



Full Stack Web Development



Day 6 - Introduction to HTML5

of HTML5 and its Tags

Difference between HTML and HTML5

Handling HTML5 Tags

APIs

Document Object Model (DOM)

- canvas

VG



Learning Objectives

If this module, you will be able to:

• solid understanding various tags in HTML

• with difference between the HTML and HTML5

• given web page's DOM structure to identify

• various shapes using canvas tag





Exploring HTML5 New Features and Enhancements

or Hypertext Markup Language 5, is the latest version standard markup language used to create web pages and applications.

solution of HTML, the language used to structure and content on the World Wide Web.

was developed by the *World Wide Web Consortium* and the *Web Hypertext Application Technology Working Group (WHATWG)* to improve the capabilities and functionality of web pages.



L5 (contd.)

was designed to cut out the need for non-standard proprietary technologies.

In this new version of HTML, you can create applications that work offline.

high-definition video and animations and where you are geographically located.



WHAT IS new in HTML5?

Making **code easier** to read for users and screen readers

Reducing the overlap between HTML, CSS, and JavaScript

Promoting design **responsiveness** and **consistency** across browsers

Supporting multimedia without the need for **Flash or other plugins**

HTML5 supports graphic elements using the tags <svg> and <canvas>

HTML5 included number, time, date, and calendar form components.

In HTML5, semantic elements (header, footer, article, and section) are present.

Tags in HTML5

Tags	Description
<header>	It is used inside the body tag. It is used to write the headings for our page. It is a block-level element.
<footer>	It is used inside the body tag. It is used to write the footers for our page. It is a block-level element.
<section>	It is used to define the section of the webpage, such as chapters, headers, footers, any other section. It is similar to the div tag.
<article>	It is used to define the articles in our web page. It is also a block-level element.

Tags in HTML5 (contd.)

Tags	Description
semantic	<header>, <footer>, <section> and <article> tags are semantic elements. These are block-level elements.
input types	Number, date, time, calendar are the new input form types added.
graphic	Scalable Vector Graphics (<svg>) and <canvas> elements are added.
style	Audio and video are added to your web page using <audio> and <video> elements.

HTML4 vs. HTML5

HTML4

`div id = “header”`

`div id = “post”`

`div id = “header”`

`div id = “post”`

versus

HTML5

`header`

`nav`

`article`

`article`

HTML4 vs. HTML5 (contd.)

HTML4	HTML5
es not support APIs.	It supports APIs.
not mobile-friendly.	It is mobile-friendly.
es not have drag and drop effects.	It has drag and drop effects.
es not have an integral SVG.	It has integral SVG.
es cookies to store the data.	It uses the local storage APIs to store data.
ng audio and video are not possible.	Adding audio and video are possible with <audio> and <video> tags.

