3.1.2 Button Element

Q. Explain Button element with example .

In HTML following are three types of buttons that we can create using <input> element :

1. **Submit** - Submit buttons send form data to whatever action has been mentioned in the action attribute of the <form> element. Set the *type* attribute of the <input> tag to "*submit*" in order to place a submit button on a web page.

   Example - <input type="submit" value="Send" />

2. **Reset** - A reset button allows users to clear their web form data. It wipes values from all fields by "resetting" the form to its default appearance.

   Example - <input type="reset" value="Reset" />

3. **Button** – Button will create simple push buttons, which can be programmed to control custom functionality anywhere on a webpage as required when assigned an event handler function (typically for the click event).

   Example - **<input type="button" value="Click" >**

Attributes :

Attribute	Value	Explanation
type=" "	button submit reset	Creates a general purpose push button.
value=" "	button text	Value is the text displayed on the button.
Name=" "	field name	The field name is used to identify the form field.

Program 1:

```
<!DOCTYPE html>

<html>

<head>

  <title> Button Element  </title>

        <script language="javasript" type="text/javascript">

        </script>


</head>

<body>

  <form name="buttondemo">


        <input type="submit" name="b1" value="SUBMIT"> <br>

        <input type="reset" name="b2" value="RESET"> <br>

        <input type="button" name="b3" value="CLICK">


  </form>

</body>

</html>
```

3.1.3 Text

Q. Explain Text element with example

The INPUT element defines an input field. A textbox is created by specifying the type attribute to "text".

Attributes :

Attribute	Value	Description
type=" "	text	It creates a textbox on form
name=" "	field name	The field name is used to identify the form field.
size=" "	number of characters	The input field width is specified by the number of characters.
maxlength=" "	number of characters	Specifies the maximum number of characters allowed in the input field.
value=" "	initial value	Specifies the initial text displayed in the input field.

Program 2:

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Text Element  </title>
        <script language="javasript" type="text/javascript">
        </script>
  </head>

        <body>
                <form>
                        Enter your Name:
                        <input type="text" name="t1" size="15"> <br>
                        Enter your Mobile No:
                        <input type="text" name="t2" size="15" maxlength="10"> <br>
                        Enter your Address:
                        <input type="text" name="t3" size="15" value="Enter text here"> <br>
                </form>
          </body>
</html>
```

3.1.4 TextArea

Q. Explain TextArea element with example.

The TEXTAREA element defines a multi-line text area.

Attributes :

Attribute	Value	Description
name=" "	field name	The field name is used to identify the form field.
cols=" "	number	Specifies the number of visible columns in the text area. (The text area's width)
rows=" "	number	Specifies the number of visible rows in the text area. (The text area's height)
wrap=" " (Extension)	hard	the input text is wrapped (and the submitted text contains line breaks)
	soft	the input text is wrapped (but the submitted text doesn't contain line breaks)
	off	the input text is not wrapped

Program 3:

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Textarea Demo  </title>
        <script language="javasript" type="text/javascript">
        </script>
  </head>

        <body>
                <form>
                        Feedback: <br>
                        <textarea name="ta1" cols="20" rows="5">Enter your Feedback...
                        </textarea>

                </form>
          </body>
</html>
```

3.1.5 Checkbox

The INPUT element defines an input field. When you specify "checkbox" for the type attribute of this element, a checkbox is created.

Attributes :

Attribute	Value	Explanation
type=" "	checkbox	Creates a checkbox on the form.
name=" "	field name	The field name is used to identify the form field. Several checkboxes can share the same field name.
value=" "	initial value	This value is submitted to the server when selected.
checked	checked	That checkbox is checked in the initial state.

