TOP 100

HTML CSS BOOTSTRAP



Interview Questions

Happy Rawat

PREFACE

ABOUT THE BOOK

This book contains 200 very important JavaScript interview questions.

ABOUT THE AUTHOR

Happy Rawat has around 15 years of experience in software development.
He helps candidates in clearing technical interview in tech companies.

Top 100 HTML, CSS, Bootstrap Interview Q&A

HTML

CSS

Bootstrap

1. HTML Basics & Document Structure

Q. What is HTML? Differences between HTML & HTML5? Advantages of HTML5?

Q. What is the difference between HTML and XHTML?

Q. What is the role of DOCTYPE in HTML?

Q. What if you remove <!DOCTYPE html> from your HTML?

Q. Difference between Head & Body in HTML? Where to place JS link reference?

Q. What is Title Tag in HTML? What are the 4 advantages of Title tag?

Q. What are Meta Tags? What are the 5 types of meta tags?

Q. What is HTML? Differences between HTML & HTML5? Advantages of HTML5? V. IMP.

- HTML(HyperText Markup Language) is the standard **markup language** used to create web pages.
- Markup language meaning a language which define the **structure of a document using elements** like headings, paragraphs, links, lists, and more.
- **HTML is not a programming language**, it's a markup language like XML.

Hyper

Text

Markup

Language

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML</title>
  </head>
  <body>
    <a href="https://abc.com">Link</a>
    <h1>Heading Text</h1>
    <p>Para Text</p>
  </body>
</html>
```

Q. What is HTML? Differences between HTML & HTML5? Advantages of HTML5? V. IMP.

- HTML5 is the fifth and **latest version** of HTML.

Old HTML/ HTML4

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<html>
  <head>
    <title>Old HTML Example</title>
    <meta http-equiv="Content-Type"
    content="text/html; charset=UTF-8" />
  </head>
  <body>
    <h1>Tag</h1>
    <p>Paragraph</p>
    <a href="https://abc.com">Link</a>
  </body>
</html>
```

HTML/ HTML5

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>HTML5 Example</title>
  </head>
  <body>
    <header>
      <h1>Tag</h1>
    </header>
    <section>
      <p>Paragraph</p>
      <a href="https://abc.com">Link</a>
    </section>
  </body>
</html>
```

Q. What is HTML? Differences between HTML & HTML5? Advantages of HTML5? V. IMP.

5 Advantages of HTML5:

1. New Semantic Elements: <header>, <nav>, <article>, <section>, <aside>, <footer>
2. Form Input Types: <input type="date">, <input type="email">
3. Audio and Video Support: <audio>, <video>
4. Mobile Compatibility
5. Simpler Code

Q. What is the difference between HTML and XHTML?

- XHTML(eXtensible HyperText Markup Language) is a markup language that follows the **rules of XML** to define the structure of web pages.

XHTML

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html
PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    <title>XHTML Example</title>
  </head>
  <body>
    <h1>XHTML</h1>
    <img src="image.jpg" />
  </body>
</html>
```

HTML/ HTML5

```
<!DOCTYPE html>
<html>
  <head>
    <title>HTML</title>
  </head>
  <body>
    <h1>HTML</h1>
    <img src="image.jpg">
  </body>
</html>
```

Q. What is the difference between HTML and XHTML?

HTML/ HTML5	XHTML
1. HTML has more **lenient syntax** rules. Closing tag is not mandatory for some elements. <img src="image.jpg" alt="Description">.	XHTML follows **stricter syntax** rules. All tags must be properly nested and closed.
2. HTML is **not case-sensitive**. <DIV>, <P> will work.	XHTML is case-sensitive. , <p> will work.
3. HTML is **widely supported** by all browsers and web platforms.	XHTML has limited support by browsers.

Q. What is the difference between HTML and XHTML?

HTML/ HTML5	XHTML
1. HTML has more **lenient syntax** rules. Closing tag is not mandatory for some elements. <img src="image.jpg" alt="Description">.	XHTML follows **stricter syntax** rules. All tags must be properly nested and closed.
2. HTML is **not case-sensitive**. <DIV>, <P> will work.	XHTML is case-sensitive. , <p> will work.
3. HTML is **widely supported** by all browsers and web platforms.	XHTML has limited support by browsers.

Q. What is the role of DOCTYPE in HTML? V. IMP.

- DOCTYPE(Document Type) declaration specifies the **version** of HTML.
- DOCTYPE tells the browser which version of HTML it is and how to interpret the code.

```html
<!DOCTYPE html>
<html>
  <head>
    <title>DocType</title>
  </head>
  <body>
    <h1>DocType</h1>
  </body>
</html>
```

```html
<!-- Different kind of Doctypes -->
<!-- HTML 4.01 Strict -->
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">

<!-- XHTML 1.0 Strict: -->
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

<!-- HTML5 (Preferred for modern web development): -->
<!DOCTYPE html>
```

Q. What if you remove <!DOCTYPE html> from your HTML?

- Then browsers can still render the page, but they will not be able to validate the version of HTML, therefore it may lead to some **compatibility issues** with SEO or debugging.

```
<!DOCTYPE html>
<html>
  <head>
    <title>DocType</title>
  </head>
  <body>
    <h1>DocType</h1>
  </body>
</html>
```

Q. Difference between Head & Body in HTML? Where to place JS link reference? V. IMP.

- The <head> element is where you place **meta-information** (information about the document). For example, <title>, <meta>, <link>, <script>, <style> are normally kept under head element.

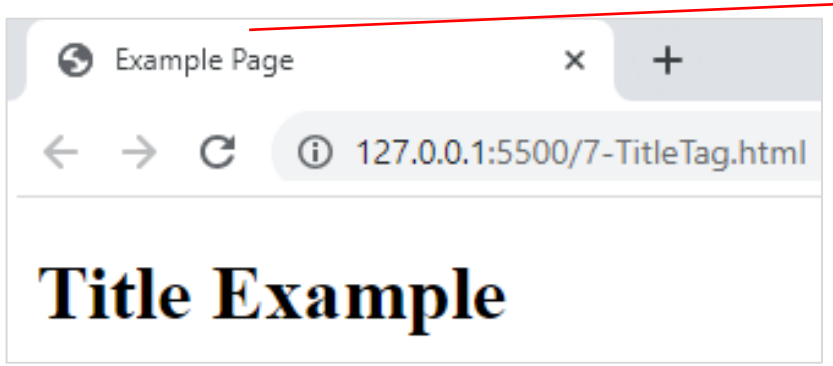
- The <body> element is where you place the **actual content** of your HTML web page. For example, , <h1>, <p>, <img>, <a> are normally kept under body element.

- Head will load before the body, therefore, if you are manipulating HTML elements in your JS functions, then place the JS link at the end of the body section because until then all the elements will be loaded. Else, place it inside the head tag.

```
<!DOCTYPE html>
<html>
<head>
    <title>Sample Page</title>
    <meta charset="UTF-8">
    <link rel="stylesheet" href="styles.css">
    <script src="script.js"></script>
</head>
<body>
    <h1>Hello, World!</h1>
    <p>This is a sample webpage.</p>
    <img src="image.jpg">
</body>
</html>
```

Q. What is Title Tag in HTML? What are the 4 advantages of Title tag? V. IMP.

- The <title> tag in HTML is used to define the **title** of a web page.

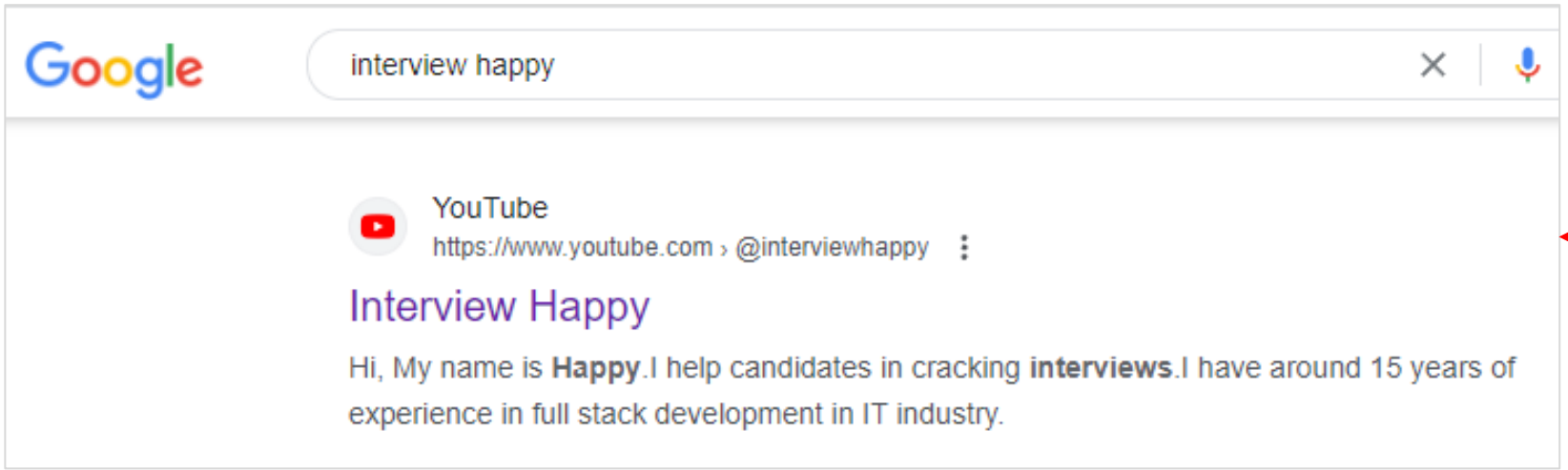



```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Example Page</title>
  </head>
  <body>
    <h1>Title Example</h1>
  </body>
</html>
```

Q. What is Title Tag in HTML? What are the 4 advantages of Title tag? V. IMP.

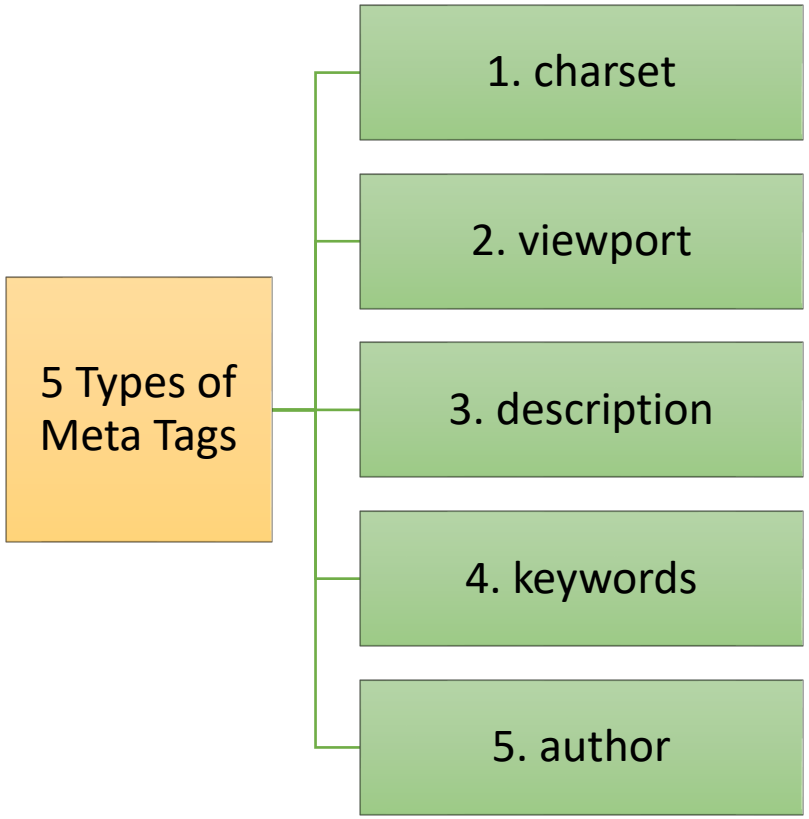
❖ Advantages of title tag:

1. **Window Title:** <title> element text is displayed in the tab of the browser.
2. **SEO:** Search engines use the title as the main heading for search results.
3. **Bookmarks/Favorite:** Title tag is used as the default name for the bookmark.
4. **Social Media Sharing:** Title tag is used as the default title in the shared post.



Q. What are Meta Tags? What are the 5 types of meta tags?

- Meta tags in HTML are elements used to provide metadata or additional information about a web page.




```html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Example Page</title>
  </head>
  <body>
    <h1>Title Example</h1>
  </body>
</html>
```

Q. What are Meta Tags? What are the 5 types of meta tags?

1. Character Encoding:
This helps browsers to interpret the different characters in the document. The charset="UTF-8" attribute ensures that characters from **different languages** will be displayed correctly in the browser.

2. Responsive Design:
viewport meta tag is crucial for creating mobile-friendly, **responsive web designs** for various screen sizes.

3. Description for SEO:
This meta tag provides description of the content of the page. Search engines may use this for search results.