L5 Tags

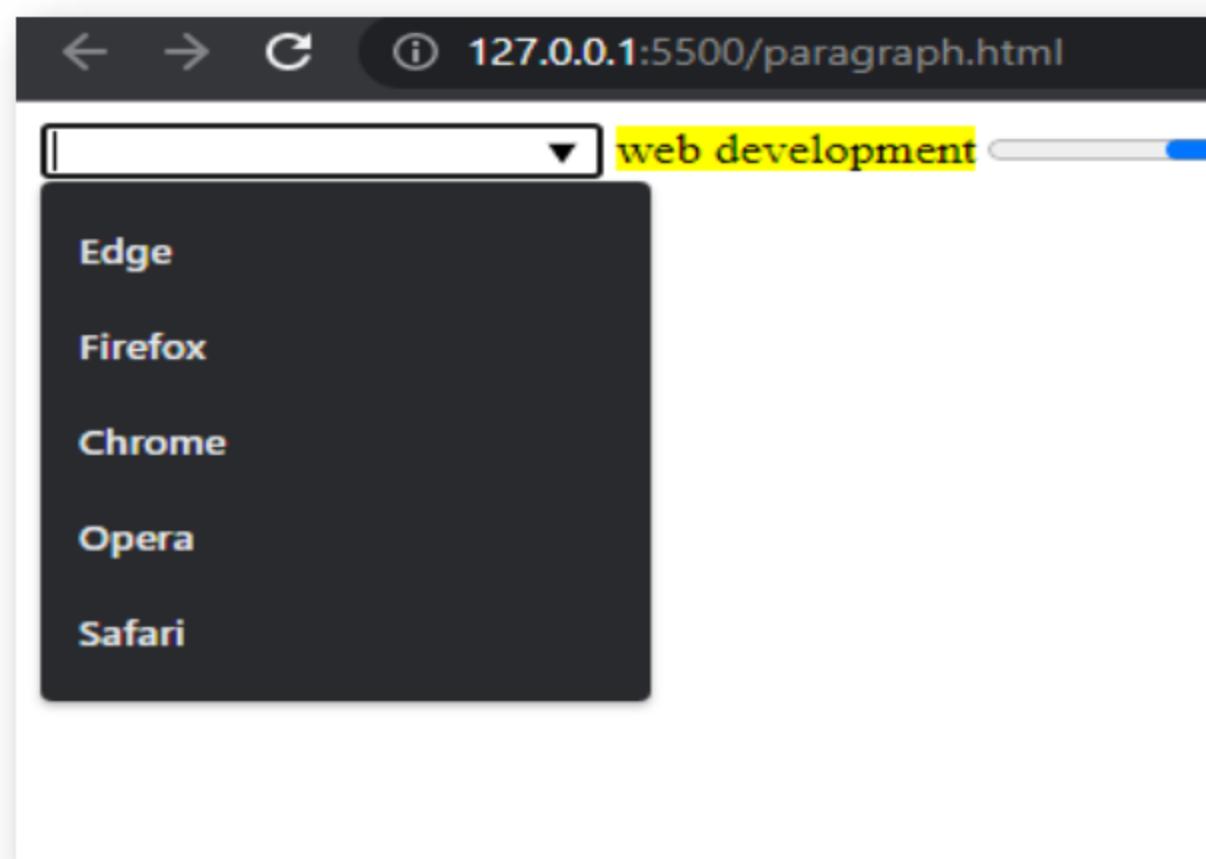
Tags	Description
<header>	The heading element is a block-level element used inside the body tag to define the headings for the web page.
<footer>	A block-level element used to create footer in a webpage.
<section>	It is similar to the div tag but is used to define specific sections of a webpage such as chapters, headers, and footers.
<article>	A block-level element used to define articles in webpages.

L5 Tags: <datalist> Tag

The <datalist> tag in HTML5 is used with the <input> element to display predefined options or suggestions for user input. Forms often use it to allow users to select an option from a list while they type.

Output:

```
list="browsers" >
<input id="browsers" type="text" value="web development"/>
<option value="Edge">
<option value="Firefox">
<option value="Chrome">
<option value="Opera">
<option value="Safari">
</datalist>
</input>
<mark> web development</mark>
```



L5 Tags: <audio> and <video> Tags

The <audio> tag allows direct embedding of audio content, offering an integrated playback experience without requiring additional plugins or players, while the <video> tag, an HTML5 element, achieves the same for video, ensuring a standardized experience.

With the <audio> and <video> tag, you can embed audio sources (e.g., MP3, WAV, OGG, WebM, Ogg) within <source> elements inside the <audio> or <video> tag.

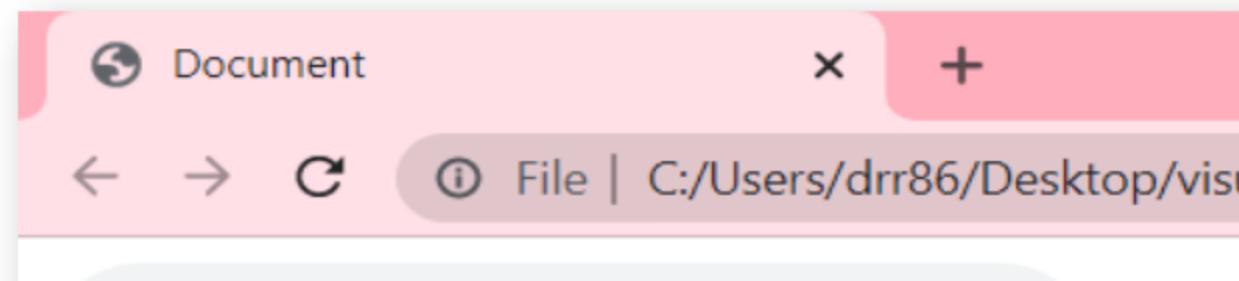
Ensures compatibility across various devices.