Program 4:

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Checkbox Demo  </title>
        <script language="javasript" type="text/javascript">
        </script>
  </head>

        <body>
                <form>

                        Select Subjects: <br>
                        <input type="checkbox" name="subject1" value="PIC">C
                        <input type="checkbox" name="subject2" value="OOP">C++
                        <input type="checkbox" name="subject3" value="AJP">Java
                        <input type="checkbox" name="subject4" value="WPD" checked>HTML

                </form>
          </body>
</html>
```

3.1.6 Radio Button

The INPUT element defines an input field. When you specify "radio" for the type attribute of this element, a radio button is created.

Attributes

Attribute	Value	Description
type=" "	radio	Creates a radio button on the form.
name=" "	field name	The field name is used to identify the form field. Several radio buttons can share the same field name, and only one can be selected within that.
value=" "	initial value	This value is submitted to the server when selected.
checked	checked	That button is checked in the initial state.

Program 5:

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Radio Demo  </title>
        <script language="javasript" type="text/javascript">
        </script>
  </head>

        <body>
                <form>

                        Subject Choice: Select Elective <br>
                        <input type="radio" name="r1" value="yes">Client Side Scripting Language <br>
                        <input type="radio" name="r1" value="no">Advance Computer Network


                </form>
          </body>
</html>
```

3.1.7 Select Element

The SELECT element defines a selectable list, and the OPTION element is used to define a list item.

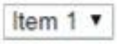
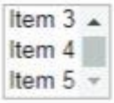
Example :

```
<select name="example">
<option value="item1">Item 1</option>
</select>
```

The OPTION element is placed inside the SELECT element.

Attribute :

Attribute	Value	Designation
name=" "	field name	The field name is used to identify the form field.

size=" "	number	Specifies the number of visible items in the list.
		size="1" : 
		size="3" : 

multiple	multiple	Multiple items can be selected at a time. Use the SHIFT or CTRL key to select multiple items.

Option element :

Example - <option value="item1" selected>Item 1</option>

Attributes :

Attribute	Value	Description
value=" "	initial value	This value is submitted to the server when selected.
selected	selected	That item is selected in the initial state.

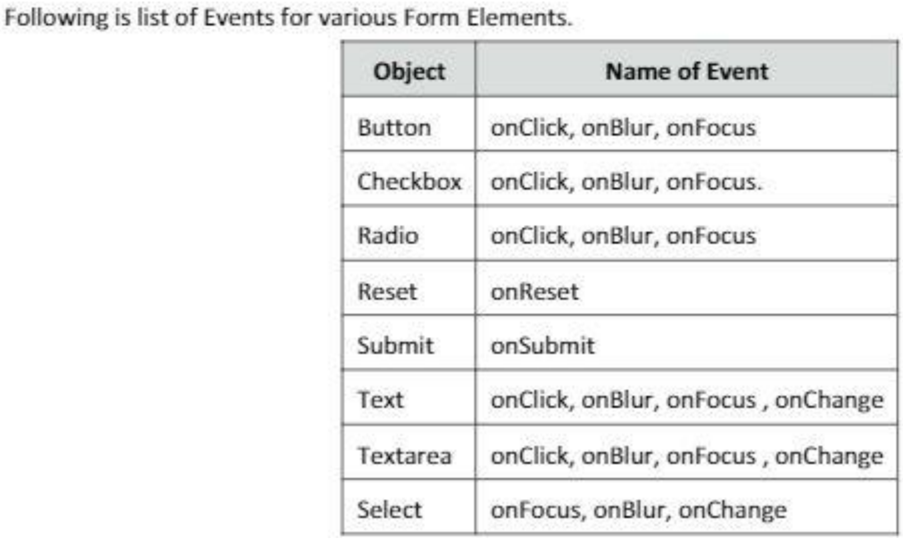
Program 6:

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Select Demo  </title>
        <script language="javasript" type="text/javascript">
        </script>
  </head>

        <body>
                <form>
                        Select Colours: <br>
                        <select name="s1">
                        <option value="white">White   </option>
                        <option value="red">Red       </option>
                        <option value="yellow">Yellow </option>
                        <option value="blue">Blue     </option>
                        <option value="green">Green   </option>