4. Keywords for SEO:
This meta tag used to be important for search engines, but it's now of less importance.

5. Author Info:
This meta tag can be used to specify the author of the document.

```html
<html>
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Interview Happy</title>
    <meta name="description" content="Technical interview preparation" />
    <meta name="keywords" content=".NET, Angular, React, JS, HTML" />
    <meta name="author" content="Happy Rawat" />
  </head>
```

2. HTML Elements & Tags – Part I

Q. What are HTML Elements? What is the difference between Element & Tag?

Q. What are the roles and uses of the element in HTML?

Q. What is the difference between and element?

Q. What is the role of <p>, <a>,
, <hr>, , <img>, <input> & <button> elements?

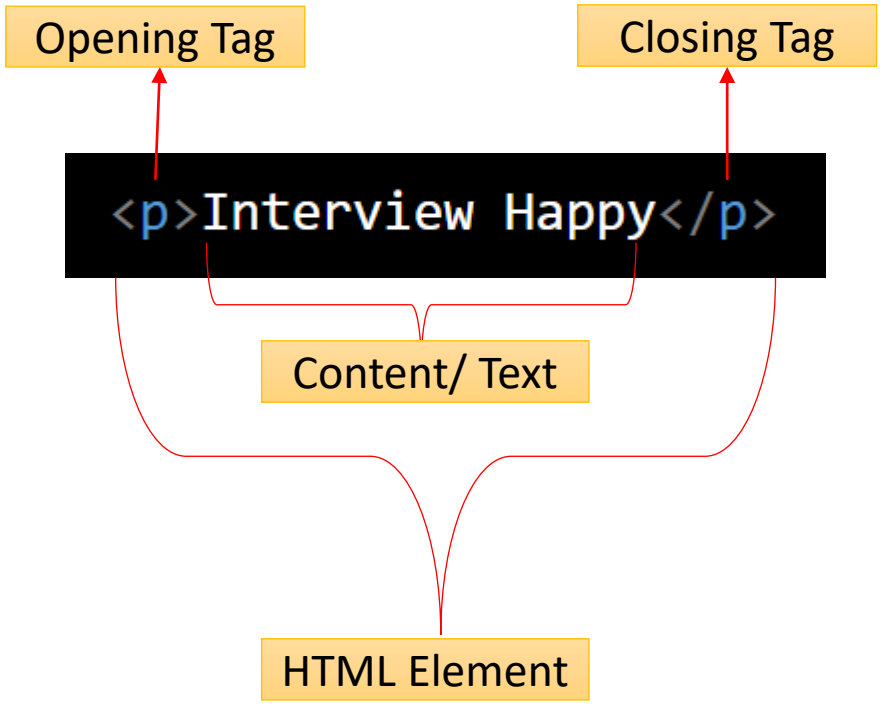
Q. What is the role of header, main, section, footer & address elements in HTML?

Q. What is the role of Heading tags in HTML? How does it impact SEO?

Q. What is the difference between <section> & <article> elements?

Q. What are HTML elements? What is the difference between Element & Tag? V. IMP.

- HTML elements are the **building blocks** of web pages.
- Element consists of a start tag, content, and an end tag.
- Tag is a specific part of an element that denotes the beginning or end of that element.




```
<!DOCTYPE html>
<html>
  <head>
    <title>Title Element</title>
  </head>
  <body>
    <h1>Heading Element</h1>
    <ul><!-- List Element -->
      <li>List Item 1</li>
      <li>List Item 2</li>
    </ul>
    <p>Paragraph Element</p>
    <div>Division Element</div>
    <img src="ImageElement.jpg" />
    <a href="~/AnchorElement.html"></a>
  </body>
</html>
```

Q. What are HTML elements? What is the difference between Element & Tag? V. IMP.

Top 10 most common used HTML elements

1. **Paragraph Element**: Defines a block of text.<p>.
2. **Heading Elements**: <h1> <h2><h3><h4><h5><h6>.
3. **Header and Footer Elements**: <header> and <footer>.
4. **Sectioning Elements**: <section>, <article>, <nav>, <aside>, <main>.
5. **Image Element**: Embeds images into a webpage.<img> (self-closing).
6. **List Elements**: <ul> (unordered list), <ol> (ordered list)), <li> (list item).
7. **Anchor Element (Link)**: Creates hyperlinks to other pages or resources.<a>.
8. **Division Element**: groups together content for applying styles or scripting. .
9. **Table Elements**: <table>, <tr> (table row), <th> (table header) and <td> (table data).
10. **Form Elements**: Used for creating forms.<form>, <input>, <button>, <select>, <textarea>, etc.

Q. What are the roles and uses of the element in HTML? V. IMP.

- The (division) element in HTML is a **container** that is used to **group and structure** the content on a webpage.

- **Top 3 uses of element:**

1. Grouping & structuring Content:

- It allows you to group together related elements.

2. Styling and Layout:

- elements are used to apply common styles or css to grouped elements.

3. Scripting:

- JavaScript and other scripting languages can target elements to manipulate their content or behavior.

```
<body>
  <div class="container">

    <h1>Heading 1</h1>
    <p>Paragraph 1</p>

  </div>
</body>
```

Heading 1

Paragraph 1

Q. What is the difference between and element?

- The (division) element in HTML is a **container** that is used to **group and structure** the content on a webpage.
- The element in HTML is an **inline container** used to apply styles or scripting to a specific section of text or content.
- is a block-level element and is an inline element.

```
<body>
  <div>
    <h1>Main Heading</h1>
    <p>Interview <span style="color: blue">Happy</span></p>
  </div>
</body>
```

Q. What is the role of <p>, <a>,
, <hr>, , <img>, <input> & <button> elements?

1. <p>

- Defines a paragraph of text.

This is a paragraph.

```
<p>This is a paragraph.</p>
```

2. <a>

- Creates a hyperlink. The href attribute specifies the url and content is the text of hyperlink.

link.

```
<a href="https://abc.com">link</a>.
```

3.

- Inserts a line break.

first line.
second line.

```
<p>first line.<br />second line.</p>
```

4. <hr>

- Creates a horizontal line break.

```
<hr />
```

Q. What is the role of <p>, <a>,
, <hr>, , <img>, <input> & <button> elements?

**1. **

- Emphasizes text displayed in italics.

inline *italics text.*

```
<p>inline <em>italics text.</em></p>
```

2. <img>

- Embeds an image. The src attribute specifies the image file, and alt provides alternative text.

Image

```
<img src="abc.jpg" alt="Image" />
<br /><br />
```

3. <input>

- Creates an input field within a form.

```
<input type="text" id="user" />
<br /><br />
```

4. <button>

- Creates a clickable button.

Click

```
<button type="button">Click</button>
```

Q. What is the role of header, main, section, footer & address elements in HTML?

- <header>, <main>, <section> and <footer> are semantic elements, used to define the layout and **structure of a webpage** in a meaningful and organized way.

<header>	Contains the header content of website.
<main>	Contains the main content of the document.
<footer>	contains the group related content.
<address>	provides contact information.

Website Header

Section 1

Section 2

India

```
<body>
  <header>
    <h1>Website Header</h1>
  </header>
  <main>
    <section id="section1">
      <h2>Section 1</h2>
    </section>
    <section id="section2">
      <h2>Section 2</h2>
    </section>
  </main>
  <footer>
    <address>India</address>
  </footer>
</body>
```

Q. What is the role of Heading tags in HTML? How does it impact SEO?

- Heading tags in HTML are used to define the **headings of sections** within a webpage.

Electronics Mobiles Samsung Mobiles Samsung Galaxy S Series Samsung Galaxy S20 Samsung Galaxy S20 Ultra

- Top 3 uses of heading tags:

1. Organization & Readability: Heading tags organize the content in better way, making it easier for users to understand the organization of the page.
2. SEO (Search Engine Optimization): Search engines use headings to understand the importance of content.
3. Styling and Layout: For example, same CSS can be applied to all h1 tags to show their importance. That will help in creating consistent layouts.

Electronics

Mobiles

Samsung Mobiles

Samsung Galaxy S Series

Samsung Galaxy S20

Samsung Galaxy S20 Ultra

```
<body>
  <h1>Electronics</h1>

  <h2>Mobiles</h2>

  <h3>Samsung Mobiles</h3>

  <h4>Samsung Galaxy S Series</h4>

  <h5>Samsung Galaxy S20</h5>

  <h6>Samsung Galaxy S20 Ultra</h6>
</body>
```

Q. What is the difference between <section> & <article> elements?

- The <section> element is a generic container used to **group related content together**.

- The <article> element represents a self-contained or **independent piece of content** with a title and content. For example: a blog post, a news article, a forum post, a comment, etc.

- Article example: offer


```
<body>
  <section>
    <h2>Mobile Section</h2>
    <p>Samsung, Nokia etc</p>
  </section>
  <section>
    <h2>Books Section</h2>
    <p>Technology, Philosphy etc</p>
  </section>

  <article>
    <h1>Festival Sale</h1>
    <p>10% Discount on all items</p>
    <footer>
      <p>Terms and conditions apply</p>
    </footer>
  </article>
</body>
```

3. HTML Elements & Tags – Part II

Q. What are Root, Parent, Child & Nested elements?

Q. What are Empty Elements?

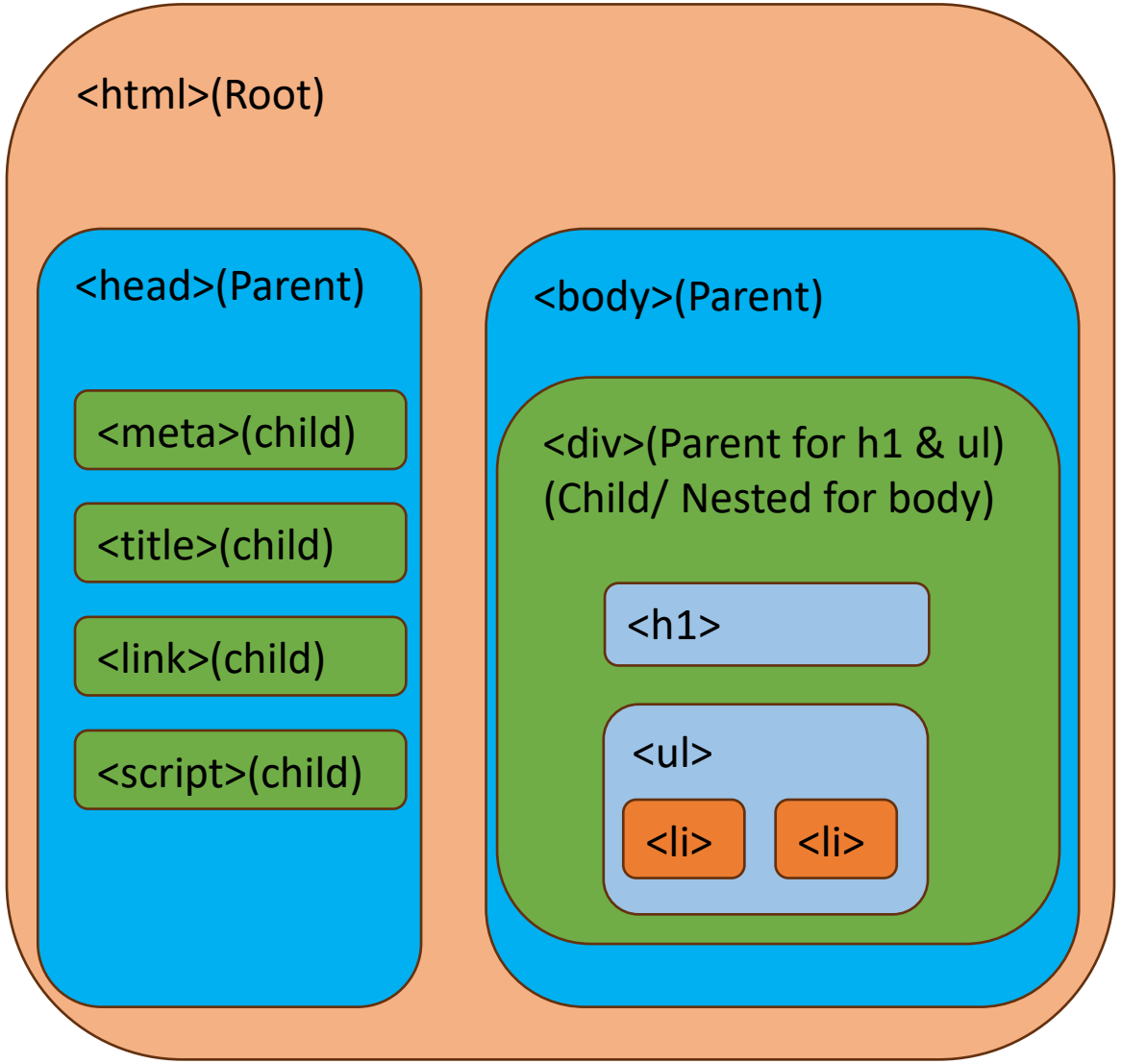
Q. What are Semantic Elements in HTML?

