**Discuss TCP/IP Protocol**

The TCP/IP (Transmission Control Protocol/Internet Protocol) suite is a set of communication protocols that form the foundation for the Internet and most local networks. Developed in the 1970s and 1980s, TCP/IP has become the standard protocol suite for networking, and it provides a reliable and efficient means of communication between devices across diverse networks. Here's an overview of the key components of the TCP/IP protocol suite:

TCP (Transmission Control Protocol):

Functionality: TCP provides reliable, connection-oriented communication between devices. It ensures that data is delivered in the correct order and without errors. To achieve this, TCP uses a combination of sequence numbers, acknowledgments, and retransmission mechanisms.

Connection Establishment and Termination: TCP follows a three-way handshake process to establish a connection and a four-way handshake to terminate it.

IP (Internet Protocol):

Functionality: IP is responsible for the addressing and routing of packets across a network. It provides an addressing scheme to uniquely identify devices in a network and enables the routers to forward packets from the source to the destination.

Versions: There are two main versions of IP in use today: IPv4 (Internet Protocol version 4) and IPv6 (Internet Protocol version 6). IPv4 is the older version and uses 32-bit addresses, while IPv6 uses 128-bit addresses and was introduced to address the exhaustion of IPv4 addresses.

**What is a form ? Explain form components with example.**

In the context of web development and user interfaces, a "form" refers to an HTML (Hypertext Markup Language) element that allows users to input data and submit it to a server for processing. Forms are a crucial part of web applications and websites, providing a means for users to interact with and input information.

A basic form structure in HTML looks like this:

```html

<form action="/submit" method="post">

<!-- Form components go here -->

<label for="username">Username:</label>

<input type="text" id="username" name="username" required>

<label for="password">Password:</label>

<input type="password" id="password" name="password" required>

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

</form>

```

Let's break down the components of this form:

1. \*\*`<form>` element:\*\* This is the container for all the form components. It has attributes like `action` (specifying where the form data should be sent) and `method` (specifying the HTTP method to be used for submitting the form, commonly "get" or "post").

2. \*\*`<label>` element:\*\* This is used to associate a text label with a form control, enhancing accessibility and user experience. The `for` attribute in the `<label>` tag should match the `id` attribute of the associated form control.

3. \*\*`<input>` element:\*\* This is one of the most versatile form elements. It is used for various types of user input, such as text, password, checkbox, radio button, etc. The `type` attribute determines the kind of input field, and the `id` and `name` attributes help identify and associate the input with labels.

- \*\*Text Input Example:\*\*

```html

<label for="email">Email:</label>

<input type="text" id="email" name="email" required>

```

- \*\*Password Input Example:\*\*

```html

<label for="password">Password:</label>

<input type="password" id="password" name="password" required>

```

4. \*\*`<textarea>` element:\*\* This is used for multiline text input. It's useful when users need to enter longer passages of text.

```html

<label for="message">Message:</label>

<textarea id="message" name="message" rows="4" cols="50"></textarea>

```

5. \*\*`<select>` and `<option>` elements:\*\* These are used for dropdown menus. Users can select an option from the list.

```html

<label for="cars">Choose a car:</label>

<select id="cars" name="cars">

<option value="volvo">Volvo</option>

<option value="saab">Saab</option>

<option value="mercedes">Mercedes</option>

<option value="audi">Audi</option>

</select>

```

6. \*\*`<button>` element:\*\* This is used to create a clickable button within the form. It can be used to trigger form submission or perform other actions.

```html

<button type="submit">Submit</button>

```

These are some of the basic form components in HTML. Depending on the requirements, you can use additional attributes and elements to enhance the functionality and appearance of forms.

