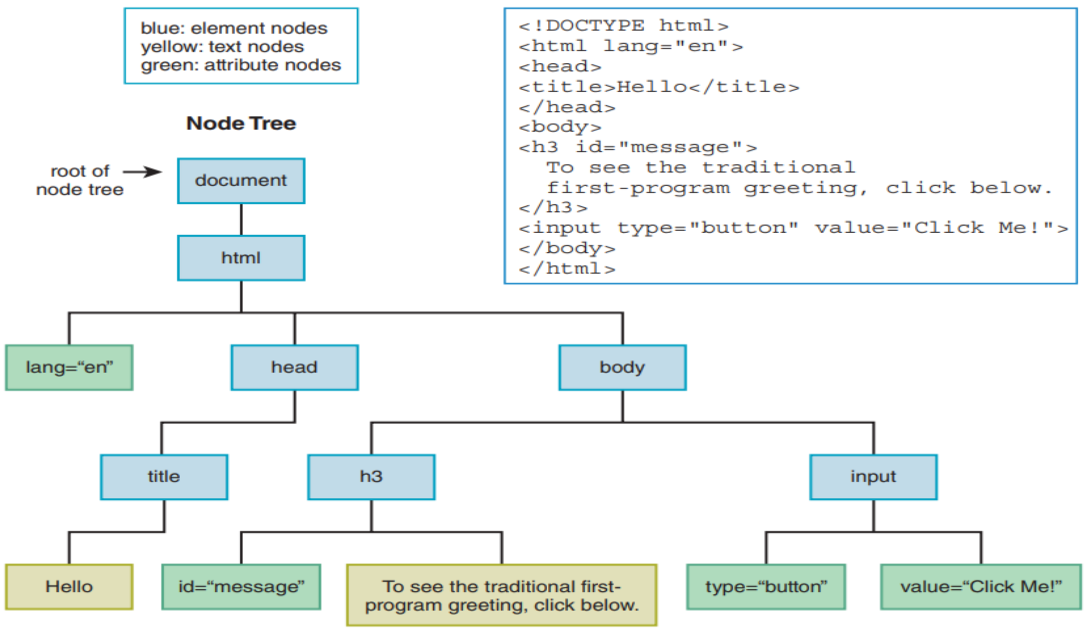
1. **What is Document Object Model (DOM) and what are the three ways to access the nodes in the node tree. Write a DOM node tree for any simple web page**

The Document Object Model, which is normally referred to as the DOM, models all of the parts of a web page document as nodes in a node tree. A node tree is similar to a directory tree, except instead of showing directories that include other directories (and files), it shows web page elements that include other elements (and text and attributes). Each node represents either (1) an element, (2) a text item that appears between an element's start and end tags, or (3) an attribute within one of the elements



```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Hello</title>
</head>
<body>
<h3 id="message">
  To see the traditional
  first-program greeting, click below.
</h3>
<input type="button" value="Click Me!">
</body>
</html>
```

FIGURE 8.3 Node tree for simplified Hello web page

The DOM provides different ways to access the nodes in the node tree. Here are three common techniques:

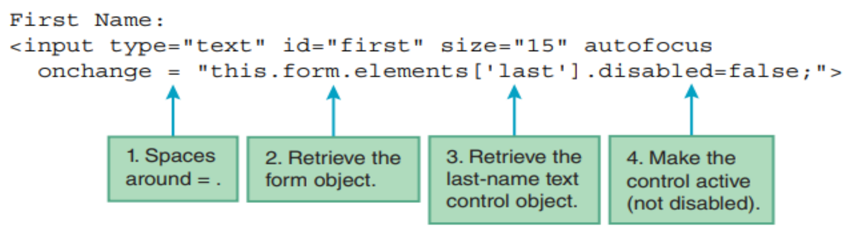
1. You can retrieve the node tree's root by using document (for the document object) in your code and then use the root object as a starting point in traversing down the tree.
2. You can retrieve the node that the user just interacted with (e.g. a button that was clicked) and use that node object as a starting point in traversing up or down the tree.
3. You can retrieve a particular element node by calling the document object's getElementById method with the element's id value as an argument.