Q. Can HTML tags be written in Uppercase?

Q. What are the 3 differences between Block-Level & Inline Elements?

Q. List all Block-Level & Inline Elements in HTML.

Q. What are Root, Parent, Child & Nested elements? V. IMP.




```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Nested Elements</title>
    <link rel="stylesheet" href="styles.css" />
    <script src="script.js"></script>
  </head>
  <body>
    <div>
      <h1>Inside container</h1>
      <ul>
        <li>Item 1</li>
        <li>Item 2</li>
      </ul>
    </div>
  </body>
</html>
```

Q. What are Root, Parent, Child & Nested elements? V. IMP.

1. The **root element** is the highest-level element in the hierarchy of an HTML document. In HTML5, the root element is <html>.

2. A **parent element** is an element that contains other child elements.

3. A **child or nested element** is an element that is contained within a parent element. Child and nested elements are same.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <title>Nested Elements</title>
    <link rel="stylesheet" href="styles.css" />
    <script src="script.js"></script>
  </head>
  <body>
    <div>
      <h1>Inside container</h1>
      <ul>
        <li>Item 1</li>
        <li>Item 2</li>
      </ul>
    </div>
  </body>
</html>
```

Q. What are Empty Elements?

- An empty element in HTML is an element that doesn't need content between opening and closing tags.
- Empty elements are also called a **self-closing** or void elements.
- Empty elements in HTML: <img>, <input>,
, <hr>, <meta>, <link>, <area>, <base>, <col>, <embed>.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Empty Elements Example</title>
    <link rel="stylesheet" href="style.css" />
    <meta charset="UTF-8" />
  </head>
  <body>
    <img src="test.jpg" alt="Example" />
    <input type="text" />
    <br />
  </body>
</html>
```

Q. What are Semantic Elements in HTML? Is div a semantic element? V. IMP.

- Semantic elements in HTML are elements that provide **meaning to the content** they contain.
- is not a semantic element, because div is a general-purpose structural element. It doesn't give any meaning to the content.

Top 5 Semantic elements

1. <header>
2. <main>
3. <section>
4. <footer>
5. <address>

```
<body>
  <header>
    <h1>Website Header</h1>
  </header>
  <main>
    <section id="section1">
      <h2>Section 1</h2>
    </section>
  </main>
  <aside>
    <h2>Aside Content</h2>
  </aside>
  <footer>
    <address>India</address>
  </footer>
</body>
```

Q. What are Semantic Elements in HTML? Is div a semantic element? V. IMP.

❖ **10 more semantic elements:**

1. **<progress>** - Displays the progress of a task.
2. **<nav>** - Contains navigation links for a webpage.
3. **<time>** - Represents a specific period in time or a date.
4. **<mark>** - Highlights parts of the text for reference or emphasis.
5. **<summary>** - Provides a summary, caption, or legend for a <details> element.
6. **<meter>** - Represents a scalar measurement within a known range (like a gauge).
7. **<figure>** - Contains content like images, illustrations, diagrams, etc., along with a caption.
8. **<details>** - Represents additional information or controls that can be toggled open or closed.
9. **<aside>** - Contains content that is related to the main content but can be considered separate.
10. **<article>** - Represents a self-contained composition in a document, such as a blog post or a news article.

Q. Can HTML tags be written in Uppercase?

- **Yes**, HTML tags not case-sensitive therefore can be written in uppercase, lowercase, or a combination of both.
- But it is not recommended as per standards.

```
<!DOCTYPE HTML>
<HTML>
  <HEAD>
    <TITLE>ALL UPPERCASE HTML</TITLE>
  </HEAD>
  <BODY>
    <H1>HELLO, THIS IS AN H1 HEADING</H1>
    <P>THIS IS A PARAGRAPH.</P>
    <A HREF="https://www.abc.com">THIS IS A LINK</A>
  </BODY>
</HTML>
```

Q. What are the 3 differences between Block-Level and Inline Elements? V. IMP.

1. Block-level elements create "blocks" of content.
   , <p>, <h1>, <ul>, <li>, <table>, <form> etc.
2. By default, Block-level elements start on a new line.
3. You can set both width and height for block-level elements.

1. Inline elements length depends on their content length.
   , <a>, , , <img>, <input>,
 etc.
2. Inline elements do not start on a new line.
3. You can't set width and height for inline elements.

```
<!DOCTYPE html>
<html>
  <head>
    <title>BlockLevel Elements</title>
  </head>
  <body>
    <div>block-level element</div>
    <p>block-level element</p>
  </body>
</html>
```

```
<!DOCTYPE html>
<html>
  <head>
    <title>Inline Elements</title>
  </head>
  <body>
    <p>My name is<strong> Happy</strong></p>
    <p>Click <a href="https://abc.com">here</a></p>
  </body>
</html>
```

Q. List all Block-Level & Inline elements in HTML.

❖ Top 20 block-level elements in HTML:

1. **<h1>, <h2>, <h3>, <h4>, <h5>, <h6>:** Heading elements from highest (h1) to lowest (h6) level.
2. **<header>:** Represents the introductory content or a group of navigational links.
3. ** :** The generic container element used for grouping and applying styles.
4. **<footer>:** Represents the footer of a section or document.
5. **<nav>:** Defines a section with navigation links.
6. **<section>:** Defines a section of a document.
7. **<ol>:** Creates an ordered (numbered) list.
8. **<p>:** Represents a paragraph of text.
9. **<li>:** Represents a list item in a list.
10. **<ul>:** Creates an unordered list.

Q. List all Block-Level & Inline elements in HTML.

❖ Top 20 block-level elements in HTML(Continued…):

11. **<article>:** Represents a self-contained piece of content, such as a blog post or article.
12. **<aside>:** Represents content related to the main content but is considered separate.
13. **<blockquote>:** Represents a section that is quoted from another source.
14. **<figure>:** Used for encapsulating media and its caption.
15. **<form>:** Used for creating forms to collect user input.
16. **<hr>:** Creates a horizontal rule (a thematic break).
17. **<figcaption>:** Represents a caption for a <figure>.
18. **<table>:** Defines structure of a table.
19. **<tr>:** Represents a table row.
20. **<td>:** Represents a table cell.

Q. List all Block-Level & Inline elements in HTML.

❖ Top 20 inline elements in HTML:

1. **:** A generic container that doesn't have any specific semantic meaning.
2. **:** Indicates strong importance, often displayed as bold text.
3. **:** A generic container used for applying styles or scripting.
4. **:** Emphasizes on text, often displayed as italicized text.
5. **<small>:** Reduces the text size, often used for fine print.
6. **<a>:** Creates hyperlinks to other pages or resources.
7. **<label>:** Provides a label for an input element.
8. **<input>:** Used to create form input fields.
9. **<code>:** Represents a snippet of code.
10. **<cite>:** Specifies the title of a work.

Q. List all Block-Level & Inline elements in HTML.

❖ Top 20 inline elements in HTML(Continued…):

11. **
:** Inserts a line break, forcing content after it onto a new line.
12. **<select>:** Creates a dropdown menu for selecting options.
13. **<time>:** Represents a specific period in time or a date.
14. **<small>:** Reduces the text size, used for fine print.
15. **<abbr>:** Defines an abbreviation or acronym.
16. **<del>:** Indicates deleted or removed text.
17. **<q>:** Defines a short inline quotation.
18. **<sup>:** Renders text as superscript.
19. **<sub>:** Renders text as subscript.
20. **<img>:** Embeds an image.

4. HTML Attributes

Q. What are HTML Attributes? What are the Types of HTML attributes?

Q. What are the Id, Style & Class attributes of an element? When to use what?

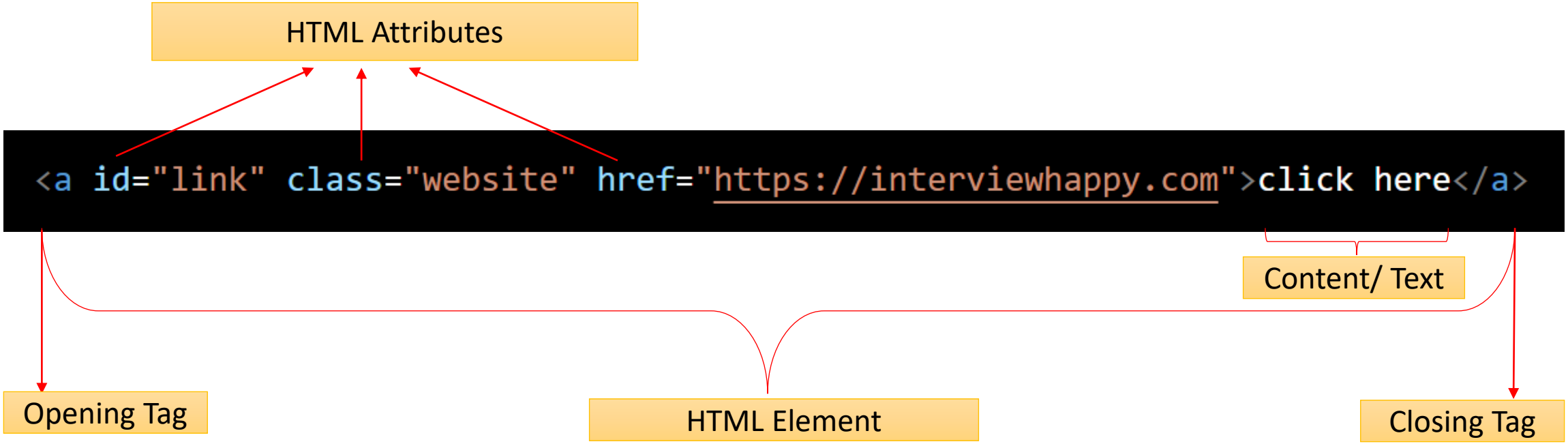
Q. What will happen if two elements have same ids?

Q. How to specify Multiple Classes for single element? What is the style precedence?

Q. What are Data Attributes in HTML?

Q. What are HTML Attributes? What are the Types of HTML attributes? V. IMP.

- HTML attributes provide **additional information** about HTML elements.



Q. What are HTML Attributes? What are the Types of HTML attributes? V. IMP.

❖ **Types of attributes:**

1. Common Attributes (Global Attributes)

- Example: class, id, style, data-*
- They are common and applicable for all the elements.

2. Specific Attributes

- Example: href, src, alt, width, height, target, rel, type, value, name, placeholder, disabled, readonly, checked, selected
- These are specific to the elements.

Q. What are the Id, Style & Class attributes of an element? When to use what? V. IMP.

- **id** attribute is used to **uniquely identify** an element on a page.
  - The primary purpose of the id attribute is to allow JavaScript and CSS to target and manipulate specific elements.
- **style** attribute allows you to apply **inline styles** directly to an individual element.
- **class** attribute is used to group together multiple elements that **share common styles**.
  - classes are recommended for large website not inline styles.

```
<html>
  <head>
    <title>Id, Style and Class</title>
    <style>
      .highlighted {
        color: red;
      }
    </style>
  </head>
  <body>
    <h1 id="uniqueId">Main heading</h1>

    <p style="color: blue">Para 1</p>

    <p class="highlighted">Para 2</p>
    <p class="highlighted">Para 3</p>
  </body>
</html>
```

Q. What will happen if two elements have same ids?

- May be in browser there is no direct impact, but it will be considered as **invalid HTML.**
- Invalid HTML means it can lead to **unexpected behavior** in your webpage. For example, problems can occur in css styling and JavaScript interactions.

```
<html lang="en">
  <head>
    <title>uplicate IDs</title>
  </head>
  <body>
    <div id="uniqueId1">First Element</div>
    <p id="uniqueId1">Second Element</p>
  </body>
</html>
```

Q. How to specify Multiple Classes for single element? What is the style precedence?

- To specify multiple classes for an element in HTML, you can simply add a **space-separated list of class names** within the class attribute.
- In case of multiple classes with same style attribute, style defined in the **last class** will take precedence.

single class

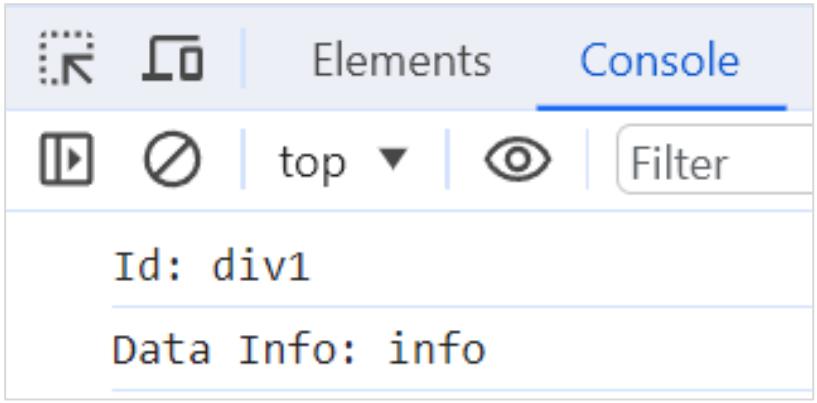
multiple classes

```html
<html>
  <head>
    <title>Multiple Classes Example</title>
    <style>
      .class1 {
        color: red;
      }
      .class2 {
        font-size: 20px;
        color: green;
      }
    </style>
  </head>
  <body>
    <div class="class1">single class</div>
    <div class="class1 class2">multiple classes</div>
  </body>
</html>
```