                        </select>
                </form>
          </body>
</html>
```

Form Events:

Form events are those events that get trigerred when a user interacts with a form in a website. The purpose of using form events is to make form filling process interactive and informative for the user.

Event Performed	Event Handler	Description
Change	onchange	When the user modifies or changes the value of a form element
Select	onselect	When the user selects the form element
Blur	onblur	When the focus is away from a form element
Focus	onfocus	When the user focuses on an element
Submit	onsubmit	When the user submits the form
Reset	onreset	When the user resets the form



Following is list of Events for various Form Elements.

Object	Name of Event
Button	onClick, onBlur, onFocus
Checkbox	onClick, onBlur, onFocus.
Radio	onClick, onBlur, onFocus
Reset	onReset
Submit	onSubmit
Text	onClick, onBlur, onFocus , onChange
Textarea	onClick, onBlur, onFocus , onChange
Select	onFocus, onBlur, onChange

Program 7: onblur event

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Submit Form  </title>
        <script language="javasript" type="text/javascript">


        function display()
                {
                   alert("Lost Focus");
                }


        </script>
  </head>
        <body>
                <form>
                        Enter your Name:
                        <input type="text" name="t1" onblur="display()"> <br>
                        Enter your Age:
                        <input type="text" name="t2"> <br>
                </form>
  </body>
</html>
```

Program 8: onfocus event

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Submit Form  </title>
        <script language="javasript" type="text/javascript">


        function display(a)
                {
                    a.style.background = "lightgreen";
                }


        </script>
  </head>
        <body>
                <form>
                        Enter your Name:
                        <input type="text" name="t1" > <br>
                        Enter your Age:
                        <input type="text" name="t2" onfocus="display(this)" > <br>
                </form>
  </body>
</html>
```

Program 9: onchange event

```
<!DOCTYPE html>
         <html>
  <head>
   <title> Submit Form  </title>
         <script language="javasript" type="text/javascript">

         function display()
                  {
                     alert("Selection changed");
                  }
         </script>
  </head>
         <body>
                  <form>
              Select Colours: <br>
                           <select name="s1" onchange="display()">
                           <option value="white">White   </option>
                           <option value="red"> Red       </option>
                           <option value="yellow"> Yellow </option>
                           </select>
                  </form>
  </body>
</html>
```

Program 10: onsubmit event

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Submit Form  </title>
        <script language="javasript" type="text/javascript">

        function display()
                {
                    alert("Form Submitted");
                }
        </script>
  </head>
        <body>
                <form onsubmit="display()">
                        Enter your Name:
                        <input type="text" name="t1" size="15"> <br>

                        <input type="submit" value="Submit"> <br>
                </form>
  </body>
</html>
```

Program 11: onreset event

```
<!DOCTYPE html>
        <html>
  <head>
   <title> Submit Form  </title>
        <script language="javasript" type="text/javascript">

        function display()
                {
                   alert("Form Reset");
                }
        </script>
  </head>

        <body>
                <form onreset="display()">
                        Enter your Name:
                        <input type="text" name="t1" size="15"> <br>

                        <input type="reset" value="Reset"> <br>
                </form>
  </body>
</html>
```

3.2.1 Mouse Event

Q. Explain various mouse events with example

The object mouse has numerous events associated with it which depend on the user's actions.

There following other 7 events which are generated by mouse when it comes in contact of any HTML tag.

Event Name	Description
onclick	Javascript runs when a mouse click
ondblclick	Javascript runs when a mouse double-click
onmousedown	Javascript runs when mouse button is pressed
onmousemove	Javascript runs when mouse pointer moves
onmouseout	Javascript runs when mouse pointer moves out of an element
onmouseover	Javascript runs when mouse pointer moves over an element
onmouseup	Javascript runs when mouse button is released

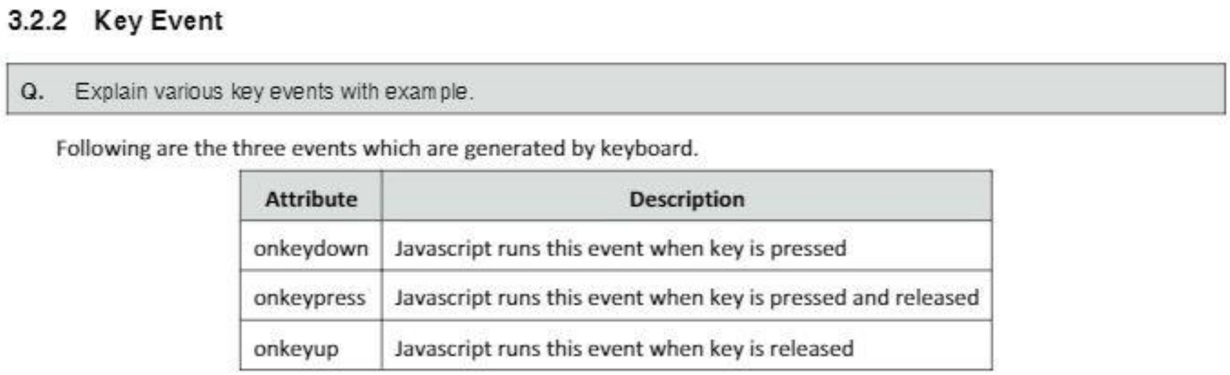
Program 12: onclick()

```
<!DOCTYPE html>
<html>
<head>
        <script language="javasript" type="text/javascript">

                function display()
                {
                    alert("Button Clicked");
                }
        </script>

</head>
<body>
  <form action="" method="post" name="entry">

                Country:<input type="text" name="cname"><br>
                <input type="button" name="b1" value="Click" onclick="display()">
  </form>
</body>
</html>
```