2. List and explain any 5 event handler attributes and their associated events

Event-Handler Attributes	Events
onclick	User clicks on an element.
onfocus	An element gains focus.
onchange	The value of a form control has been changed.
onmouseover	Mouse moves over an element.
onmouseout	Mouse moves off an element.
onload	An element finishes loading.

Example for Onchange:

The next step involves adding an onchange event handler to the first-name text control's input element. Note the onchange event handler:



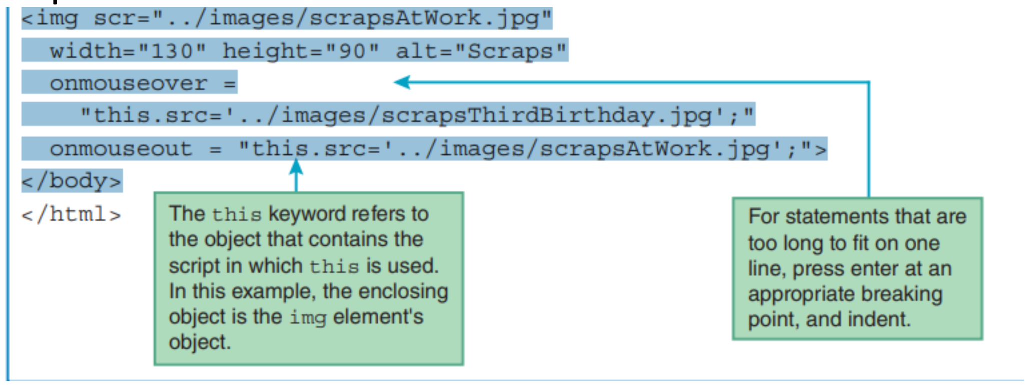
```
First Name:
<input type="text" id="first" size="15" autofocus
  onchange = "this.form.elements['last'].disabled=false;">
```

1. Spaces around = .
2. Retrieve the form object.
3. Retrieve the last-name text control object.
4. Make the control active (not disabled).

Example for OnClick

onclick="generateEmail(this.form)";

Example for Onmouseover and onmouseout

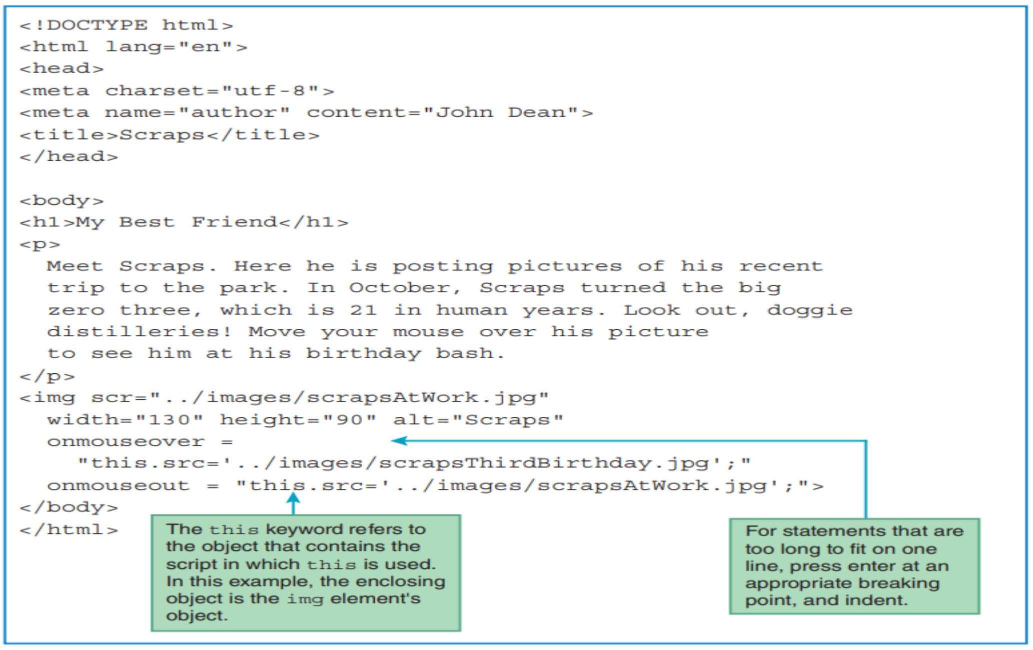


```
<img scr="../images/scrapsAtWork.jpg"
  width="130" height="90" alt="Scraps"
  onmouseover =
    "this.src='../images/scrapsThirdBirthday.jpg';"
  onmouseout = "this.src='../images/scrapsAtWork.jpg';">
</body>
</html>
```