Q. What are Data Attributes in HTML? V. IMP.

- Data attributes in HTML provide a way to add **custom data attributes** to add additional informational in elements.
- Data attributes are specified using the **data-** prefix.
- Data attributes can be accessed by dataset property in JS.

```html
<body>
  <div id="div1" data-info="info">Data Attributes</div>

  <script>
    const element = document.querySelector("div");
    const id = element.id;
    const info = element.dataset.info;
    const dataid = element.dataset.id;

    console.log(`Id: ${id}`);
    console.log(`Data Info: ${info}`);
  </script>
</body>
```



5. HTML Links & Navigation

Q. What are the 5 Types of Links in HTML?

Q. What is the difference between Absolute & Relative URLs?

Q. What is the purpose of the <nav> element in HTML?

Q. What is a Fragment Identifier in a URL?

Q. What is the purpose of the <base> element in HTML.?

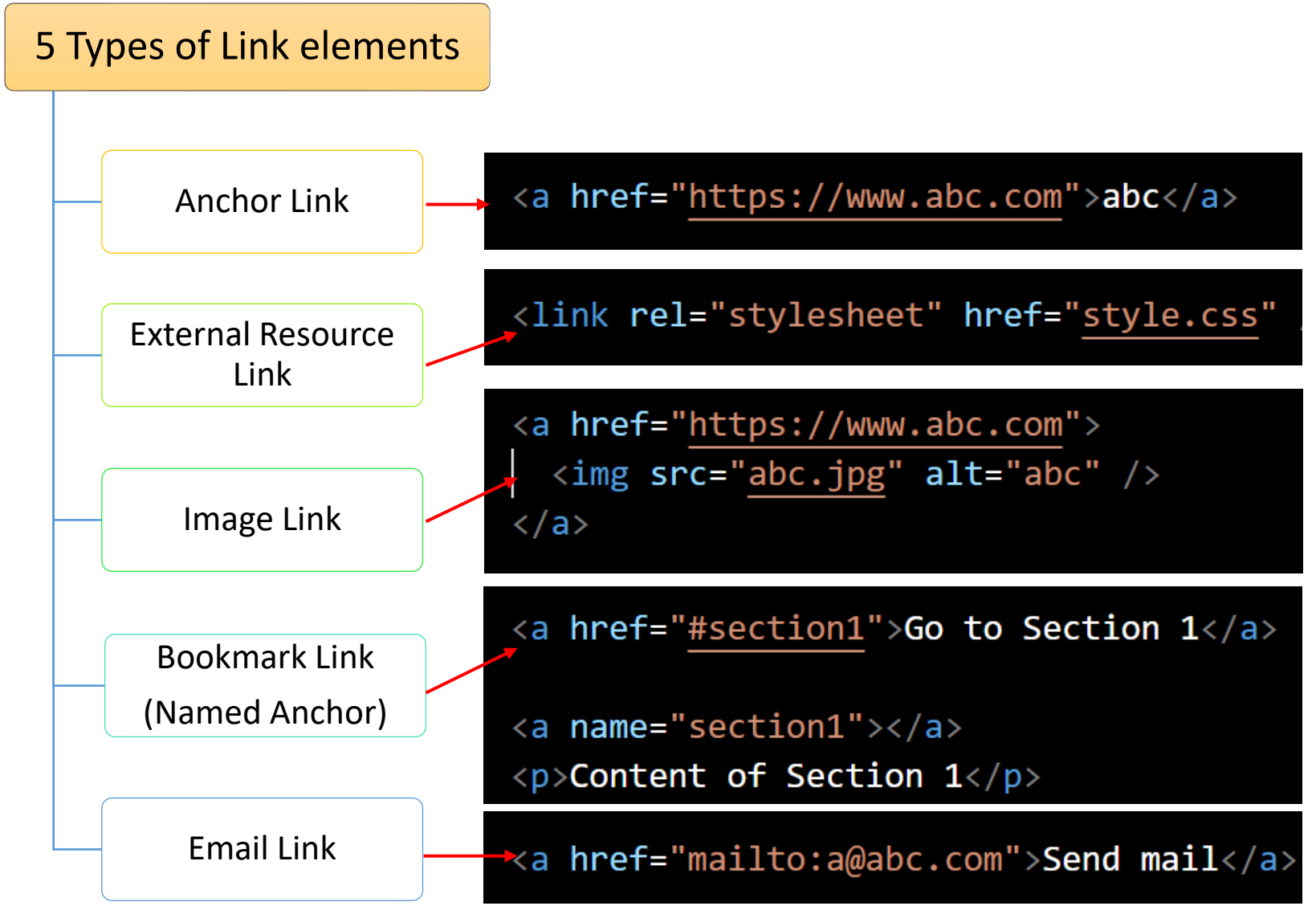
Q. How do you add an external stylesheet in your HTML?

Q. How do you open a link in a new tab?

Q. How do you create an Email Link?

Q. What are the 5 Types of Links in HTML? V. IMP.

1. Anchor Link (<a>) used for navigating from one webpage to another.
2. External Resource Link (<link>) used in the <head> section to connect to an external resource like a stylesheet or an icon.
3. Image Link (<img>) used to create a clickable image that leads to another webpage.
4. Bookmark Link (Named Anchor) points to a specific location within a webpage using a named anchor.
5. Email Link creates a clickable link that opens the user's default email client with a pre-filled email address.



Q. What is the difference between Absolute and Relative URLs? V. IMP.

❖ Absolute URLs:

1. Absolute URLs provide the **complete web address** of a resource.
2. Absolute URLs are typically used to link to resources on **different websites**.

```
<h2>Absolute URLs</h2>
<ul>
  <li><a href="http://www.abc.com">HTTP URL</a></li>
  <li><a href="ftp://ftp.abc.com/doc.pdf">FTP URL</a></li>
  <li><a href="mailto:info@abc.com">Mailto URL</a></li>
</ul>
```

❖ Relative URLs:

1. Relative URLs specify the location of a resource in **relation** to the current document. Full url is not required.
2. They are used when linking to resources within the **same website**.

```
<!-- example/index.html -->
<h2>Relative URLs</h2>
<ul>
  <li><a href="page.html">Same Directory</a></li>
  <li><a href="sub/page.html">Subdirectory</a></li>
  <li><a href="../page.html">Parent Directory</a></li>
  <li><img src="images/image.jpg" alt="Image" /></li>
  <li><link rel="stylesheet" href="styles/styles.css"/></li>
  <li><script src="scripts/script.js"></script></li>
</ul>
```

Q. What is the purpose of the <nav> element in HTML?

- <nav> element in HTML is used to define a section of a web page that contains **navigation links**.
- <nav> element provide clear navigation structure and keep navigation separate from other body content which is good from SEO perspective.

```
<body>
  <header>
    <h1>Website Header</h1>
  </header>
  <nav>
    <ul>
      <li><a href="home.html">Home</a></li>
      <li><a href="about.html">About</a></li>
      <li><a href="services.html">Services</a></li>
      <li><a href="contact.html">Contact</a></li>
    </ul>
  </nav>
</body>
```

- Home
- About
- Services
- Contact

Q. What is a Fragment Identifier in a URL? V. IMP.

- A fragment identifier is used to navigate to a **specific section** of the same webpage.
- Fragment identifier is preceded by a **# (hash)** symbol.

Go to Section 1

Go to Section 2

Section 1

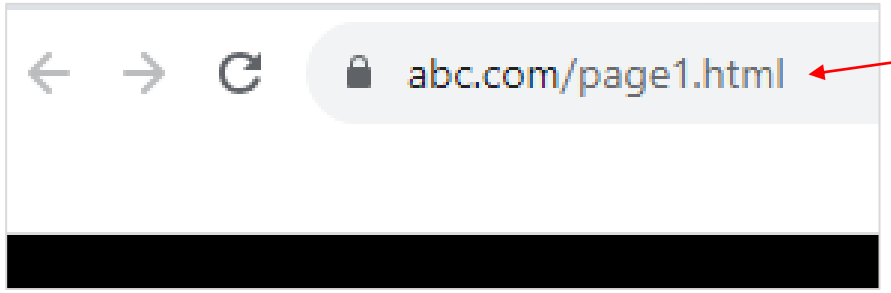
Content for Section 1.

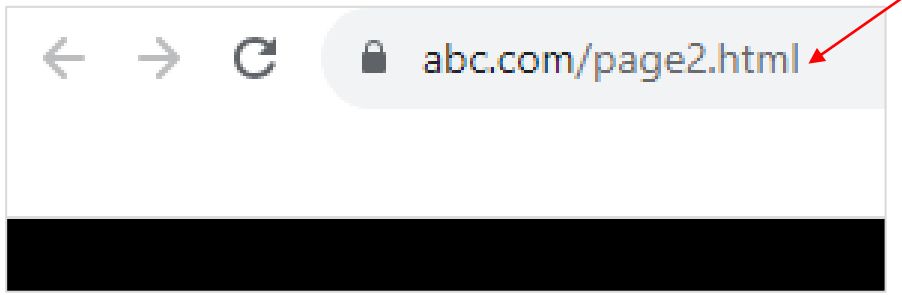
Section 2

```
<body>
  <a href="#section1">Go to Section 1</a>
  <a href="#section2">Go to Section 2</a>

  <section id="section1">
    <h2>Section 1</h2>
    <p>Content for Section 1.</p>
    <div style="height: 500px">Test</div>
  </section>

  <section id="section2">
    <h2>Section 2</h2>
    <p>Content for Section 2.</p>
  </section>
</body>
```

Q. What is the purpose of the <base> element in HTML.?

- The <base> element in HTML is used to specify a **base URL** for relative URLs within a document.
- The <base> element is typically placed within the **<head>** section.






```html
<html lang="en">
  <head>
    <title>Base Element Example</title>
    <base href="https://www.abc.com/" />
  </head>
  <body>
    <a href="page1.html">Link to Page 1</a>
    <a href="page2.html">Link to Page 2</a>
  </body>
</html>
```

Q. How do you add an external stylesheet in your HTML?

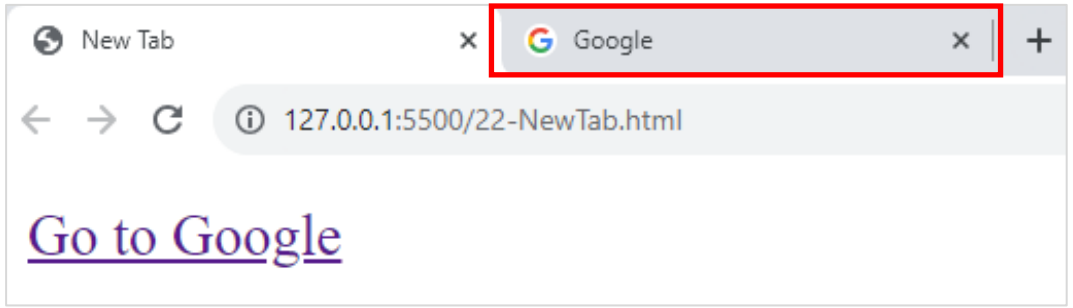
- **<link> element** is an empty element used to link **external resources** like stylesheets to the HTML document.
- **rel** attribute specifies the **relationship** between the current document and the linked resource.
- **href** attribute specifies the **location (URL)** of the external stylesheet.

```
<html>
  <head>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <!-- Body content -->
  </body>
</html>
```

Q. How do you open a link in a new tab?

- Target attribute specifies how the linked content should be displayed.
- Target values: _blank(new tab), _self(default), _parent, _top

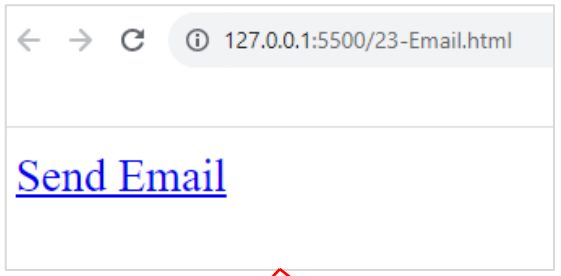
```html
<html>
  <head>
    <title>New Tab</title>
  </head>
  <body>
    <a href="https://www.google.com" target="_blank">Go to Google</a>
  </body>
</html>
```