Program 13: onmouseover(), onmouseout()

```
<!DOCTYPE html>
<html>
<head>
        <script language="javasript" type="text/javascript">
                function over()
                {
                   document.entry.b1.value="Mouse over";
                }
                        function out()
                {
                   document.entry.b1.value="Mouse Out";
                }
        </script>
</head>
<body>
  <form action="" method="post" name="entry">


                Country:<input type="text" name="cname"><br>
                <input type="button" name="b1" value="Click" onmouseover="over()"
onmouseout="out()">
  </form>
</body>
</html>
```

}



3.2.2 Key Event

Q. Explain various key events with example.

Following are the three events which are generated by keyboard.

Attribute	Description
onkeydown	Javascript runs this event when key is pressed
onkeypress	Javascript runs this event when key is pressed and released
onkeyup	Javascript runs this event when key is released

Program 14:

```
<!DOCTYPE html>

<html>

<head>

        <script language="javasript" type="text/javascript">

                function down()

                {

                   document.entry.b1.value="Key Down";

                }

                function up()

                {

                   document.entry.b1.value="Key Up";

                }

                function press()

                {

                   document.entry.b2.value="Key Pressed";

                }
        </script>


</head>

<body>

  <form action="" method="post" name="entry">


                <input type="button" name="b1" value="Click" onkeydown="down()" onkeyup="up()"> <br><br>


                Enter Key:<input type="text" name="cname" onkeypress="press()"><br>

                        <input type="button" name="b2" value="Key">


  </form>

</body>

</html>
```

3.3 Form Objects and Elements

Q. Explain form objects and elements.

- Webpage is a collection of various elements including **window** as a first element.
- A window contains an HTML document which is known as **document** object.
- The document object has various properties like document.write() which allow access to and modification of document content. Documents can have more than one form and form can have multiple elements.
- The objects are represented in a hierarchical order :
  - **Window object** – this object is at the top of the hierarchy. It is the outmost element of the object hierarchy.
  - **Document object** – Every HTML document which loads into a window is a document object and it contains the elements of the page.
  - **Form object** – all tags those enclosed in the <form>...</form> tags set the form object.
  - **Form control elements** – form object includes all the elements like text fields, buttons, radio buttons, and checkboxes etc. defined for that object
- All the form objects are stored in array known as **forms** and it keeps the order in which they appeared in the document.

Program 15:

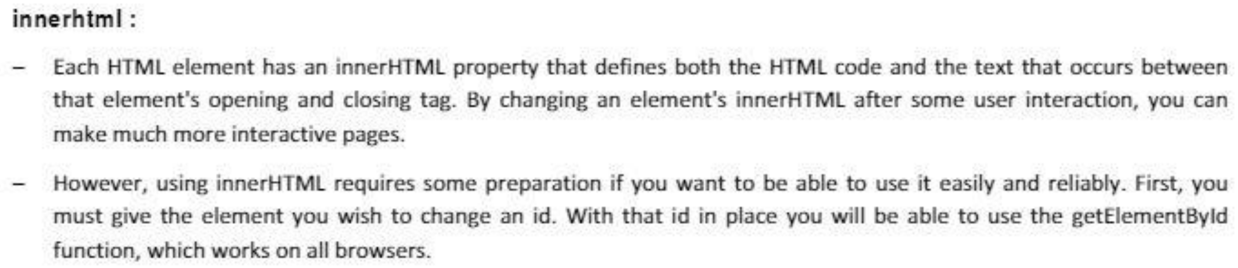
```
<!DOCTYPE html>

<html>

<head>

        <script language="javasript" type="text/javascript">


                function display()

                {

                   alert("Value="+document.forms.entry.cname.value);

                }


        </script>

</head>

<body>

  <form action="" method="post" name="entry">


                Country:<input type="text" name="cname"><br>

                <input type="button" name="b1" value="Print" onclick="display()">

  </form>

</body>

</html>
```

