
XML
eXtensible Markup Language

8. Consider that your school is hosting an inter-school IT fair. Design a form webpage that contains a form for accepting registrations. The form page should contain facility to enter school name, user name, password and a mobile phone number. It should also contain buttons for saving and clearing data entered.

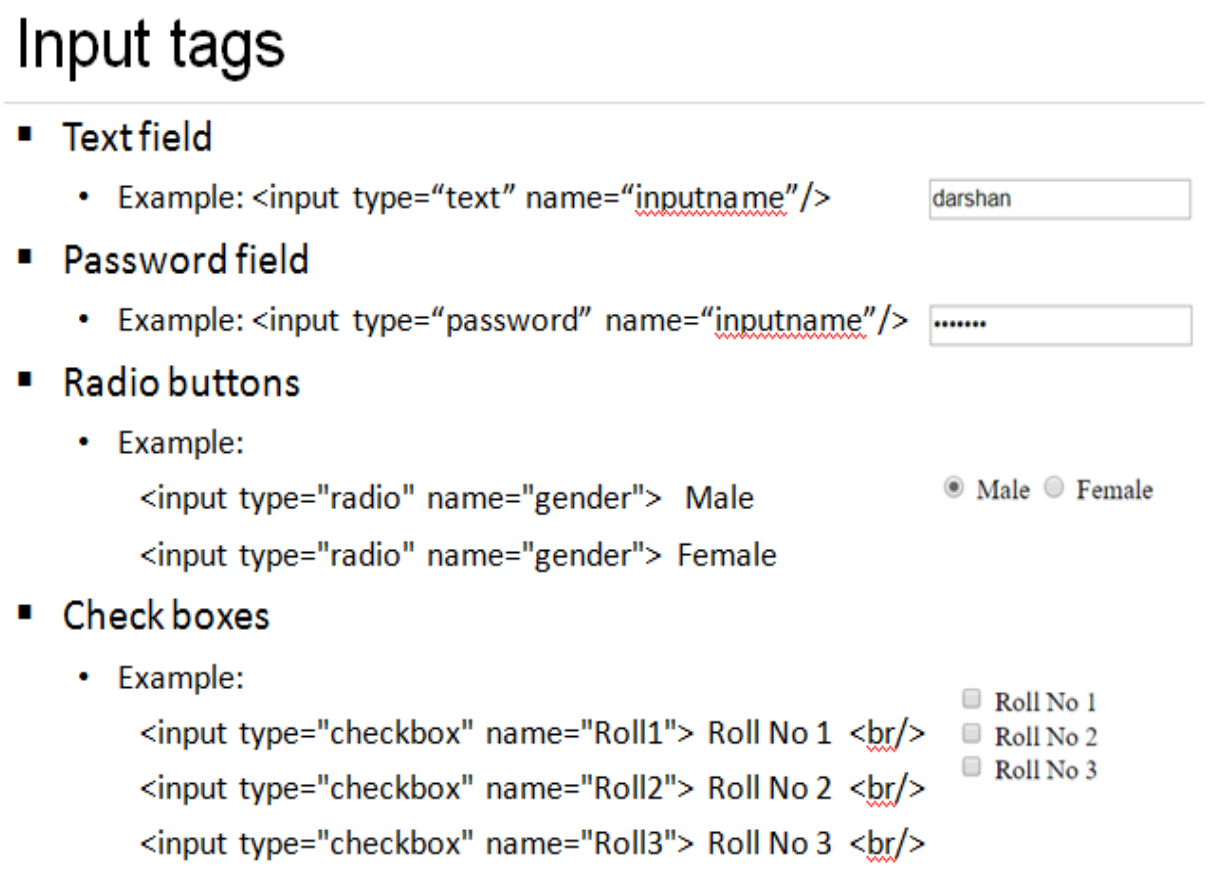
9. Create Registration Form shown as below

10. Create Employee Registration Form as shown below



Creating a form Example

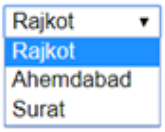
```
<!DOCTYPE HTML >
<HTML>
<HEAD>
        <TITLE>Form</TITLE>
</HEAD>
<BODY>
        <!-- starting of form -->
        <FORM>
                Username: <input type="text" name="user"/><br/>
                <input type="submit">
        </FORM>
        <!--ending of form -->
</BODY>
</HTML>
```



Input tags (cont.)