The controls attribute enables user-controlled playback controls like play, pause,

Example:

```
<audio controls>
  <source src="hey.mp3" type="audio/mpeg">
  Your browser does not support the audio
</audio>
```

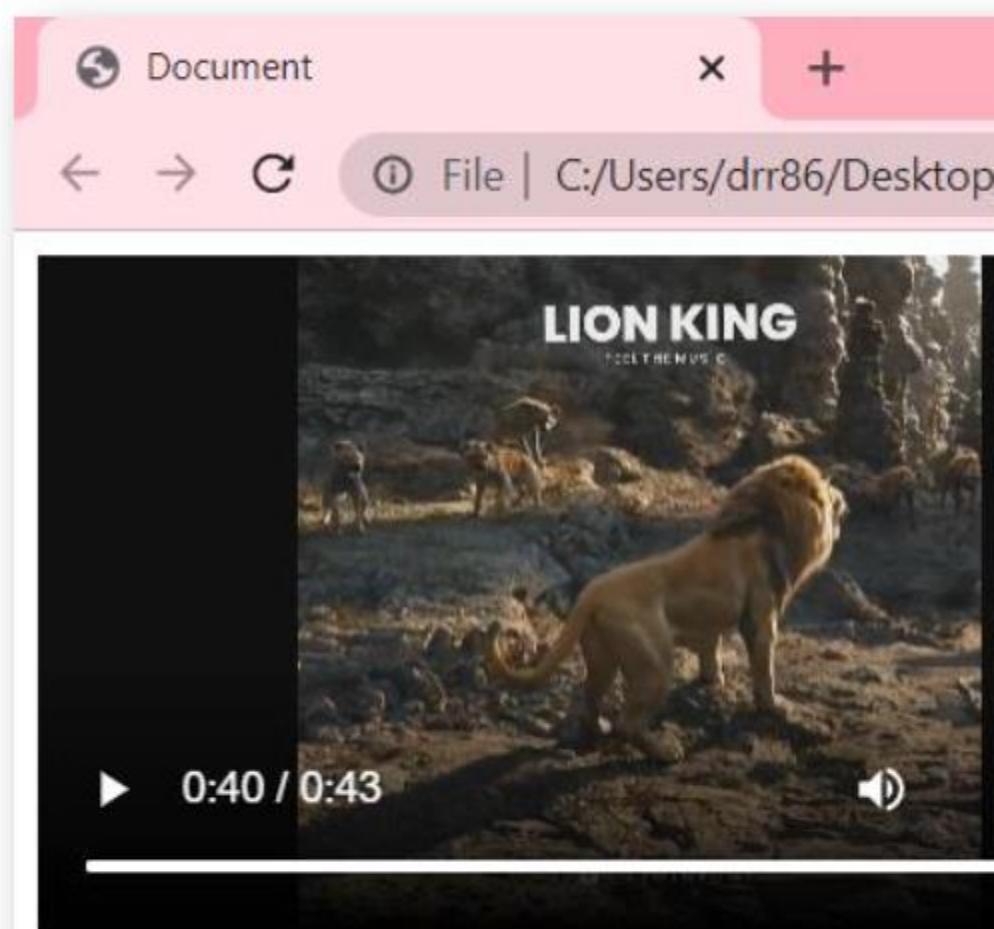
Output:



L5 Tags: <audio> and <video> Tags(c)

```
controls width="400px">  
src="./Lion King.mp4" type="video/mp4" />  
Your browser does not support the audio element.
```

Output:



HTML5 APIs

APIs, or HTML Application Programming Interfaces, are rules and protocols for communication between different software applications or components.