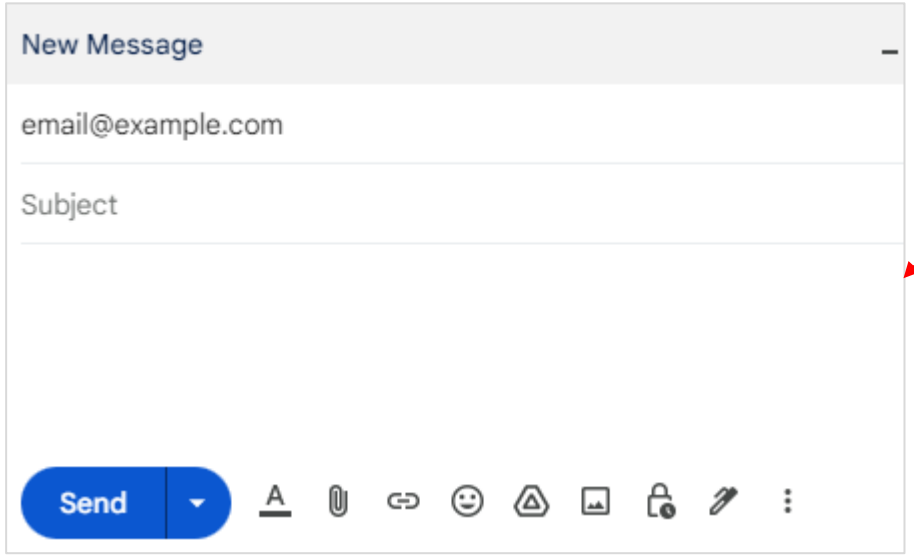


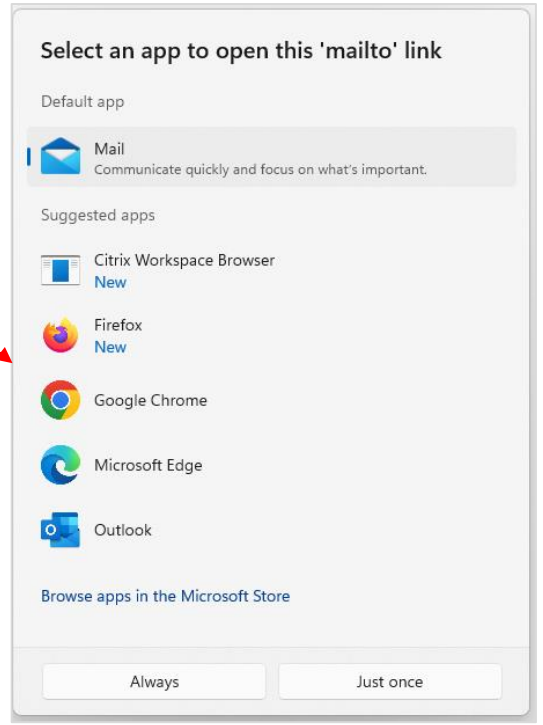
Q. How do you create an Email Link?

- Set the href attribute "mailto:" followed by the email address.
- This will prompt the user's default email with the specified email address pre-filled in the "To" field. Else it will prompt for the other options.

```
<body>
  <a href="mailto:email@example.com">Send Email</a>
</body>
```







6. HTML Lists & Tables

Q. What are the different Types of Lists in HTML?

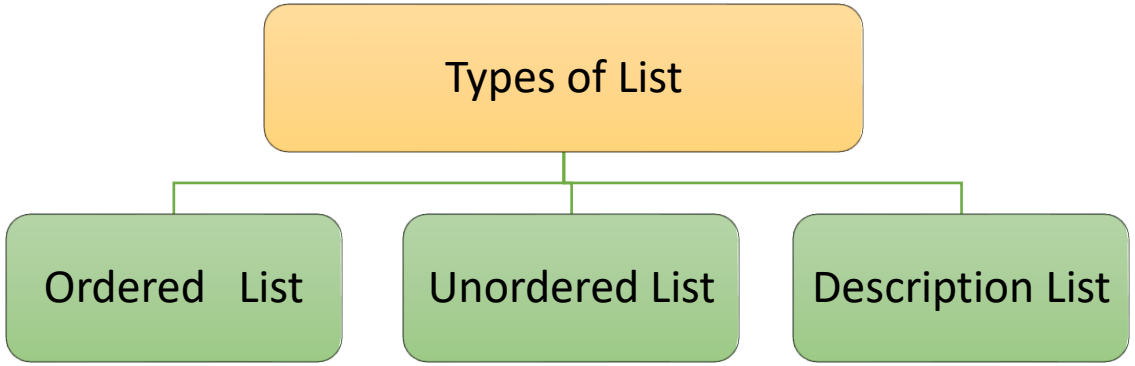
Q. What is a Nested List in HTML?

Q. What are table, tr, th, td elements? What are table advantages & disadvantages?

Q. What is the colspan attribute in HTML?

Q. What is the best way to add a border to a table, column and cell?

Q. What are the different Types of Lists in HTML? V. IMP.



1. An ordered list is a list where the items are **numbered or ordered** in a specific sequence.
2. An unordered list is a list where the items are **marked with bullets**.
3. A description list is used to display a list of terms along with their **corresponding descriptions** or definitions.

Ordered List:

1. Item 1
2. Item 2

```
<h2>Ordered List:</h2>
<ol>
  <li>Item 1</li>
  <li>Item 2</li>
</ol>
```

Unordered List:

- Item A
- Item B

```
<h2>Unordered List:</h2>
<ul>
  <li>Item A</li>
  <li>Item B</li>
</ul>
```

Description List:

Term 1
Description 1
Term 2
Description 2

```
<h2>Description List:</h2>
<dl>
  <dt>Term 1</dt>
  <dd>Description 1</dd>
  <dt>Term 2</dt>
  <dd>Description 2</dd>
</dl>
```

Q. What is a Nested List in HTML?

- A nested list in HTML is a list that is placed within another list item.

Technologies

- Backend
  - Java
  - .NET
  - Node
- Frontend
  - HTML
  - JS
  - React

```
<h1>Technologies</h1>
<ul>
  <li>
    Backend
    <ul>
      <li>Java</li>
      <li>.NET</li>
      <li>Node</li>
    </ul>
  </li>
  <li>
    Frontend
    <ul>
      <li>HTML</li>
      <li>JS</li>
      <li>React</li>
    </ul>
  </li>
</ul>
```

Q. What are table, tr, th, td elements? What are table advantages & disadvantages? V. IMP.

❖ Table elements:

1. <table> is the container for the entire table.
2. <tr>(table row) is used to define a row in the table.
3. <th>(table header) is used to represent the column headers.
4. <td>(table data) is used to represent the regular cells in a table.

Tables are a powerful tool for styling and displaying structured data.

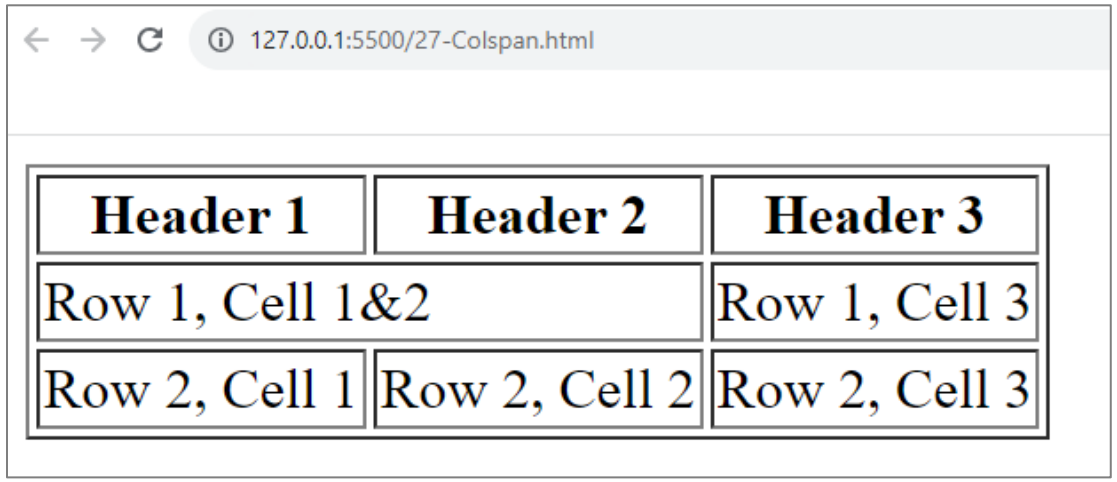
Tables multiple column's structure is not good for mobile devices(not responsive).

Header 1	Header 2
Row 1, Cell 1	Row 1, Cell 2
Row 2, Cell 1	Row 2, Cell 2

```
<body>
    <table border="1">
        <tr>
            <th>Header 1</th>
            <th>Header 2</th>
        </tr>
        <tr>
            <td>Row 1, Cell 1</td>
            <td>Row 1, Cell 2</td>
        </tr>
        <tr>
            <td>Row 2, Cell 1</td>
            <td>Row 2, Cell 2</td>
        </tr>

    </table>
</body>
```

Q. What is the colspan attribute in HTML?

- The colspan attribute is used to **merge multiple cells** horizontally into a single cell.
- colspan attribute is applicable to **<th>** and **<td>** only.




```
<body>
  <table border="1">
    <tr>
      <th>Header 1</th>
      <th>Header 2</th>
      <th>Header 3</th>
    </tr>
    <tr>
      <td colspan="2">Row 1, Cell 1&2</td>
      <td>Row 1, Cell 3</td>
    </tr>
    <tr>
      <td>Row 2, Cell 1</td>
      <td>Row 2, Cell 2</td>
      <td>Row 2, Cell 3</td>
    </tr>
  </table>
</body>
```

Q. What is the best way to add a border to a table, column and cell?

- Setting the common style for table, tr, th and td, then all the multiple tables in your webpage will follow consistent same style and format.

Header 1	Header 2
Row 1, Cell 1	Row 1, Cell 2
Row 2, Cell 1	Row 2, Cell 2

```
<head>
  <title>Styled Table</title>
  <style>
    table {
      border: 2px solid #000;
      width: 25%;
    }

    th,
    td {
      border: 1px solid #000;
      padding: 8px;
      text-align: center;
    }
  </style>
</head>
```

```
<body>
  <table>
    <tr>
      <th>Header 1</th>
      <th>Header 2</th>
    </tr>
    <tr>
      <td>Row 1, Cell 1</td>
      <td>Row 1, Cell 2</td>
    </tr>
    <tr>
      <td>Row 2, Cell 1</td>
      <td>Row 2, Cell 2</td>
    </tr>
  </table>
</body>
```

7. HTML Images & Multimedia

Q. How to embed an Image in HTML? Why alt attribute is so important?

Q. What are the difference between JPEG & PNG image formats?

Q. How to embed Video element in HTML? Why we need multiple source elements?

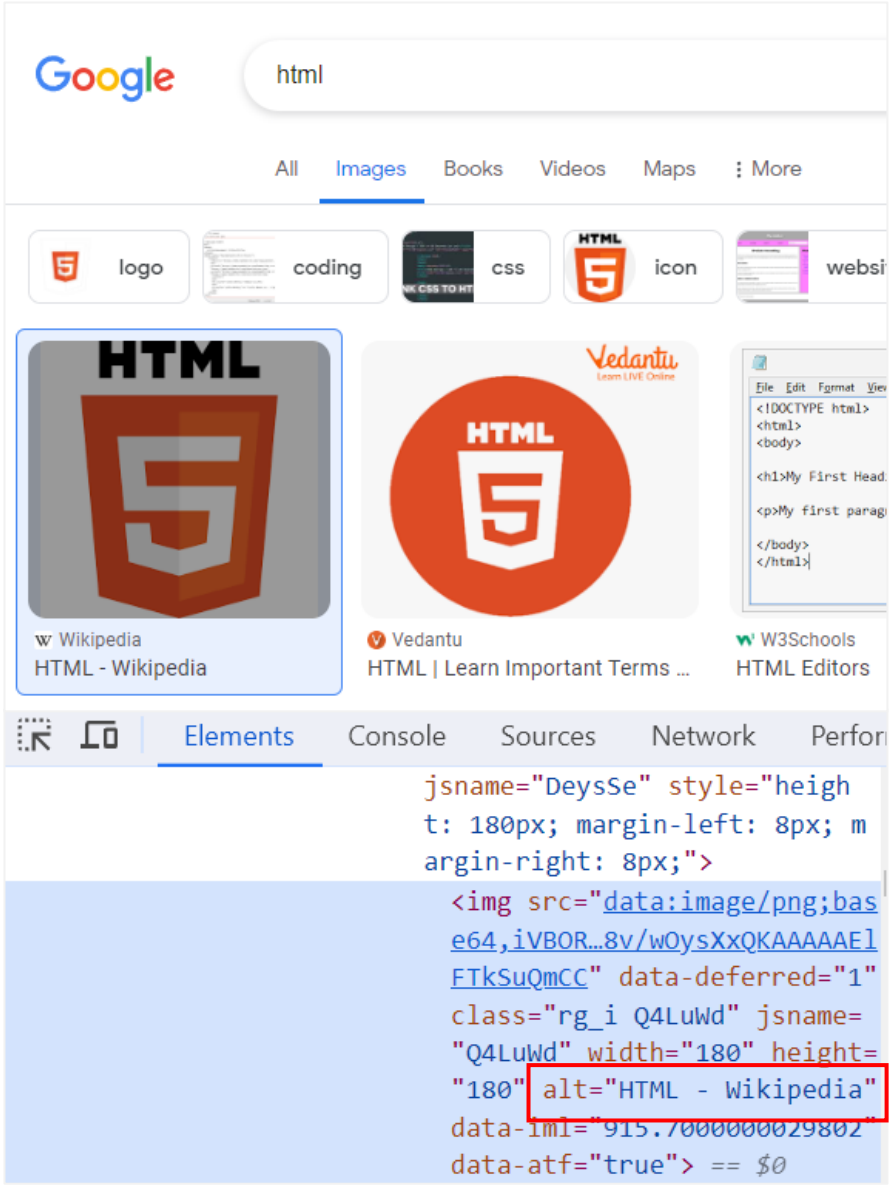
Q. How to embed Audio element in HTML?