Program 16:

```
<!DOCTYPE html>
<html>
<head>
      <script language="javasript" type="text/javascript">
            function display()
            {
               with(document.forms.entry)
               {
               alert("Value="+elements[0].value+elements[1].value+elements[2].value);
               }
            }
      </script>
</head>
<body>
  <form action="" method="post" name="entry">

            Country:<input type="text" name="cname1"><br>
            State:<input type="text" name="cname2"><br>
            City:<input type="text" name="cname3"><br>
            <input type="button" name="b1" value="Print" onclick="display()">
  </form>
</body>
</html>
```

Program 17:

```
<!DOCTYPE html>

<html>

<head>

        <script language="javasript" type="text/javascript">


                function display()

                {

                   var i= document.getElementById("cname");

                   alert("You Entered="+i.value);


                }


        </script>

</head>

<body>

  <form action="" method="post" name="entry">


                Country:<input type="text" id="cname"><br>


                <input type="button" name="b1" value="Click" onclick="display()">

  </form>

</body>

</html>
```



innerhtml :

- Each HTML element has an innerHTML property that defines both the HTML code and the text that occurs between that element's opening and closing tag. By changing an element's innerHTML after some user interaction, you can make much more interactive pages.
- However, using innerHTML requires some preparation if you want to be able to use it easily and reliably. First, you must give the element you wish to change an id. With that id in place you will be able to use the getElementById function, which works on all browsers.

Program 18:

```
<!DOCTYPE html>

<html>

<head>

        <script language="javasript" type="text/javascript">


                function display()

                {

                   document.getElementById("myText").innerHTML="Good Day";

                }

        </script>

</head>

<body>

  <h4 id="myText">Welcome</h4>

 <form action="" method="post" name="entry">

  <input type="button" name="b1" value="Click" onclick="display()">

 </form>

</body>

</html>
```

Changing Attribute Value Dynamically: In javascript, we can change the attribute value of any form elements dynamically.

Program 19:

```
<!DOCTYPE html>
<html>
<head>
        <script language="javasript" type="text/javascript">
                function change(Element)
                {
                    Element.style.backgroundColour="blue";
                }
        </script>
</head>
<body>
  <form action="" method="post" name="entry">


                Country:<input type="text" value="India" name="cname1"
onchange="change(this)"/><br>
                City:<input type="text" value="Pune" name="cname2"
onchange="change(this)"/><br>


                <input type="button" name="b1" value="Click">
  </form>
</body>
</html>
```

Changing Option List Dynamically:

Program 20:

```
<!DOCTYPE html>
      <html>
 <head>
  <title> Select Demo  </title>
      <script language="javasript" type="text/javascript">

            function display(ElementValue)
            {
               with(document.forms.frm1)
               {
                if(ElementValue==1)
                {
                  op1[0].text="WPD";
                  op1[0].value=1;
                  op1[1].text="PIC";
                  op1[1].value=2;
                  op1[2].text="Maths";
                  op1[2].value=3;
                }

                if(ElementValue==2)
                {
```

```
             op1[0].text="OOP";
             op1[0].value=1;
             op1[1].text="DSU";
             op1[1].value=2;
             op1[2].text="DBMS";
             op1[2].value=3;
            }
           }

         }
    </script>

</head>

    <body>
         <form name="frm1">

              <select name="op1" size="3">
              <option value=1>WPD   </option>
              <option value=2>PIC       </option>
              <option value=3>Maths </option>
              </select> <br>
```

```
                <input type="radio" name="r1" value=1 checked="true"
onclick="display(this.value)">First Year <br>

                        <input type="radio" name="r1" value=2
onclick="display(this.value)">Second Year



                </form>

          </body>

</html>
```