In web development, HTML APIs refer to the various interfaces and functionalities defined by the HTML5 specification to interact with web documents and applications.

These APIs extend the capabilities of HTML and JavaScript, enabling developers to build more powerful and feature-rich web applications.



HTML5 APIs (contd.)

In the realm of HTML5, there exist a multitude of useful APIs that are frequently employed by developers. Some of the most fascinating ones are outlined below for your perusal.

APIs	Description
Geolocation	It helps us to locate user's location.
Drag and Drop	It helps you to add a feature which allows user to drag and drop an object.
Local storage	It helps you to store the user's data locally in the web browser's.
Web workers	Specify whether the element will be displayed or not.

DOM Demystified

Understanding HTML5 DOM

The Document Object Model (DOM) is a programming interface provided by HTML5 that represents the structure and content of a web page as a hierarchical tree of objects.

cepts:

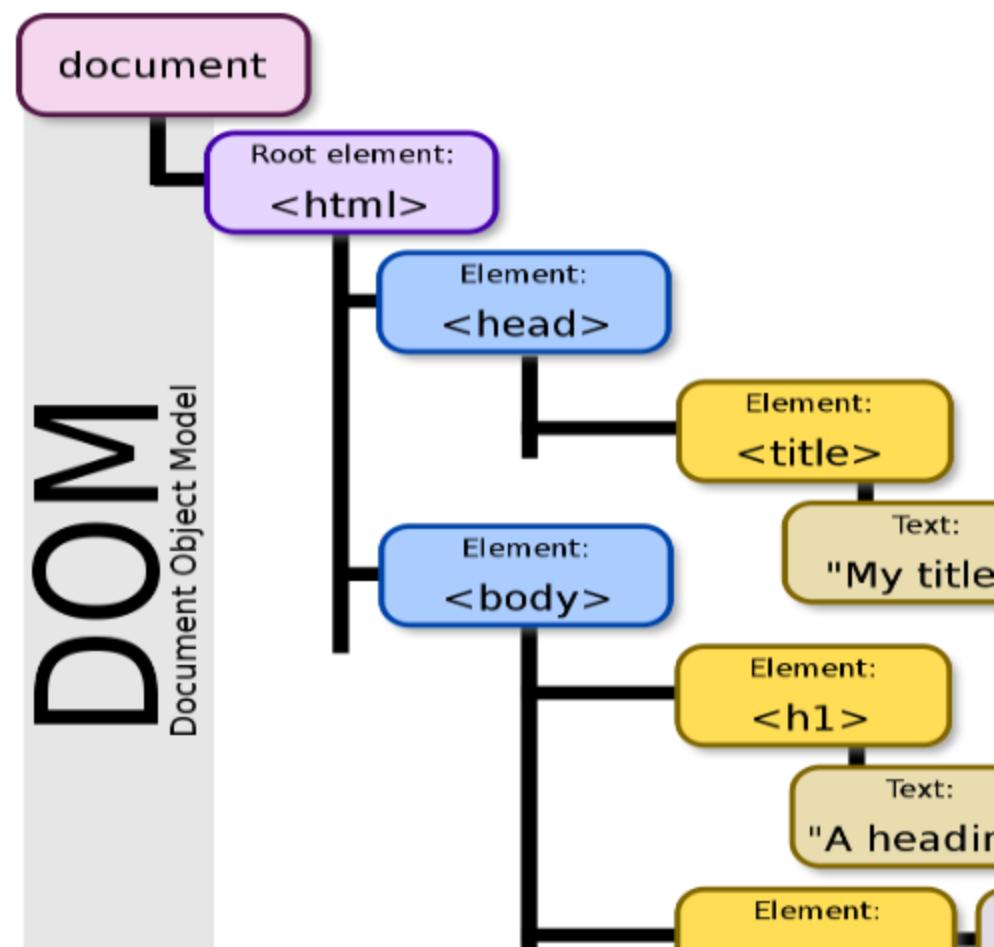
Hierarchy

and Manipulation

Handling

al

ic Updates



Understanding HTML5 DOM (contd.)

```
myParagraph">This is the original text.</p>
  id="myButton">Click me to change the text</button>
```

```
rences to the DOM elements
graph = document.getElementById("myParagraph");
button = document.getElementById("myButton");
// Add event listener to the button
button.addEventListener("click", () => {
  myParagraph.textContent = 'Text changed using the DOM!';
```

the original text.