**Explain rowspan and colspan with an example.**

**What is a form ? What are the major attributes of the form ? Explain any six form components with example**

**Write a HTML code to generate following output**

**a) Write a HTML code to generate following output**

**Maharashtra**

**o Pune**

**I. Dighi**

**II. Moshi**

**III. Shivajinagar**

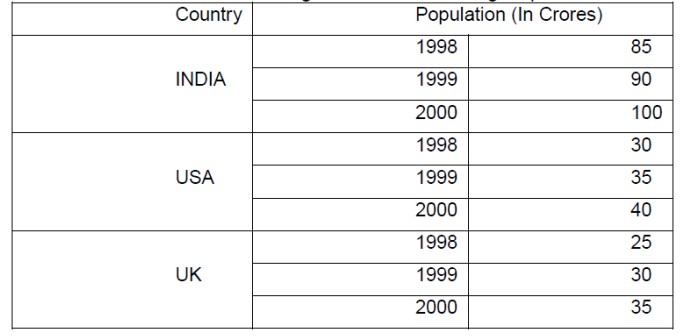
**o Mumbai**

**I. Santakruiz**

**II. Vikroli**

**III. Mumbra**

**Write HTML code to generate the following output**

****

**Explain the structure of the HTML webpage with an example.**

**Write a Javascript code which checks the contents entered in a forms text element. If the text entered is in the lower cases convert to upper case.**

**Explain various datatypes used in Javascript. 5 b) Write a Javascript to find factorial of a number.**

**Explain pattern matching. What is the different meta characters used in pattern matching ?**

**Create a form for student information. Write Java Script code to find total, average, result and grade.**

**Write a short note on DOM.**

**What is an array object ? How to create an array object ? Explain any six array methods.**

**Explain GET and POST request methods.**

**Design A JavaScript to display whether given number is prime or not.**

**. Explain briefly about Built in Java script Objects?**

Unit III

**Explain the CSS selectors .**

Answer on this link https://www.tutorialspoint.com/jquery/jquery-selectors.htm

1 Element Selectors

Element selectors select all instances of a given HTML element. You can select all

elements by using the universal element selector, which is the \* (asterisk) character.

You can select a group of elements by separating the different element names with

commas. This is a sensible way to reduce the size and complexity of your CSS files,

by combining multiple identical rules into a single rule.

2 Class Selectors

A class selector allows you to simultaneously target different HTML elements

regardless of their position in the document tree. If a series of HTML elements have

been labeled with the same class attribute value, then you can target them for styling

by using a class selector, which takes the form: period (.) followed by the class name.

3 Id Selectors

An id selector allows you to target a specific element by its id attribute regardless of

its type or position. If an HTML element has been labeled with an id attribute, then

you can target it for styling by using an id selector, which takes the form: pound/hash

(#) followed by the id name.

4 Attribute Selectors

An attribute selector provides a way to select HTML elements either by the

presence of an element attribute or by the value of an attribute. This can be a very

powerful technique, but because of uneven support by some of the browsers, not all

web authors have used them. Attribute selectors can be a very helpful technique in

the styling of hyperlinks and images. For instance, perhaps we want to make it more

obvious to the user when a pop-up tooltip is available for a link or image. We can do

this by using the following attribute selector: [title] { … } This will match any

element in the document that has a title attribute.

5 Pseudo-Element and Pseudo-Class Selectors

A pseudo-element selector is a way to select something that does not exist explicitly

as an element in the HTML document tree but which is still a recognizable selectable

object. For instance, you can select the first line or first letter of any HTML element

using a pseudo-element selector. A pseudo-class selector does apply to an

HTML element, but targets either a particular state or, in CSS3, a variety of family

relationships. The most common use of this type of selectors is for targeting link

states. By default, the browser displays link text blue and visited text links purple.

Do be aware that this state does not occur on touch screen devices. Note the syntax of

pseudo-class selectors: the colon (:) followed by the pseudo-class selector name. Do

be aware that a space is not allowed after the colon. Believe it or not, the order of

these pseudo-class elements is important. The :link and :visited pseudo-classes should

appear before the others. Some developers use a mnemonic to help them remember

the order. My favorite is “Lord Vader, Former Handle Anakin” for Link, Visited,

Focus, Hover, Active.

6 Contextual Selectors

A contextual selector (in CSS3 also called combinators) allows you to select

elements based on their ancestors, descendants, or siblings. That is, it selects

elements based on their context or their relation to other elements in the document

tree. While some of these contextual selectors are used relatively infrequently, almost

all

web authors find themselves using descendant selectors.A descendant selector

matches all elements that are contained within another element. The character used to

indicate descendant selection is the space character.

Selector Matches Example

Descendant A specified element that is contained somewhere within another

specified element. div p Selects a <p> element that is contained somewhere within a

<div> element. That is, the <p> can be any descendant, not just a child. Child A

specified element that is a direct child of the specified element.

div>h2

Selects an <h2> element that is a child of a <div> element.