Evaluating Checkbox Selection:

Program 21:

```
<!DOCTYPE html>
          <html>
  <head>
   <title> Checkbox Demo  </title>
          <script language="javasript" type="text/javascript">

                    function display()
                    {
                        with(document.forms.frm1)
                        {

                         if(c1.checked==true)
                         {
                         alert("C");
                         }

                         if(o1.checked==true)
                         {
                         alert("OOP");
                         }

                         if(j1.checked==true)
                        {
                        alert("Java");
                        }


                         if(p1.checked==true)
                        {
                        alert("PHP");
                        }
                       }
                    }
            </script>
    </head>
            <body>
                    <form name="frm1">
                    <input type="checkbox" name="c1" value="c">C <br>
                    <input type="checkbox" name="o1" value="oop">OOP <br>
                    <input type="checkbox" name="j1" value="java">JAVA <br>
                    <input type="checkbox" name="p1" value="php">PHP <br>


        <input type="reset" name="r1" value="Show" onclick="display()"> <br>
                    </form>
              </body>
</html>
```

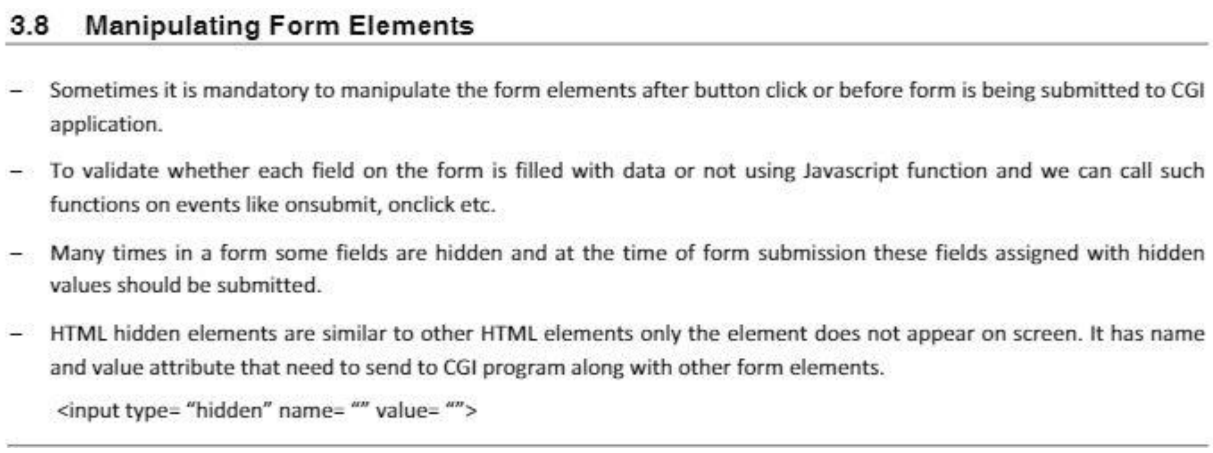
Changing a Label Dynamically:

Program 22:

```
<!DOCTYPE html>
          <html>
  <head>
   <title> Label Demo  </title>
          <script language="javasript" type="text/javascript">

                    function display(ElementValue)
                    {
                        with(document.forms.frm1)
                        {
                         if(ElementValue=="FY")
                         {
                            b1.value="SY";
                            op1[0].text="WPD";
                            op1[0].value=1;
                            op1[1].text="PIC";
                            op1[1].value=2;
                            op1[2].text="Maths";
                            op1[2].value=3;
                         }
```

```
               if(ElementValue=="SY")
               {
                 b1.value="FY";
                 op1[0].text="OOP";
                 op1[0].value=1;
                 op1[1].text="DSU";
                 op1[1].value=2;
                 op1[2].text="DBMS";
                 op1[2].value=3;
               }
              }

            }
     </script>

</head>

     <body>
            <form name="frm1">

                  <select name="op1" size="3">
                  <option value=1>WPD   </option>
                  <option value=2>PIC      </option>
                  <option value=3>Maths </option>
```

```
                </select> <br> <br>

            <input type="reset" name="b1" value="SY" onclick="display(this.value)"> <br>

            </form>
        </body>
</html>
```



3.8 Manipulating Form Elements

- Sometimes it is mandatory to manipulate the form elements after button click or before form is being submitted to CGI application.
- To validate whether each field on the form is filled with data or not using Javascript function and we can call such functions on events like onsubmit, onclick etc.
- Many times in a form some fields are hidden and at the time of form submission these fields assigned with hidden values should be submitted.
- HTML hidden elements are similar to other HTML elements only the element does not appear on screen. It has name and value attribute that need to send to CGI program along with other form elements.