Text changed using the DOM!

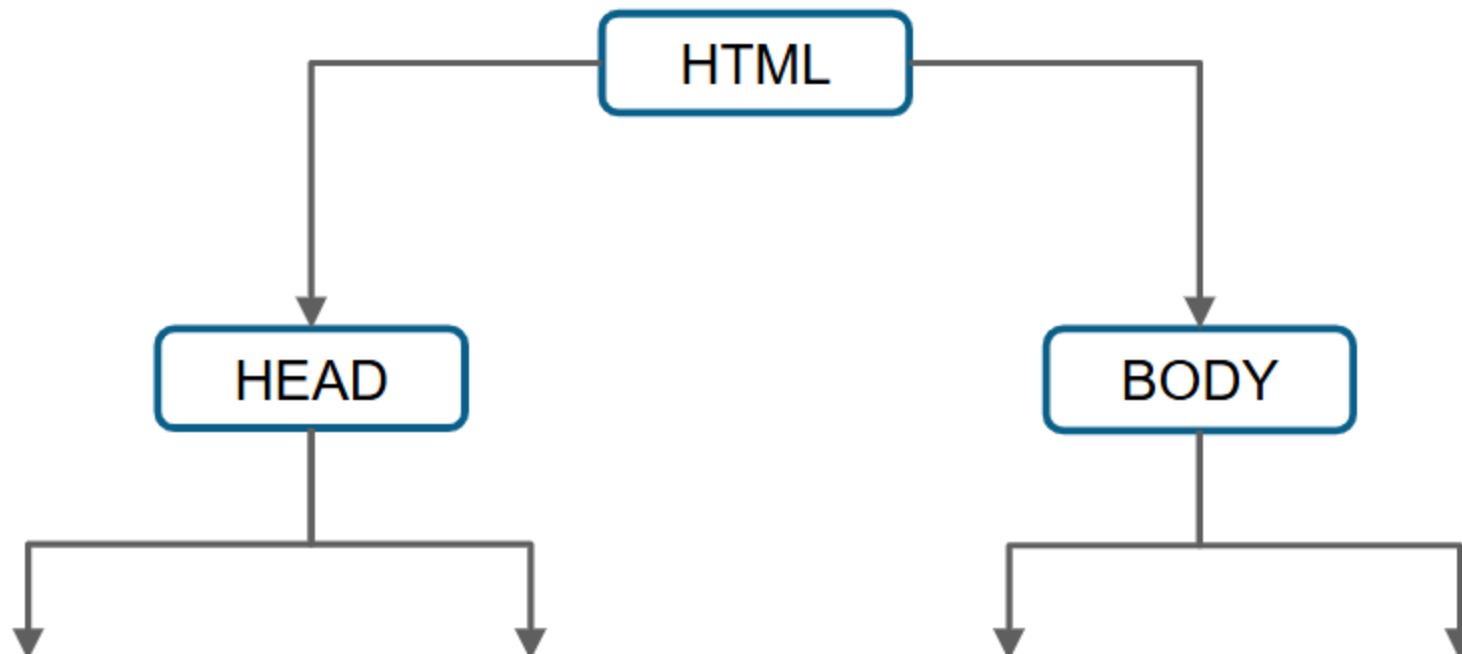
Understanding HTML5 DOM (contd.)

creates a document object model of the web page like a tree structure.

constructed as a tree of objects.

uses HTML elements as objects.

uses properties, methods, and events of all HTML elements.



