Ex. No. :1 a **Date :**

Reg. No. :

HTML – Element - Tables

Aim:

Program to design a class timetable using HTML basic elements.

Procedure:

1. Tables are defined with the **table** element
2. Use the ***border*** attribute specifies the table’s border width in pixels. To create a table without a border, set **border** to **"0"**.
3. Use the ***tr*** element to define an individual *table row*.
4. The columns in the head section are defined with **th** elements.
5. *Data cells* contain individual pieces of data and are defined with **td** (*table data*) elements within each row.
6. Table cells are sized to fit the data they contain. Document authors can create larger data cells by using attributes ***rowspan*** and ***colspan***. The values assigned to these attributes specify the number of rows or columns occupied by a cell.
7. Use the attribute **rowspan = "2"** to allow the cell to use two vertically adjacent cells (thus the cell *spans* two rows).
8. Use the attribute **colspan = "4"** to widen the header cell to span four cells.

Ex. No. :1 b **Date :**

Roll No. : **Reg. No. :**

HTML – Form Elements

Aim:

Program to design a email registration form using HTML Form elements.

Procedure:

1. The form is defined by a ***form*** element.
   ```
   <form method = "post" action = "/cgi-bin/formmail">
   ```
2. Use thea attribute ***method*** specifies how the form's data is sent to the Web server.Using ***method = "post"*** appends form data to the browser request, which contains the protocol (i.e., HTTP) and the requested resource's URL. Scripts located on the Web server's computer (or on a computer accessible through the network) can access the form data sent as part of the request. For example, a script may take the form information and update an electronic mailing list. The other possible value, ***method = "get"*** appends the form data directly to the end of the URL.
3. The ***action*** attribute in the **<form>** tag specifies the URL of a script on the Web server; in this case, it specifies a script that e-mails form data to an address. Most Internet Service Providers (ISPs) have a script like this on their site; ask the Web site system administrator how to set up an XHTML document to use the script correctly.
4. Use the **type** of **input** as ***"text"*** **input** inserts a *text box* into the form. Users can type data in text boxes.
5. The **input** element's ***size*** attribute specifies the number of characters visible in the text box. Optional attribute ***maxlength*** limits the number of characters input into the text box.
6. There are two types of **input** elements in lines
   ```
   <input type = "submit" value = "Submit Your Entries" />
   <input type = "reset" value = "Clear Your Entries" />
   ```
7. The ***"submit"*** **input** element is a button. When the user presses a **"submit"** button, the browser sends the data in the form to the Web server for processing. The ***value*** *attribute* sets the text displayed on the button (the default value is **Submit Query**).
8. The ***"reset"*** **input** element allows a user to reset all **form** elements to their default values. The **value** attribute of the **"reset" input** element sets the text displayed on the button (the default value is **Reset**).
9. The ***textarea*** element inserts a multiline text box, called a *text area*, into the form. The number of rows is specified with the ***rows*** *attribute* and the number of columns (i.e., characters) is specified with the ***cols*** *attribute*. In this example, the **textarea** is four rows high and 36 characters wide. To display default text in the text area, place the text between the **<textarea>** and **</textarea>** tags. Default text can be specified in other **input** types, such as text boxes, by using the **value** attribute.
10. The ***"password"*** input in lines inserts a password box with the specified **size**. A password box allows users to enter sensitive information, such as credit card numbers and passwords, by "masking" the information input with asterisks. The actual value input is sent to the Web server, not the character that mask the input.

11. Checkboxes enable users to select from a set of options. When a user selects a checkbox, acheck mark appears in the check box. Otherwise, the checkbox remains empty. Each ***"checkbox"*** **input** creates a new checkbox. Checkboxes can be used individually or in groups. Checkboxes that belong to a group are assigned the same **name**.
12. Radio buttons are similar to checkboxes, except that only one radio button in a group of radio buttons may be selected at any time. The radio buttons in a group have the same **name** attributes and are distinguished by their different **value** attributes. The attribute-value pair ***checked* = *"checked"*** indicates which radio button, if any, is selected initially. The **checked** attribute also applies to checkboxes.
13. The ***select*** element provides a drop-down list of items from which the user can select an item. The **name** attribute identifies the drop-down list. The ***option*** element adds items to the drop-down list. The **option** element's ***selected*** *attribute* specifies which item initiallyis displayed as the selected item in the **select** element.