The this keyword refers to the object that contains the script in which this is used. In this example, the enclosing object is the img element's object.

For statements that are too long to fit on one line, press enter at an appropriate breaking point, and indent.

3 Source code for Scraps the Dog web page



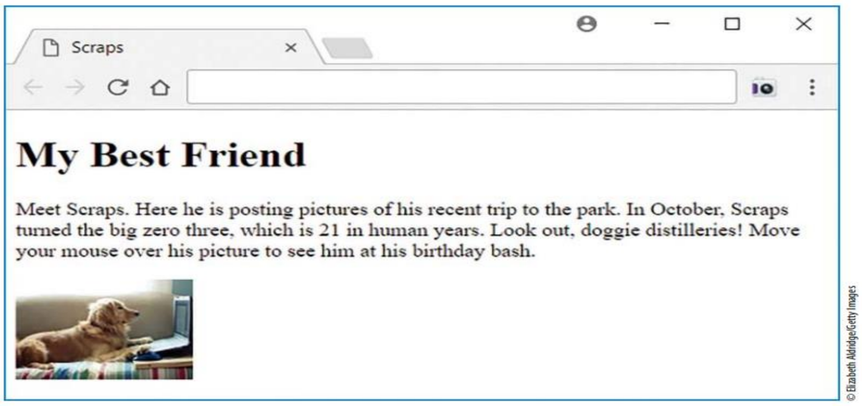
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="utf-8">
<meta name="author" content="John Dean">
<title>Scraps</title>
</head>