Adjacent sibling A specified element that is the next sibling (i.e., comes directly

after) of the specified element.

h3+p

Selects the first <p> after any <h3>. General sibling A specified element that shares

the same parent as the specified element.

h3~p

Selects all the <p> elements that share the same parent as the <h3>

What is JQuery? Explain JQuery Selectors?

What is a box model? And what are the different elements of a box model?

Difference between JavaScript and JQuery

Answer to the above question

Both of these are useful in programming, but there is a primary difference between JavaScript and JQuery. JavaScript, on the one hand, is a scripting language used in programming. It is dynamic and helps in web development. But it is more complex since the programmer has to write the complete JavaScript code in a program. JQuery, on the other hand, is a JavaScript library. It is comparatively less complex- since a programmer only has to write the important JQuery code.

In this article, we will dive deeper into the difference between JavaScript and JQuery. Read ahead to know more.

**What is JavaScript?**

It is a scripting language used by programmers to make their websites more interactive and responsive. Along with CSS and HTML, JavaScript (commonly abbreviated as JS) is one of the most important resources used for creating web pages. CSS and HTML are used to design and decorate web pages. But JS helps us create a more dynamic web page. We can say that it breathes life into the pages.

JS is a major client-side language, but it isn’t confined to web development alone. We can also use JS in various server and desktop programs (the most common example here is Node.JS). A few databases also make use of JS, such as CouchDB and MongoDB.

Whenever any browser parses the web pages, then its primary responsibility is the creation of a tree-structure presentation in the memory.

**What is JQuery?**

It is a JS framework that is developed from JavaScript. JQuery is a very popular JS library. It is free, fast, rich-featured, concise, open-source, and has cross-browser compatibility. John Resign invented it, and it was released at BarCamp NYC back in January 2006.

The reason why JQuery was developed was to make the programmers’ lives easier. It acts as a feature-rich library that provides better client-side services. This way, JavaScript can be used effectively for creating browser-based apps and websites.

Here are a few features that JQuery offers:

* Manipulation of DOM: We can easily traverse and modify the DOM elements.
* Built-in features for animations in an app or a website.
* Event handling in HTML and manipulation.
* CSS manipulation.
* It is only 19kb in size. Very lightweight to operate on and work with.
* It consists of a very high-level widget library for UI.

**Difference between JavaScript and JQuery**

Let us talk about the differences between JavaScript and JQuery.

|  |  |  |
| --- | --- | --- |
| **Parameters** | **JavaScript** | **JQuery** |
| Basics | JS is a programming language. It is a scripting language that is dynamic and helps in web development. | It is a JS library. |
| Complexity | It is more complex since the programmer has to write the complete JavaScript code in a program. | It is comparatively less complex. A programmer only has to write the important JQuery code. |
| Consumption of Time | Since one has to write the whole script, it consumes much more time. | It consumes comparatively much less time since it makes programming easy and fast. |
| Multi-Browser Compatibility | To handle multi-browser compatibility, a developer has to develop code of their own. | One doesn’t need to focus on their app’s issue with compatibility on multiple browsers. This feature comes as a prerequisite. |
| Browser Support | Every browser supports JavaScript. We don’t need to include any additional plugins to make JS work. | We have to include the JQuery library’s URL in the page’s header so that JQuery works. |
| Inter-Dependency | JQuery forms a part of JS. The JS code does not necessarily depend on JQuery. | JQuery always depends on JS- since it is a JavaScript library. |
| Lines of Code | The code can be long and complicated. It takes time to program using JavaScript. | It consists of very few lines of code; it is easy to operate and work with. |
| Uses | It is a very crucial programming language used for web designing and programming desktop programs and servers. | It optimises the working of the JavaScript language. We can make all apps and websites more interactive, fast, and efficient while still decreasing the complexity of development. |
| Type of Approach | It is a very weakly typed approach used in programming. | It is a fast, simple, and easy approach used in programming. |
| DOM | It is very slow for the creation of DOM. | It is capable of creating DOM much faster. |

Define AJAX. Explain working of Ajax

Answer to the above question in the following link

https://www.tutorialspoint.com/ajax/what\_is\_ajax.htm

Explain AJAX request

Discuss the Jquery Dom manipulation methods

Anwer on this link <https://www.tutorialspoint.com/jquery/jquery-dom.htm>

Explain Jquery Dom traversing

Answer to the above question in the link given below

https://www.tutorialspoint.com/jquery/jquery-traversing.htm

Unit 5

List and Explain different super global arrays