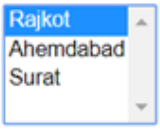
- Dropdown list
  - <select> tag is used to create a drop-down list in HTML.
  - <option> tags inside the <select> tag define the available options in the list.
  - Example:

```
<select>
  <option value="1">Rajkot</option>
  <option value="2">Ahemdabad</option>
  <option value="3">Surat</option>
</select>
```



  - Example (multiple select):

```
<select multiple="multiple">
  <option value="1">Rajkot</option>
  <option value="2">Ahemdabad</option>
  <option value="3">Surat</option>
</select>
```



Tag	Description
<form>	The <form> tag is used to create an HTML form for user input. The <form> element can contain one or more of the following form elements: <input> <textarea> <button> <select> <option> <optgroup> <fieldset> <label> <output> Attributes are : accept-charset, action, autocomplete, enctype, method, name, nonvalidate, target enctype : Specifies how the form-data should be encoded when submitting it to the server (only for method="post") (application/x-www-form-urlencoded multipart/form-data, text/plain)

action : Specifies where to send the form-data when a form is submitted (URL)

method : Specifies the HTTP method to use when sending form-data (get, post)

<input>

The <input> tag specifies an input field where the user can enter data. The <input> element is the most important form element. The <input> element can be displayed in several ways, depending on the type attribute.

The different input types are as follows:

- <input type="button">
- <input type="checkbox">
- <input type="color">
- <input type="date">
- <input type="datetime-local">
- <input type="email">
- <input type="file">
- <input type="hidden">
- <input type="image">
- <input type="month">
- <input type="number">
- <input type="password">
- <input type="radio">
- <input type="range">
- <input type="reset">
- <input type="search">
- <input type="submit">
- <input type="tel">
- <input type="text"> (default value)
- <input type="time">
- <input type="url">
- <input type="week">

attributes are : alt, autocomplet, autofocus, checked, disable, formaction, formmethod, max, maxlenght, min, minlenght, multiple, readonly, size, required, type, value

<button>

The <button> tag defines a clickable button. Inside a <button> element you can put text (and tags like <i>, <b>, ,
, <img>, etc.). That is not possible with a button created with the <input> element!

Tip: Always specify the type attribute for a <button> element, to tell browsers what type of button it is.

Attributes are : autofocus, disable, form, formaction, formenctype, formmethod,

	formvalidate, formtarget, name, type, value
<select> multiple-choice element	The `<select>` element is used to create a drop-down list. The `<select>` element is most often used in a form, to collect user input. The `name` attribute is needed to reference the form data after the form is submitted (if you omit the `name` attribute, no data from the drop-down list will be submitted). The `id` attribute is needed to associate the drop-down list with a label. The <option> tags inside the `<select>` element define the available options in the drop-down list. Attributes are : disabled, name, size, autofocus, form, multiple
<option>	The `<option>` tag defines an option in a select list. `<option>` elements go inside a <select>, <optgroup>, or <datalist> element. **Note:** The `<option>` tag can be used without any attributes, but you usually need the **value** attribute, which indicates what is sent to the server on form submission. Attributes are : label, disabled, selected, value
<optgroup>	The `<optgroup>` tag is used to group related options in a <select> element (drop-down list). If you have a long list of options, groups of related options are easier to handle for a user. Attributes are : label, disabled.
<textarea>	The `<textarea>` tag defines a multi-line text input control. The `<textarea>` element is often used in a form, to collect user inputs like comments or reviews. A text area can hold an unlimited number of characters, and the text renders in a fixed-width font (usually Courier). The size of a text area is specified by the `<cols>` and `<rows>` attributes (or with CSS). The `name` attribute is needed to reference the form data after the form is submitted (if you omit the `name` attribute, no data from the text area will be submitted). The `id` attribute is needed to associate the text area with a label. Attributes are : cols, rows, disabled, name, readonly, accesskey, autofocus, dirname, maxlength, placeholder, required, wrap.
<label>	The `<label>` tag defines a label for several elements: • <input type="checkbox"> • <input type="color"> • <input type="date"> • <input type="datetime-local"> • <input type="email"> • <input type="file"> • <input type="month"> • <input type="number">

- <input type="password">
- <input type="radio">
- <input type="range">
- <input type="search">
- <input type="tel">
- <input type="text">
- <input type="time">
- <input type="url">
- <input type="week">
- <meter>
- <progress>
- <select>
- <textarea>

Proper use of labels with the elements above will benefit:

- Screen reader users (will read out loud the label, when the user is focused on the element)
- Users who have difficulty clicking on very small regions (such as checkboxes) - because when a user clicks the text within the `<label>` element, it toggles the input (this increases the hit area)

attributes are : for, form