HTML5 Tags – <canvas> Tag

The <canvas> tag allows you to draw powerful graphics easier using JavaScript.

The <canvas> tag is a new element introduced in HTML5.

You can use this tag to draw graphs, circles, rectangles, triangles, lines, etc.

You can easily find the <canvas> element in the Document Object Model using the `document.getElementById()` method.

The <canvas> tag is supported by Chrome, Firefox, Microsoft Edge, Opera, and most modern browsers.



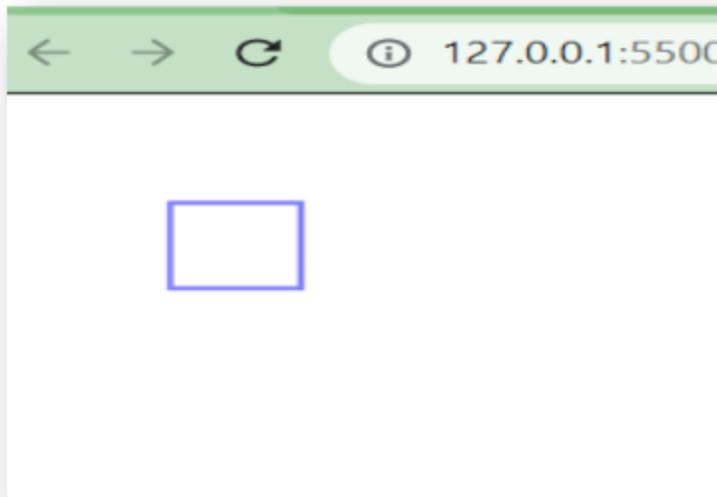
Properties of <canvas> Tag

Properties	Description
id	Specify the unique identity to an element.
class	Specify the identity to an element. This identity can or can't be unique.
style	Specify inline CSS style for an element.
width	Specifies the width of canvas element.
height	Specify the height of the canvas element.
fillRect(x,y, height)	This function helps us to draw a rectangle.
The first two parameters, x, and y specify the rectangle's position on the x axis.	The third parameter specifies the height of the rectangle.

Examples of <canvas> Tag

```
as" width="1000px" height="1000px" style="border:1px solid black;">  
document.getElementById("canvas");  
as.getContext("2d");  
e = "blue";  
(50, 50, 40, 40);
```

Output:



Methods of Canvas to Draw a Triangle

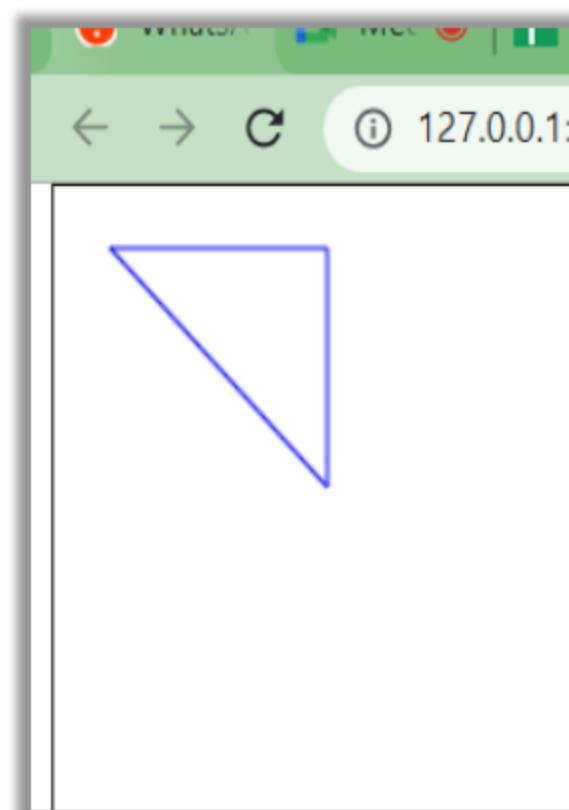
Methods	Description
<code>path()</code>	Specifies that canvas should begin the path.
<code>moveTo(x,y)</code>	Specifies where to start the line. The two x and y parameters specify the position on the x-axis and y-axis.
<code>lineTo(x,y)</code>	Specifies where to end the line. The two x and y parameters specify the position on the x-axis and y-axis.
<code>stroke()</code>	It draws the path that you have defined with the <code>moveTo(x,y)</code> and <code>lineTo()</code> commands.