Q. What are the Autoplay & Loop attributes in multimedia elements?

Q. How to embed an Image in HTML? Why alt attribute is so important? V. IMP.

- <img> element is used to embed an image in HTML.
- The src(soruce) attribute specifies the source file of the image.
- The alt(alternative text) attribute provides alternative text that describes the content of the image which is important for SEO(for searching).

```
<html lang="en">
  <head>
    <title>Image Example</title>
  </head>
  <body>
    <img src="example.jpg" alt="Image" />
  </body>
</html>
```



Q. What are the difference between JPEG and PNG image formats?

JPEG	PNG
1. JPEG stands for Joint Photographic Experts Group.	PNG stands for Portable Network Graphics
2. JPEG uses **lossy compression**, which means some data is discarded to reduce image size.	PNG uses **lossless compression**, meaning no data is lost during compression.
3. Due to lossy compression, lower image **quality**.	Due to lossless compression, higher image quality.
4. Due to lossy compression, Image **size** is reduced.	Due to lossless compression, the image has a comparatively larger size.

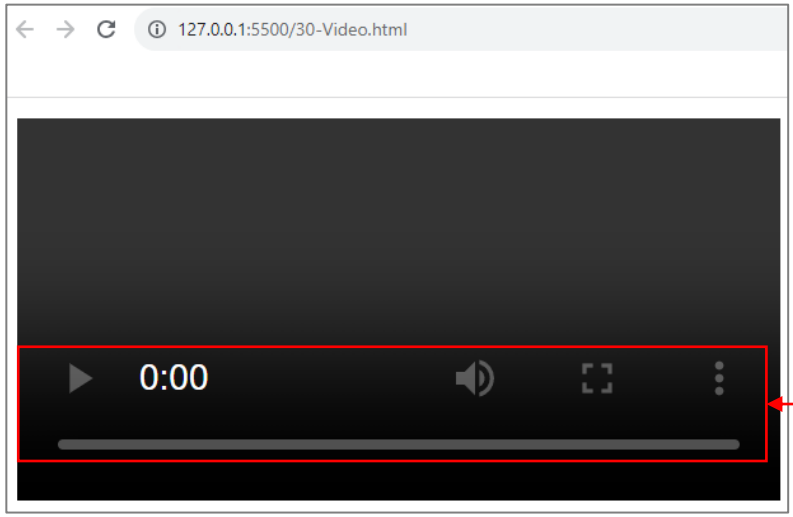
Q. What are the difference between JPEG and PNG image formats?

JPEG	PNG
1. JPEG stands for Joint Photographic Experts Group.	PNG stands for Portable Network Graphics
2. JPEG uses **lossy compression**, which means some data is discarded to reduce image size.	PNG uses **lossless compression**, meaning no data is lost during compression.
3. Due to lossy compression, lower image **quality**.	Due to lossless compression, higher image quality.
4. Due to lossy compression, Image **size** is reduced.	Due to lossless compression, the image has a comparatively larger size.

Q. How to embed Video in HTML? Why we need multiple source elements?

1. **<video>** is the element used to embed videos.
2. **controls** attribute adds video controls like play, pause & volume.
3. **<source>** element specifies multiple sources for the video. If one video format is not compatible with browser, then other will execute.
4. **src** attribute specifies the path to the video file.
5. **type** attribute indicates the MIME type.
6. The **text** "Your browser..." is a fallback message for browsers that do not support the <video> element.

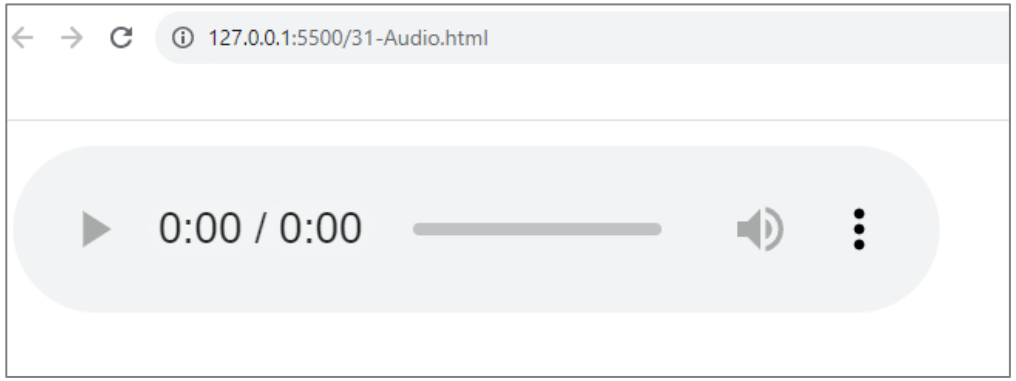
```
<body>
  <video controls>
    <source src="filepath.mp4" type="video/mp4" />
    <source src="filepath.webm" type="video/webm" />
    Your browser does not support the video tag.
  </video>
</body>
```



Q. How to embed Audio in HTML?

1. **<audio>** is the element used to embed audios.
2. **controls** attribute adds video controls like play, pause & volume.
3. **<source>** element specifies multiple sources for the audio. If one audio format is not compatible with browser, then other will execute.
4. **src** attribute specifies the path to the audio file.
5. **type** attribute indicates the MIME type.
6. The **text** "Your browser..." is a fallback message for browsers that do not support the <audio> element.

```
<body>
  <audio controls>
    <source src="filepath.mp3" type="audio/mp3" />
    <source src="filepath.ogg" type="audio/ogg" />
    Your browser does not support the audio element.
  </audio>
</body>
```



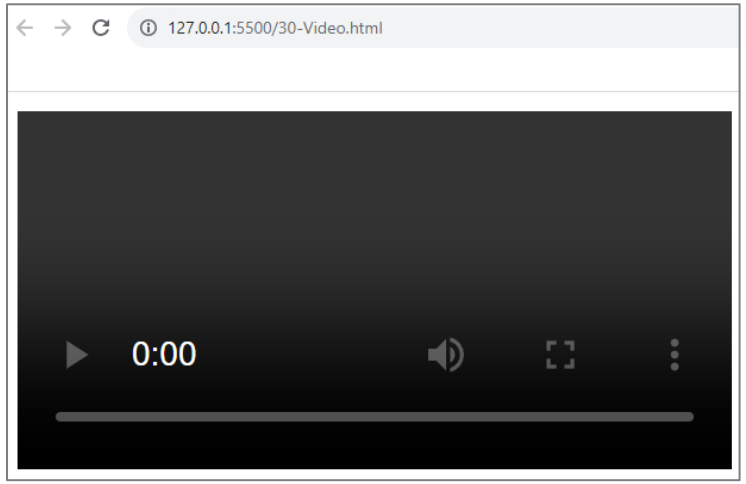
Q. What are the Autoplay & Loop attributes in multimedia elements?

- The autoplay attribute allows media content to **start playing automatically** as soon as the page is loaded.

```
<h2>Autoplay</h2>
<video src="video.mp4" autoplay controls>
  Your browser does not support the video element.
</video>
```

- The loop attribute, allows the media content to **restart** whenever it reaches the end.

```
<h2>Loop</h2>
<video src="video.mp4" loop controls>
  Your browser does not support the video element.
</video>
```



8. HTML Forms

Q. What are HTML Forms and what are its advantages?

Q. Q. What are HTML Form Elements and their Main Attributes?

Q. How to Validate for empty input field in form?

Q. What is the role of the <textarea> element in a form?

Q. What are the Types of Input elements in a form?

Q. How to create radio buttons and checkboxes? How to group them?

Q. What is the purpose of the <select> element in a form?

Q. What are HTML Forms and what are its advantages? V. IMP.

- Forms provide a way to collect and **submit** user data on a webpage.

Advantages of Form element

1. Data collection & organization is simple
2. Data validation is simple
3. Data submission on server is easy
4. Styling and Layout
5. Compatible with all browsers

Login Form

Username: happy

Password: ••••••••

Submit

Q. What are HTML Form Elements and their Main Attributes? V. IMP.

- Form elements & Attributes
  - 1. form element
  - 2. action attribute
  - 3. method attribute
  - 4. label element
  - 5. for attribute
  - 6. input element
  - 7. type attribute
  - 8. name attribute

```html
<form action="/submit" method="post">
  <label for="username">Username:</label>
  <input type="text" id="username" name="username" />
  <br /><br />
  <label for="password">Password:</label>
  <input type="password" id="pwd" name="password" />
  <br /><br />
  <input type="submit" value="Submit" />
</form>
```

Login Form

Username: happy

Password: ••••••••

Submit

Q. What are HTML Form Elements and their Main Attributes? V. IMP.

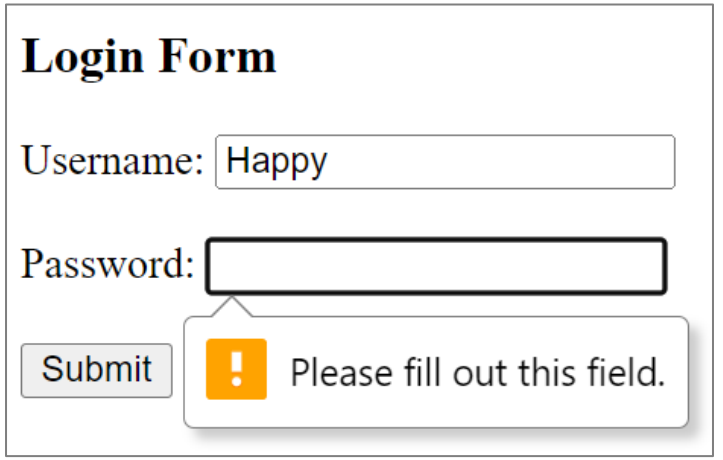
❖ **Form elements and attributes**

1. **<form>** element is the container of form elements.
2. The **action** attribute specifies the **URL** where the form data will be sent when it is submitted.
3. The **method** attribute defines the **HTTP method** to be used when sending the form data.
4. The **<label>** element is used to provide a label or **description** for an input field. The **for** attribute **associate** the label with a specific input field.
5. The **<input>** element is used to **create controls**.
6. The **type** attribute specifies the type of input field.
7. The **name** attribute is used to **define a name** for the input field.

Q. How to Validate for empty input field in form?

- **required** attribute is used to make the input filed mandatory.

```
<body>
  <h3>Login Form</h3>
  <form action="/submit" method="post">
    <label for="username">Username:</label>
    <input type="text" id="username" name="username" required />
    <br /><br />
    <label for="password">Password:</label>
    <input type="password" id="password" name="password" required />
    <br /><br />
    <input type="submit" value="Submit" />
  </form>
</body>
```



Q. What is the role of the <textarea> element in a form?

- The <textarea> element is used in HTML forms to create a **multi-line text** input field.
- rows attribute in textarea identify the **number of lines** in the text area.
- cols attribute in textarea identify the **number of characters** in one row.

```
<body>
  <form>
    <label for="comments">Comments:</label><br />
    <textarea id="comments" name="comments"
    rows="4" cols="50"></textarea>
    <br />
    <input type="submit" value="Submit" />
  </form>
</body>
```

Comments:

```
12345678910123456789101234567891012345678910123456
2
3
4
```

Submit

Q. What are the Types of Input elements in a form?

Types of Input elements

- Text Input
- Password Input
- File Input
- Submit Button
- Reset Button
- Checkbox
- Radio Button
- Select

```
<input type="text" id="uid" name="uid" />
<br /><br /><br />
```

Happy

```
<input type="password" id="pwd" name="pwd" />
<br /><br /><br />
```

••••••••••

```
<input type="file" id="file" name="file" />
<br /><br /><br />
```

Choose File No file chosen

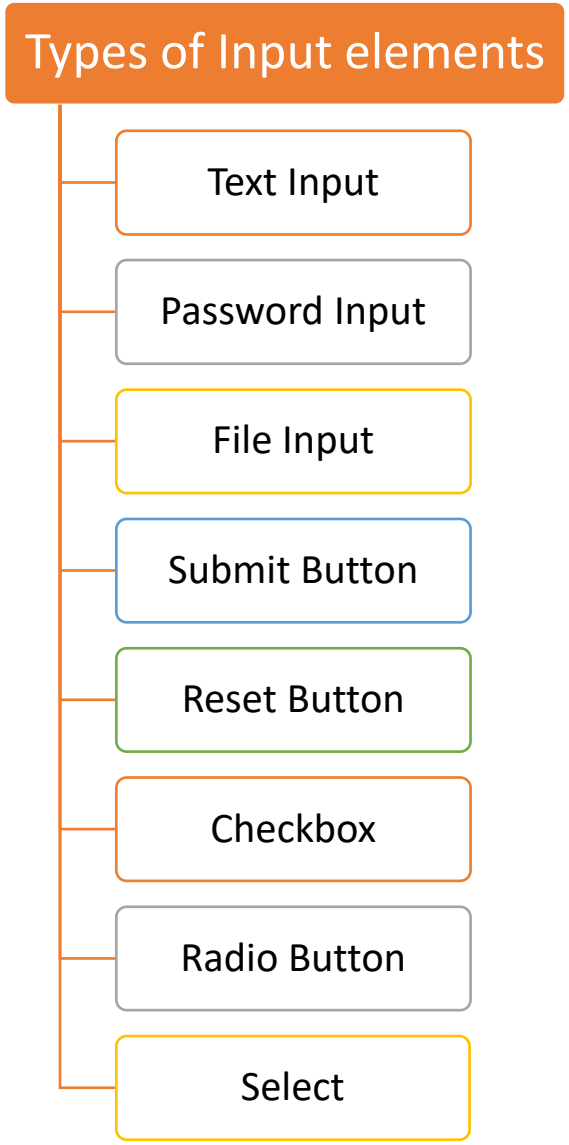
```
<input type="submit" value="Submit" />
<br /><br /><br />
```

Submit

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

Reset

Q. What are the Types of Input elements in a form?



1. **Text Input:** Allows users to type alphanumeric characters. Commonly used for names, email addresses, passwords, etc.