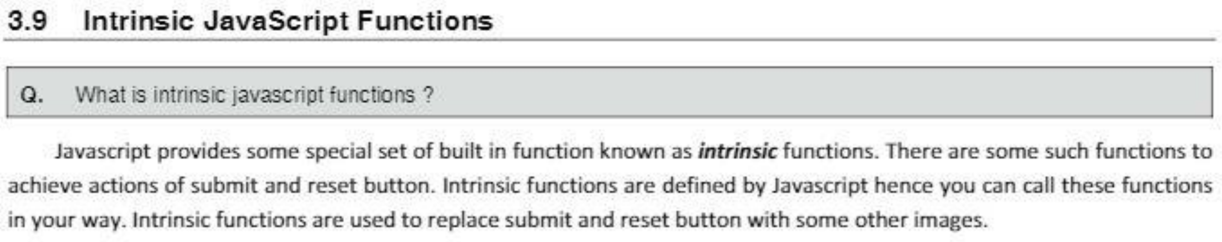
  <input type= “hidden” name= “” value= “”>

Program 23:

```
<!DOCTYPE html>
          <html>
  <head>
   <title> Form Manipulation Demo  </title>
          <script language="javasript" type="text/javascript">
                    function display()
                    {
                        with(document.forms.frm1)
                        {
                         if(year.value.length>0 && branch.value.length>0)
                         {
                            course.value=year.value+branch.value;
                            alert(course.value);
                         }
                        }
                    }
```

```
        </script>
  </head>

        <body>
                <form name="frm1">
                Year: <input type="text" name="year"/> <br>
                Branch:<input type="text" name="branch"/> <br>
                Course:<input type="hidden" name="course"/> <br><br>

                <input type="submit" value="Submit" onclick="display()"> <br>
                </form>
          </body>
</html>
```

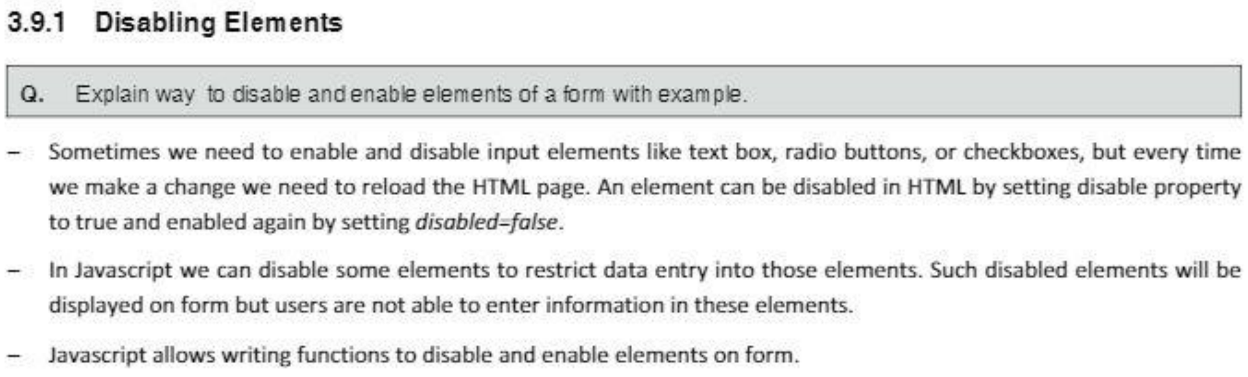
Intrinsic functions:



3.9 Intrinsic JavaScript Functions

Q. What is intrinsic javascript functions ?

Javascript provides some special set of built in function known as ***intrinsic*** functions. There are some such functions to achieve actions of submit and reset button. Intrinsic functions are defined by Javascript hence you can call these functions in your way. Intrinsic functions are used to replace submit and reset button with some other images.