<body>
<h1>My Best Friend</h1>
<p>
  Meet Scraps. Here he is posting pictures of his recent
  trip to the park. In October, Scraps turned the big
  zero three, which is 21 in human years. Look out, doggie
  distilleries! Move your mouse over his picture
  to see him at his birthday bash.
</p>
<img scr="../images/scrapsAtWork.jpg"
  width="130" height="90" alt="Scraps"
  onmouseover =
    "this.src='../images/scrapsThirdBirthday.jpg';"
  onmouseout = "this.src='../images/scrapsAtWork.jpg';">
</body>
</html>
```

The this keyword refers to the object that contains the script in which this is used. In this example, the enclosing object is the img element's object.

For statements that are too long to fit on one line, press enter at an appropriate breaking point, and indent.

FIGURE 8.13 **Source code for Scraps the Dog web page**

Output:


Scraps

My Best Friend

Meet Scraps. Here he is posting pictures of his recent trip to the park. In October, Scraps turned the big zero three, which is 21 in human years. Look out, doggie distilleries! Move your mouse over his picture to see him at his birthday bash.

© Elizabeth Aldridge/Getty Images

FIGURE 8.12 **Scraps the Dog web page**

3. Explain the coding conventions used in On change Event Handler

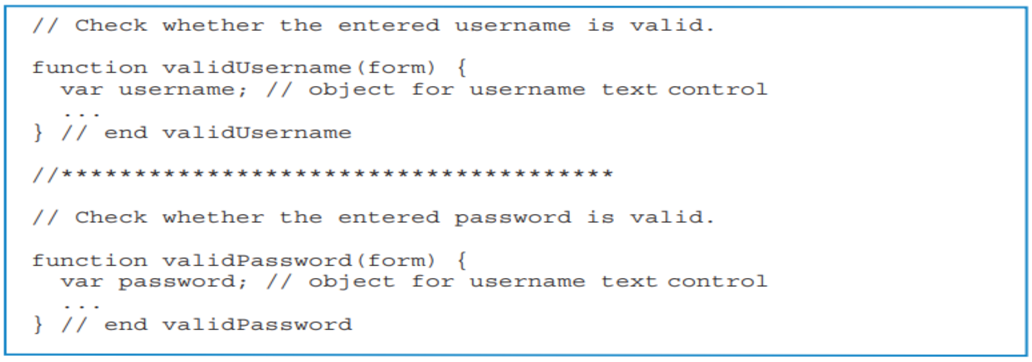
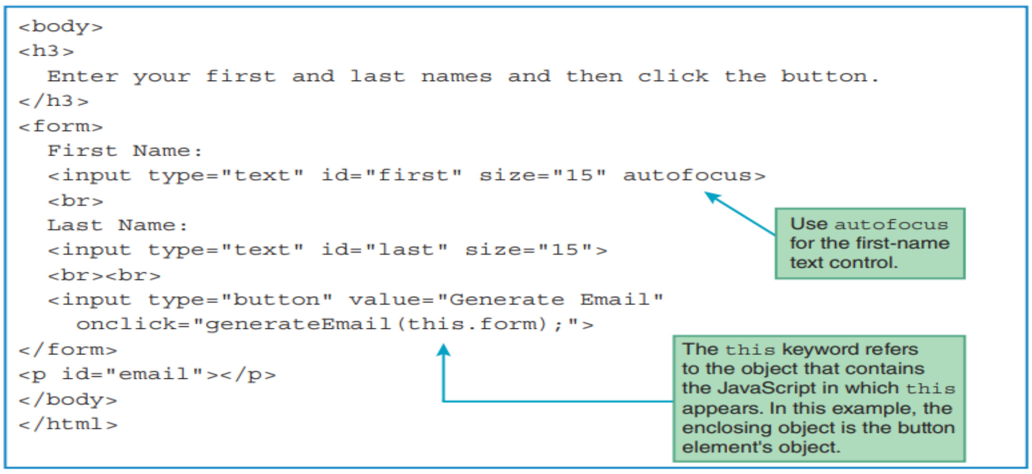

```
// Check whether the entered username is valid.

function validUsername(form) {
  var username; // object for username text control
  ...
} // end validUsername

//****************************************

// Check whether the entered password is valid.

function validPassword(form) {
  var password; // object for username text control
  ...
} // end validPassword
```

FIGURE 8.10 **Code skeleton that illustrates coding conventions**

code skeleton follows that convention:
‣ If there are two or more functions, separate each adjacent pair of functions with a line of *'s surrounded by blank lines.
‣ Put all variable declarations at the top of a function's body, and for each variable declaration, provide a comment that describes the variable.
‣ Provide an "end …" comment for each function's closing brace.
‣ Position a function's opening brace ({) at the right of the function heading, separated by a space.
‣ Position a function's closing brace (}) in the same column as the function heading's first character.
‣ Between a function's opening and closing braces, indent each statement with two spaces

4. Create a form to create an email id generator, accepting the first name, last name, age, profession and address. When the user clicks on the generate button, display the generated email

Note: accepting the first name, last name is provided in the form **add , age, profession and address. attributes**

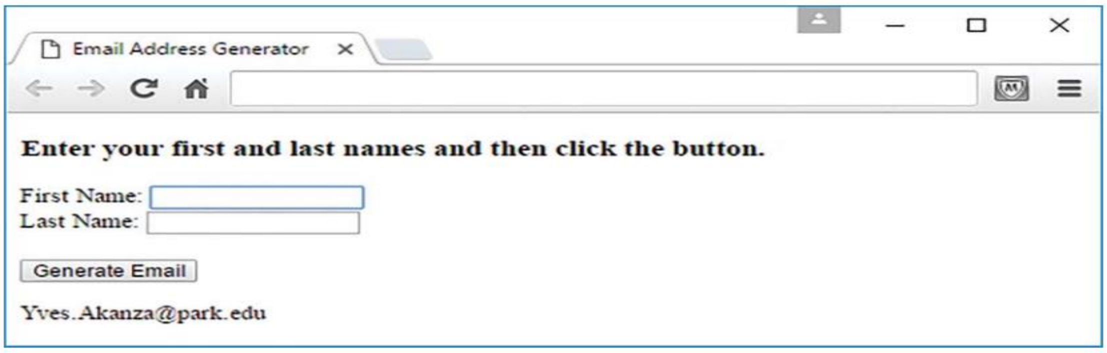