2. **Password Input:** Masks the characters entered by the user, typically used for sensitive information like passwords.

3. **File Input:** Allows users to upload files.

4. **Submit Button:** Creates a button that submits the form data to a server for processing.

5. **Reset Button:** Creates a button that resets all the form fields back to their default values.

Q. How to create radio buttons and checkboxes? How to group them?

- Radio buttons are used when you want the user to **select exactly one option** from a group of options.

```
<label> <input type="radio" name="gender" value="male" /> Male </label><br />
<label> <input type="radio" name="gender" value="female" /> Female </label><br />
<label> <input type="radio" name="gender" value="other" /> Other </label><br />
```

- Male
- Female
- Other

- Checkboxes are used when you want the user to be able to **select multiple options** from a list.

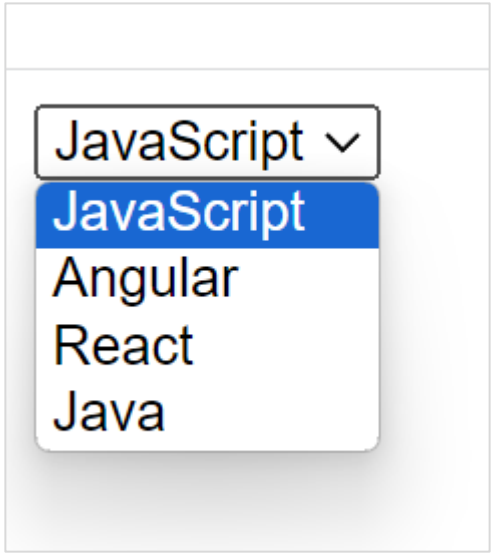
```
<label> <input type="checkbox" name="hobby" value="music" /> Music </label><br />
<label> <input type="checkbox" name="hobby" value="sports" /> Sports </label><br />
<label> <input type="checkbox" name="hobby" value="travel" /> Travel </label
```

- [x] Music
- [x] Sports
- [ ] Travel

- All radio buttons or checkboxes within a group should share the **same name** attribute.

Q. What is the purpose of the <select> element in a form?

- The <select> element in HTML is used to create a **dropdown menu** within a form.
- Dropdown menus are space-efficient compared to radio-buttons and checkboxes for displaying a **long list of options**.




```
<body>
  <form>
    <select name="Technology">
      <option value="1">JavaScript</option>
      <option value="2">Angular</option>
      <option value="3">React</option>
      <option value="4">Java</option>
    </select>
  </form>
</body>
```

9. Best Practices, Boilerplate & Deployment

Q. What are Top 10 Best Practices for writing HTML code?

Q. What is HTML5 Boilerplate code?

Q. How do you optimize an HTML website for Performance?

Q. What is SEO? What are 5 HTML Best Practices for SEO?

Q. What are the 5 steps required for HTML website deployment?

Q. What are top 10 Best Practices for writing HTML code? V. IMP.

❖ HTML Bad Practices

```
<!-- 1. Missing DOCTYPE declaration -->
<html>
  <head>
    <!-- 2. Missing charset meta tag -->
    <title>Interview Happy</title>
  </head>
  <body>
    <!-- 3. Missing semantic header tag -->
    <div id="Header">
      <h1>Interview Happy</h1>
    </div>
    <!-- 4. Meaningless ID's and classes -->
    <div id="xyz" class="abc">
      <!-- 5. Incorrect capitalization of tags -->
      <P>Help in cracking interviews</P>
      <!-- 6. Missing Alt Text for Images -->
      <img src="example.jpg" />
    </div>
  </body>
</html>
```

❖ HTML Best Practices

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
    content="width=device-width, initial-scale=1.0" />
    <title>Interview Happy</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <header>
      <h1>Interview Happy</h1>
    </header>
    <main>
      <section id="content">
        <p>Help in cracking interviews</p>
        <img src="example.jpg" alt="interview example" />
      </section>
    </main>
  </body>
</html>
```

Q. What are top 10 Best Practices for writing HTML code? V. IMP.

❖ **7 Best practices for HTML:**

1. Proper DocType declaration
2. Proper Meta Tags
3. Use Semantic Elements.
4. Give meaningful ID's and class names.
5. Give Lowercase Element Names.
6. Alternative Text for Images by alt attribute.
7. Use of proper Header Elements.(h1,h2,h3,h4,h5,h6)
8. Use External CSS Link for styling.
9. Proper Indentation and Formatting for readability.
10. Always give title for the website page.

❖ HTML with best practices

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
    content="width=device-width, initial-scale=1.0" />
    <title>Interview Happy</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <header>
      <h1>Interview Happy</h1>
    </header>
    <main>
      <section id="content">
        <p>Help in cracking interviews</p>
        <img src="example.jpg" alt="interview example" />
      </section>
    </main>
  </body>
</html>
```

Q. What is an HTML5 Boilerplate?

- An HTML5 boilerplate is a **basic template** for building modern websites using HTML5.