3.9.1 Disabling Elements

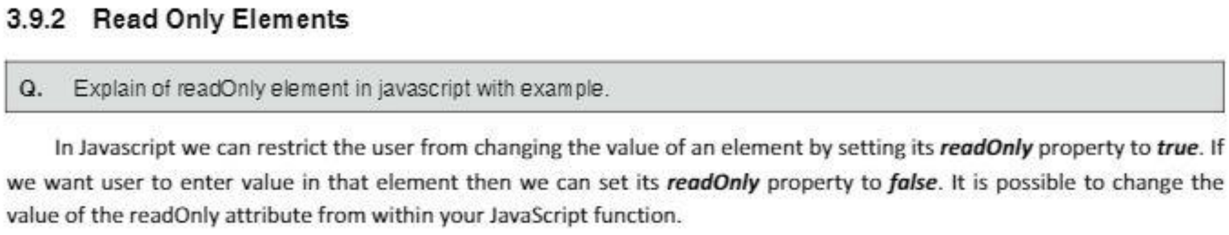
Q. Explain way to disable and enable elements of a form with example.

- Sometimes we need to enable and disable input elements like text box, radio buttons, or checkboxes, but every time we make a change we need to reload the HTML page. An element can be disabled in HTML by setting disable property to true and enabled again by setting *disabled=false*.
- In Javascript we can disable some elements to restrict data entry into those elements. Such disabled elements will be displayed on form but users are not able to enter information in these elements.
- Javascript allows writing functions to disable and enable elements on form.

Program 24:

```
<!DOCTYPE html>

        <html>

  <head>

   <title> Disable Element Demo  </title>

        <script language="javasript" type="text/javascript">



                function disable()

                {

                   document.forms.frm1.t1.disabled=true;

                }
```

```
                        function enable()

                {

                    document.forms.frm1.t1.disabled=false;

                }


        </script>


  </head>


        <body>

                <form name="frm1">




                Enter your Name: <input type="text" name="t1"/> <br>


                <input type="button" name="b1" value="Disable" onclick="disable()">

                        <input type="button" name="b2" value="Enable" onclick="enable()">


                </form>

          </body>

</html>
```



3.9.2 Read Only Elements

Q. Explain of readOnly element in javascript with example.

In Javascript we can restrict the user from changing the value of an element by setting its ***readOnly*** property to ***true***. If we want user to enter value in that element then we can set its ***readOnly*** property to ***false***. It is possible to change the value of the readOnly attribute from within your JavaScript function.

Program 25:

```
<!DOCTYPE html>

         <html>

  <head>

   <title> ReadOnly Element Demo  </title>

         <script language="javasript" type="text/javascript">


                  function readonly()

                  {

                      document.forms.frm1.t1.readOnly=true;

                  }


                  function enter()

                  {

                      document.forms.frm1.t1.readOnly=false;

                  }


         </script>
```

```
 </head>

    <body>
        <form name="frm1">



        Enter your Name: <input type="text" name="t1"/> <br>

        <input type="button" name="b1" value="ReadOnly" onclick="readonly()">
            <input type="button" name="b2" value="Write" onclick="enter()">

        </form>
     </body>
</html>
```

Intrinsic functions Programs:

Program 26:

```
<!DOCTYPE html>

<html>

<head>

<title>Intrinsic Functions</title>

<script language="javasript" type="text/javascript">


</script>

</head>


<body>

<form action=""  name="entry">


Enter Name:<input type="text" name="t1" > <br>

Enter Age:<input type="text" name="t2" > <br><br>


<img src="Submit.jpg" height="100" width="150"


onclick="javascript:document.forms.entry.submit()">


</form>

</body>

</html>
```

Program 27:

```
<!DOCTYPE html>
<html>
<head>
<title>Intrinsic Functions</title>
<script language="javasript" type="text/javascript">


</script>
</head>


<body>
<form action=""  name="entry">


Enter Name:<input type="text" name="t1" > <br>
Enter Age:<input type="text" name="t2" > <br><br>


<img src="Submit.jpg" height="100" width="150"

onclick="javascript:document.forms.entry.submit()">
<img src="Reset.jpg" height="100" width="150"

onclick="javascript:document.forms.entry.reset()">
</form>
</body>
</html>
```