Coding a Triangle Using Canvas

```
<canvas width="1000px" height="1000px" style="border:1px solid black;"></canvas>

document.getElementById("canvas");
var context = canvas.getContext("2d");
context.strokeStyle = "blue";
context.beginPath();
context.moveTo(20, 20);
context.lineTo(20, 100);
context.lineTo(100, 20);
context.closePath();
```

Output:



Methods of Canvas to Draw Circles

Methods	Description
stroke="color"	Specifies the color for the stroke of circle, rectangle, and triangle.
Used to draw and takes five parameters are:	
Parameters	Description
x	Specifies the starting position of the circle on the x-axis.
y	Specifies the starting position of the circle on the y-axis.
r	Specifies the radius of the circle.
sangle	Specifies the starting angle of the circle.

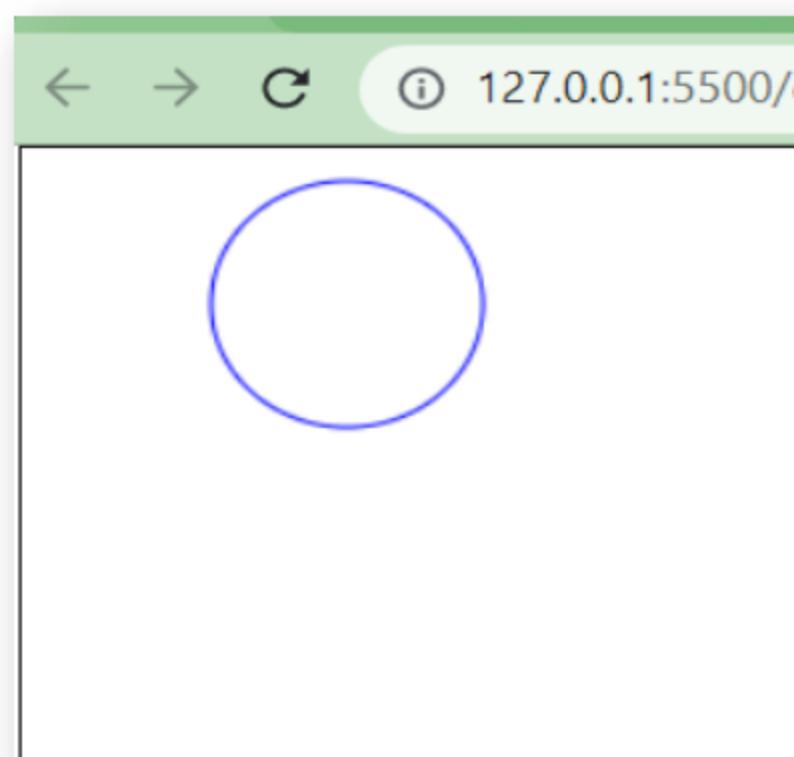
C

ing a Circle Using Canvas

```
as" width="1000px" height="1000px" style="border:1px solid black;"></canvas>

document.getElementById("canvas");
as.getContext("2d");
e = "blue";
);
, 40, 0, 180);
```

Output:



ble Vector Graphics (SVG)

ctor Graphics (SVG) is a dynamic and versatile graphics format that plays a pivotal role in m
design and development.

ages are created using **XML-based markup language**.

been developed by the **World Wide Web Consortium**.

In web development, SVG images are used as icons for web pages.

raph and **Font Awesome** are two popular web pages where

ages can be found.

A red rectangular decorative element with a white rounded corner on the left side. Inside the red area, the letters "SV" are written in a large, bold, white sans-serif font.

Photographic Experts Group (JPEG)

ch stands for Joint Photographic Experts Group, is a versatile and pervasive image file format that has played a pivotal role in shaping the digital visual landscape.