```
<body>
<h3>
  Enter your first and last names and then click the button.
</h3>
<form>
  First Name:
  <input type="text" id="first" size="15" autofocus>
  <br>
  Last Name:
  <input type="text" id="last" size="15">
  <br><br>
  <input type="button" value="Generate Email"
    onclick="generateEmail(this.form);">
</form>
<p id="email"></p>
</body>
</html>
```

Use autofocus for the first-name text control.
The this keyword refers to the object that contains the JavaScript in which this appears. In this example, the enclosing object is the button element's object.

FIGURE 8.9A body container for Email Address Generator web page

Output:



Enter your first and last names and then click the button.

First Name:
Last Name:

Generate Email

Yves.Akanza@park.edu

FIGURE 8.8 Email Address Generator web page—what happens after the user enters first and last names and what happens after the user clicks the Generate Email button

5.Create a web page to manage phone numbers of employees in a company. On clicking on the submit button display the contents on the web page.

Javascript:

```
<script type="text/javascript">
  function validateForm() {
    return checkPhone()      }

  function checkPhone() {
    var phone = document.forms["myForm"]["phone"].value;
    var phoneNum = /^\(?([0-9]{3})\)?[-. ]?([0-9]{3})[-. ]?([0-9]{4})$/;
      if(phone.value.match(phoneNum)) {
        return true;      }
      else
document.getElementById("phone").className = document.getElementById("phone").className + "
error";
        return false;   } }
               </script>
```

HTML:

```
<form name="myForm" onsubmit = "return validateForm()">
  Phone Number: <input type="text" id="phone"><br>
</form>
```

Discuss the JavaScript properties for the following text control element attributes with examples: type, placeholder, size, maxLength, value, autofocus, disabled, readOnly

To Create a HTML Input Text Box you need to dine **type="text"** attribute in **<input>** tag. A **<input>** tag is very important to give to the user to input filed to enter a content (data)

```
Example:<input type="text">
```

Attribute	**Description**
`maxlength`	Set the limit maximum number of characters the input accepted by input bo
`minlength`	Set the limit minimum number of characters long the input should be and still be considered valid
`pattern`	A validation of input using a regular expression for contents must match in order.
`placeholder`	When input text filed empty that will show as an exemplary value to display.
`readonly`	A stop to user input any data, its a Boolean attribute indicating whether input be read-only or not.
`size`	This number indicating how many characters wide the input field should be.
`spellcheck`	Enable spell checking for the input field.

maxlength & minlength

```
<!DOCTYPE html>

<html>

    <body>

    <h2>HTML Input Text Box</h2>

    <input type="text" id="name" name="name" required minlength="3"
maxlength="10">

    </body>

</html>
```

Size Input Text Box

Using the Size attribute in the input html tag will indicating how many characters wide the input field should be. The value is numeric. Where the value must be a number greater than 0, and the default value is 20.

```
<input size="25" type="text">
```

Placeholder – How to use

Just write a placeholder attribute with its value (what hint you want to show) in a **<input>** tag.