Joint Photographic Experts Group) is a popular image compression format used extensively in digital photography and mobile devices' development.

Images typically have file extensions like **.jpg** or **.jpeg**.

is a lossy compression format, which means it reduces file sizes by

removing some image data. This compression helps in storing and

mages efficiently.

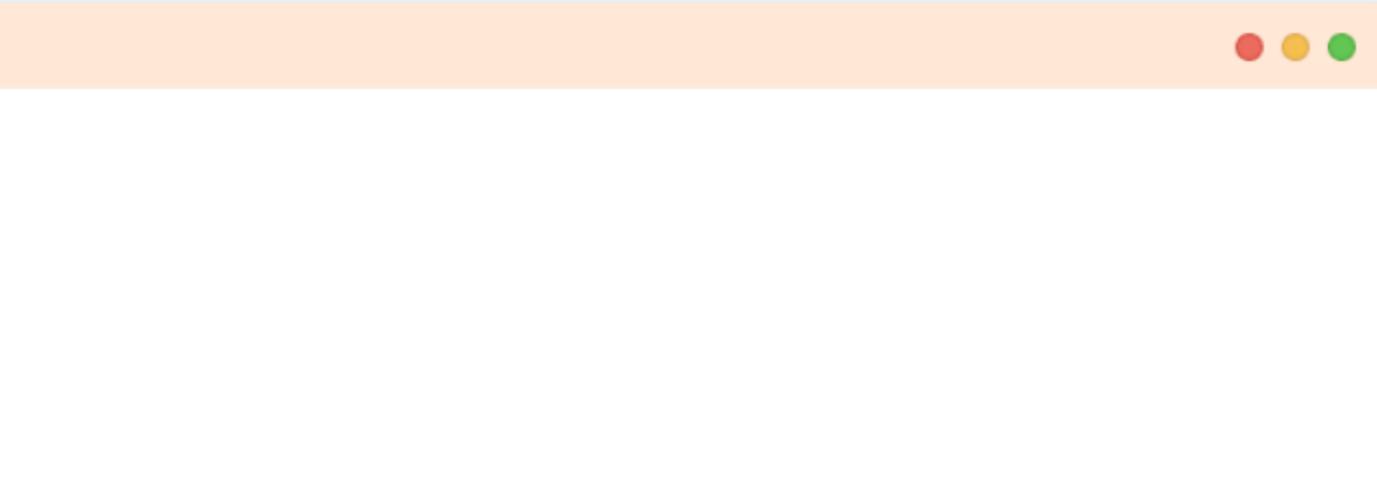


vs JPEG

ers	JPEG	SVG
or	JPEG stands for Joint Photographic Experts Group.	SVG stands for Scalable Vecto
pe	JPEG is a raster image format.	SVG is a vector image fo
ality	JPEG image quality decreases on zooming.	SVG image quality remains zooming.
ze	JPEG image is generally smaller than PNG image of same image.	SVG image is generally larger image of same image
e	JPEG images are not editable.	SVG images are text-based and edit.
ns	JPEG images use ".jpeg" or ".jpg" extension.	SVG images use ".svg" ext

Example of SVG

```
"100" height="100" style="border:1px solid black;"> <circle cx="50" cy="50" r="30"  
 > </svg>
```



Example of SVG (contd.)

```
"100" height="100" viewBox = "50 50 50 50" style="border: 1px solid black"> <circle  
"50" r="50" fill="blue" /> </svg>
```



Example of SVG (contd.)

```
= "100" height="100" viewBox = "0 0 1000 1000" style="border: 1px solid  
rcle cx="50" cy="50" r="50" fill="blue" /> </svg>
```



Example of SVG (contd.)

```
100" height="100" viewBox = "0 0 25 25" style="border: 1px solid black"> <circle  
"50" r="50" fill="blue" /> </svg>
```



Example of SVG (contd.)

```
= "100" height="150" viewBox="0 0 100 100" preserveAspectRatio="none"  
border: 1px solid black"> <circle cx="50" cy="50" r="50" /> </svg>
```