```
<!DOCTYPE html>

<html>

    <body>
    <input type="text" id="name" name="name" placeholder="Enter name"

    </body>

</html>
```

The readOnly property sets or returns whether a **text** field is **read-only**, or not.
example

```
// Set a text field to read-only
document.getElementById("myText").readOnly = true;
```

The value attribute defines a default value which will be displayed in the element on page load
<input type="**text**" id="Name" class="form-**control** form-**control**-alternative" placeholder="Usernam" value="**myValue**">

Autofocus

Value	Description
true|false	Specifies whether a text field should get focus when the page loads, or not • true - The text field gets focus • false - Default. The text field does not get focus

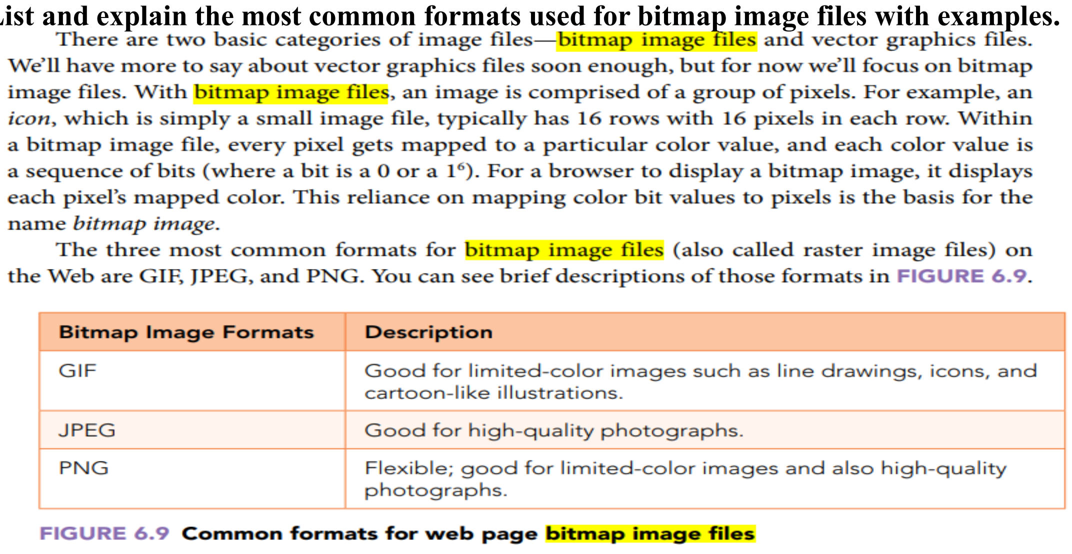
Code sample

```
// Find out if a text field automatically gets focus upon page load
var x = document.getElementById("myPassword").autofocus;
```

disabled

This Boolean disabled attribute indicates that the user cannot interact with the control or its descendant controls.
example
input disabled="true" />
<input disabled="false" /> <!-- still disabled! -->

List and explain the most common formats used for bitmap image files with examples.



There are two basic categories of image files—bitmap image files and vector graphics files. We'll have more to say about vector graphics files soon enough, but for now we'll focus on bitmap image files. With bitmap image files, an image is comprised of a group of pixels. For example, an *icon*, which is simply a small image file, typically has 16 rows with 16 pixels in each row. Within a bitmap image file, every pixel gets mapped to a particular color value, and each color value is a sequence of bits (where a bit is a 0 or a 1[6]). For a browser to display a bitmap image, it displays each pixel's mapped color. This reliance on mapping color bit values to pixels is the basis for the name *bitmap image*.

The three most common formats for bitmap image files (also called raster image files) on the Web are GIF, JPEG, and PNG. You can see brief descriptions of those formats in FIGURE 6.9.

Bitmap Image Formats	Description
GIF	Good for limited-color images such as line drawings, icons, and cartoon-like illustrations.
JPEG	Good for high-quality photographs.
PNG	Flexible; good for limited-color images and also high-quality photographs.

FIGURE 6.9 **Common formats for web page bitmap image files**

In creating a GIF file from an original picture, the original picture's colors are mapped to an 8-bit palette of colors. That means each pixel uses 8 bits, and those 8 bits determine the pixel's color. And the entire

set of colors forms the image's color palette. Each image has its own color palette with its own set of colors.
In creating a JPEG file from an original picture, the original picture's colors are mapped to a 24-bit palette of colors. With 24 bits, there are approximately 16 million permutations of 0's and 1's in each color value (224 = 16,777,216). That means approximately 16 million unique colors can be represented, and that's more colors than the human eye can discern.
The PNG format provides more flexibility in terms of clarity versus file size. In creating a PNG file from an original picture, you can choose to map each pixel to only a few bits (to create an image file that doesn't require much storage) all the way up to 64 bits per pixel (to create an image file that is very clear). The PNG format provides more flexibility in terms of transparency. You can create images with different levels of transparency for different parts of an image. GIF images can have only two levels of transparency—completely opaque or completely transparent. PNG images can have 256 levels of transparency.

How important are iframe elements? Explain.

The HTML `<iframe>` tag is used to embed a webpage within a webpage. It is also called an inline frame.
Iframes are most often used to **embed specific content from one web page** — like a video, form, document, or even a full web page — within a different web page. This is a powerful capability in HTML — you can take any content from any website (with permission) and place it on your own site to enhance your content.
For example,
<iframe src="https://programiz.pro" title="programiz pro website" height="500" width="500" ></iframe>
Here,

- `src`: It is used to specify the URL of the website to be loaded.
- `title`: It is good practice to include a `title` attribute so that screen readers can read the title to users.

Other Attributes for <iframe>

There are some importat attributes for `<iframe>`. They are:

- height and width
- name
- srcdoc

height and width

We can set the height and width of the `<iframe>` element with the `height` or `width` attribute. For example,

```
<iframe src="https://programiz.pro" height="200" width="300"></iframe>
```

We can also use CSS to set the width and height of the `<iframe>` using the `style` attribute. For example,

```
<iframe src="https://programiz.pro" style="height:200px;width:300px"></iframe>
```

name

The `name` attribute is used to specify the name for an iframe. It can be used as a target for other HTML elements like the `<a>` tag. For example,

```
<iframe src="https://parewalabs.com" name="iframe_target" height="500"
width="400"></iframe>

<a href="https://www.programiz.pro" target="iframe_target">Switch to Programiz
Pro</a>
```

srcdoc

Instead of a website URL, we can send HTML directly to the iframe, which will be displayed instead of another website. For example,

```
<iframe srcdoc="<h1>Learn to code</h1>"></iframe>
```

Mention the differences between relative URLs and path-absolute URLs

What are Absolute URLs?

An absolute URL contains the entire address from the protocol (HTTPS) to the domain name (www.example.com) and includes the location within your website in your folder system (/foldernameA or /foldernameB) names within the URL.Basically, it's the full URL of the page that you link to.
An example of an absolute URL is:

<a href = http://www.example.com/xyz.html>

What are Relative URLs?

The relative URL, on the other hand, does not use the full web address and only contains the location following the domain. It assumes that the link you add is on the same site and is part of the same root domain.The relative path starts with the forward slash and leads the browser to stay within the current site.
An example of a relative URL is:

<a href = "**/xyz.html**">

Why Choose Relative URLs?

1.Quicker Coding
Large websites are made much easier to code when you shorten your URL into a relative format.
2.Faster Load Times
Pages that use relative URLs will load more quickly than pages that use absolute URLs, for the most part, although the difference is minuscule at best.

Why Choose Absolute URLs?

1.Foils Scrapers
For URLs using the absolute method, it's harder for people to scrape information from your site directory using scraper programs. If you have all of your internal links as relative URLs, it would be very easy for a scraper to simply scrape your entire website and put it up on a new domain.
2.Disallows Duplicate Content
It's very important to use absolute URLs in order to avoid duplicate content issues. Imagine you have multiple versions of root domains that are indexed in Google without a canonical tag that points to the correct version of the site