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>HTML 5 Boilerplate</title>
    <link rel="stylesheet" href="style.css" />
  </head>
  <body>
    <!-- Website Content -->
    <script src="index.js"></script>
  </body>
</html>
```

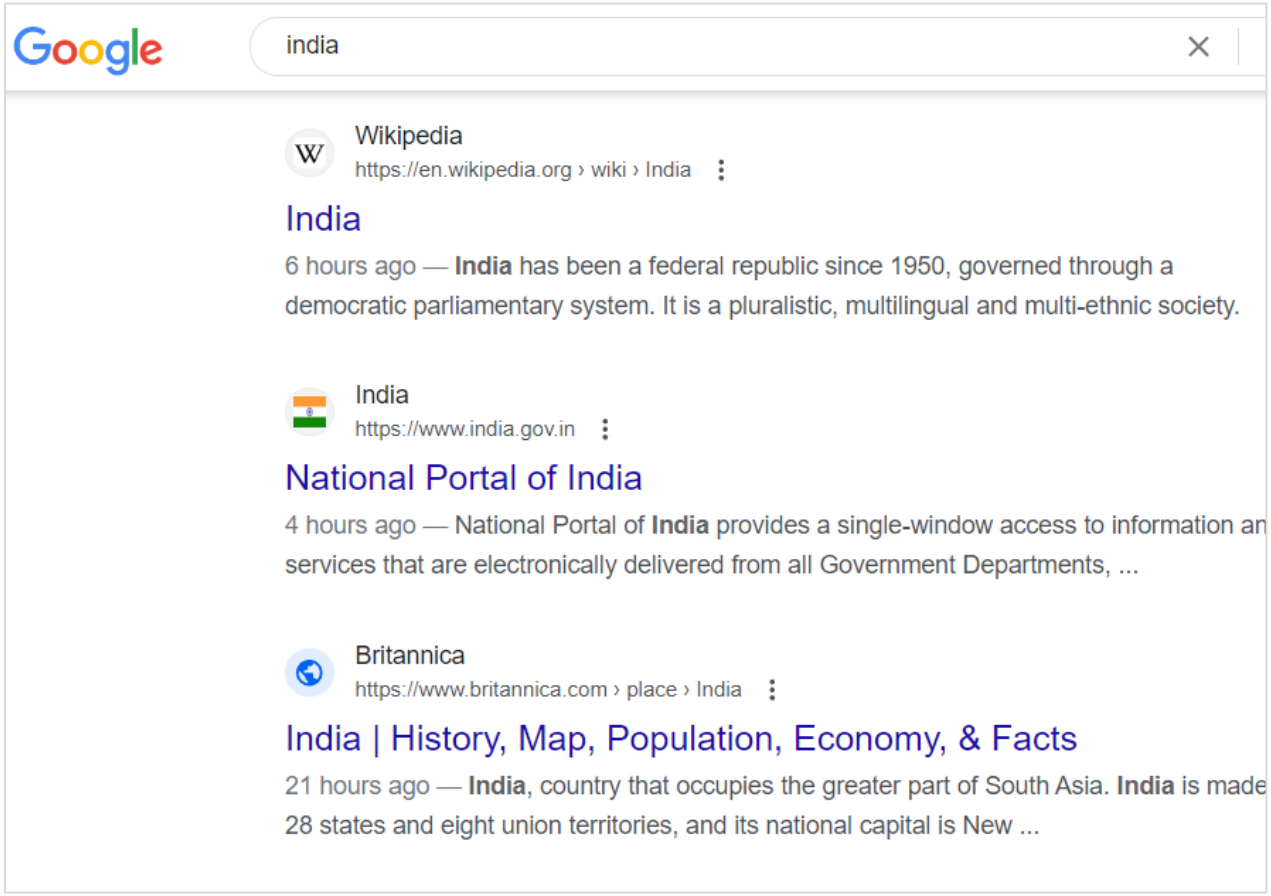
Q. How do you optimize an HTML website for Performance?

❖ 8 Best Practices for performance improvement in HTML:

1. Minimize HTTP Requests: Reduce the number of HTTP requests by combining CSS and JavaScript files.
2. Use External CSS and JavaScript Files
3. Minify CSS, JavaScript, and HTML: Remove unnecessary whitespace, comments & formatting to reduce file sizes.
4. Compress Images: Use tools to compress images and choose appropriate formats.
5. Leverage Browser Caching: Set cache headers to allow resources to be stored locally, reducing the need for repeated requests.
6. Load Critical Resources First.
7. Use Async and Await in JS.
8. Implement Lazy Loading to improve initial load times.

Q. What is SEO? What are 5 HTML Best Practices for SEO?

- SEO(Search Engine Optimization) is the practice of optimizing a website to improve its **visibility** and **ranking** in search engine results.



5 HTML Best Practices for SEO

1. Use Semantic HTML Elements
2. Optimize Page Titles
3. Use Meta Description Tags
4. Use Proper Heading Tags
5. Optimize Image Alt Attributes

Q. What are the 5 steps required for HTML website deployment?

- 5 Basic Steps for HTML website deployment:

1. Choose a Hosting Provider (ex: aws/ azure)
2. Purchase a Domain (ex: www.interviewhappy.com)
3. Prepare Your Website Files(ex: HTML/ CSS/ JS/ Image)
4. Upload Files to Hosting Server(using FTP, hosting panel, GIT)
5. Configure SSL (Optional - Enable SSL for using HTTPS and enhance security)

10. HTML Responsiveness, Frameworks & Others

Q. What is Responsive Design?

Q. What are the top 3 things required to make a website Responsive in HTML?

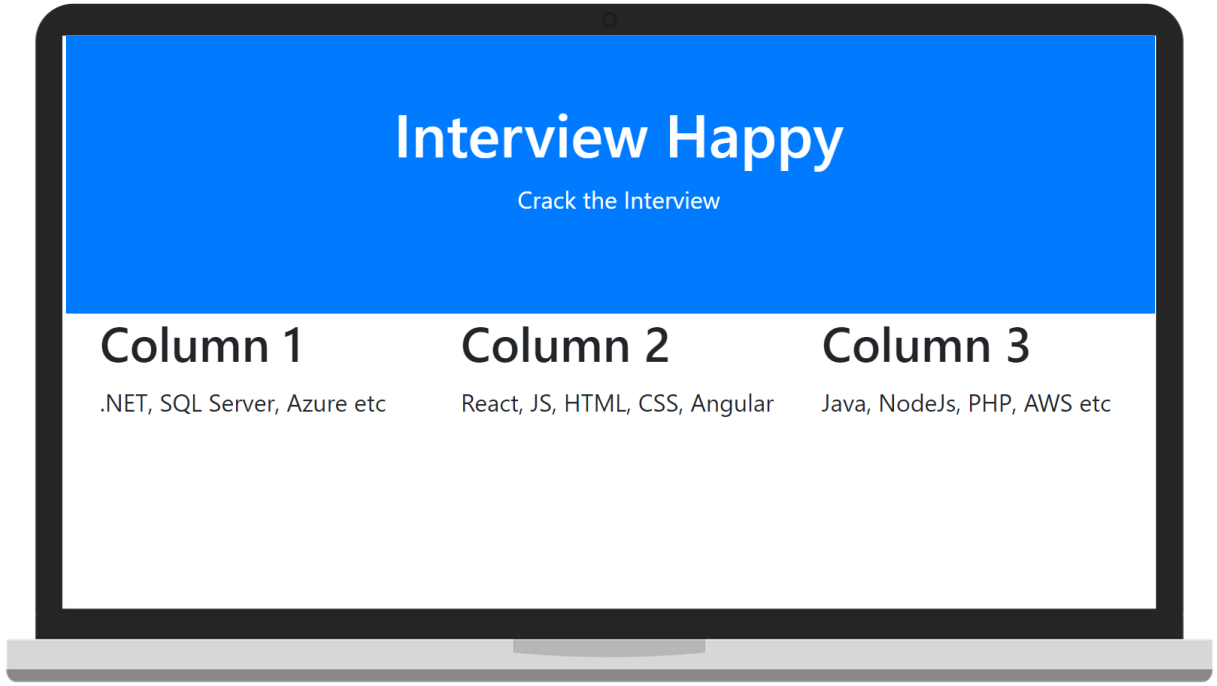
Q. What are some Responsive Design Frameworks for HTML & CSS?

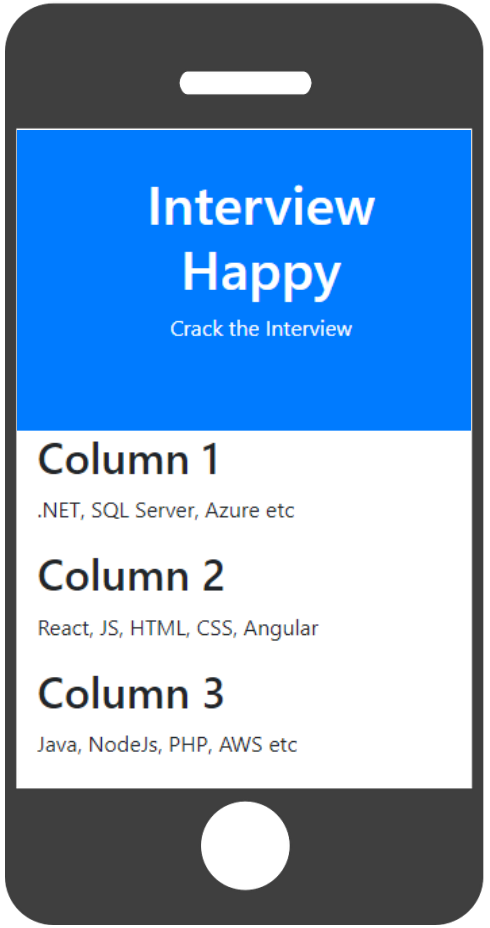
Q. What are HTML Entities? How do you display special characters in HTML?

Q. What are the Top 5 Emerging Trends in HTML development?

Q. What is Responsive Design? V. IMP.

- Responsive design is a practice of creating web pages that **adapt** and display well on various devices and **screen sizes**.





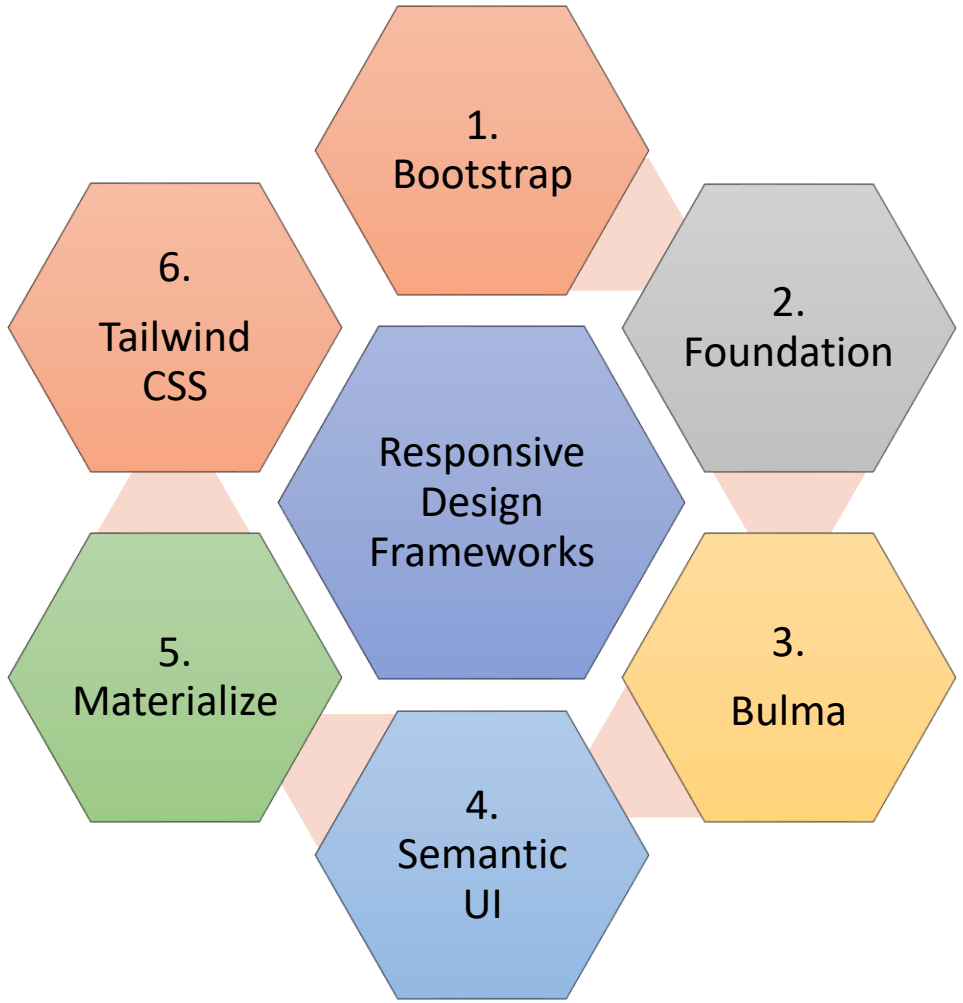
Q. What are the top 3 things required to make a website Responsive in HTML?

1. **Set the Viewport Meta Tag** : This tag tells the browser to adjust the width of the page to match the screen's width.
2. **Use Relative Units**: Instead of using fixed units like pixels, use relative units like percentages
3. **Media Queries**: They set different sizes for different screen types.

```css
.box {
  width: 30%;
}

@media (max-width: 768px) {
  .box {
    width: 45%;
  }
}
```

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport"
    content="width=device-width, initial-scale=1.0" />
    <title>Responsive HTML Example</title>
    <link rel="stylesheet" href="39-MakeResponsive.css"
  </head>
  <body>
    <div class="container">
      <div class="box" width="33%">Box 1</div>
      <div class="box" width="33%">Box 2</div>
      <div class="box" width="33%">Box 3</div>
    </div>
  </body>
</html>
```

Q. What are some Responsive Design Frameworks for HTML & CSS?



Q. What are HTML Entities? How do you display special characters in HTML?

- HTML entities are special codes or sequences of characters used to represent reserved or special characters in HTML.

Top 10 Special Characters in HTML

Ampersand: &

Lesser than: <

Greater than: >

Euro symbol: €

Copyright symbol: ©

Trademark symbol: ™

Non-breaking space:

Double quotation mark: "

Single quotation mark: '

Registered trademark symbol: ®

```
<body>
  <h5>Top 10 Special Characters in HTML</h5>
  <p>Ampersand: &amp;</p>
  <p>Lesser than: &lt;</p>
  <p>Greater than: &gt;</p>
  <p>Euro symbol: &euro;</p>
  <p>Copyright symbol: &copy;</p>
  <p>Trademark symbol: &trade;</p>
  <p>Non-breaking space:  </p>
  <p>Double quotation mark: &quot;</p>
  <p>Single quotation mark: &apos;</p>
  <p>Registered trademark symbol: &reg;</p>
</body>
```

Q. What are the Top 5 Emerging Trends in HTML development?

1. Web Components:

Web Components are a set of web platform APIs that allow you to create custom, reusable HTML elements.

2. Progressive Web Apps (PWAs)

PWAs are web applications that provide a native app-like experience to users.

3. Single Page Applications (SPAs):

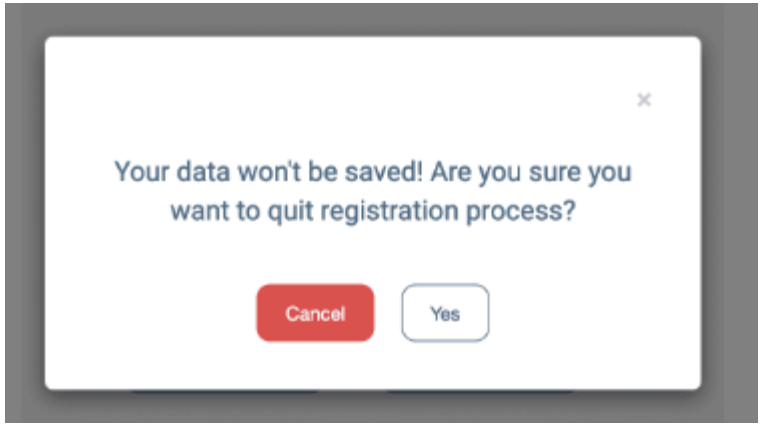
SPAs load a single HTML page and dynamically update content as the user interacts with the app.

4. 3D Graphics and Virtual Reality (VR)

VR refers to the integration of virtual reality experiences within a web browser using HTML, CSS, and JS. Example: WebGL & WebVR.

5. Augmented Reality (AR):

AR combines the physical world with digital information or content. Example: ARCore and ARKit for web development.



1. CSS Basics & Implementation

Q. What is CSS? What are the 5 ways to Implement CSS in HTML?

Q. What is Inline Style in CSS? When to use it in real applications?

Q. What is Internal Stylesheet in CSS? When to use it in real applications?

Q. What is External Stylesheet in CSS? When to use it in real applications?

Q. What are the 5 advantages of external stylesheet?

Q. How do you Include CSS in a webpage or HTML?

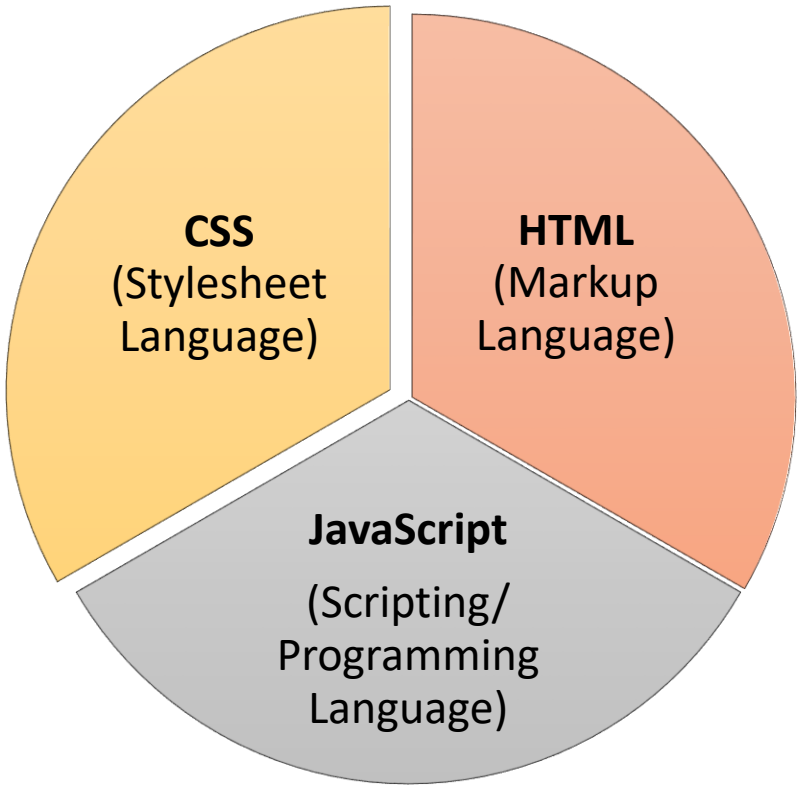
Q. How to implement CSS using @import rule?

Q. What is CSS Preprocessors? What is SCSS?

Q. What are the 3 Types of CSS Preprocessors?

Q. What is CSS? What are the 5 ways to Implement CSS in HTML? V. IMP.

- CSS(Cascading Style Sheets) is a **stylesheet language** used to control the **presentation** of web pages.



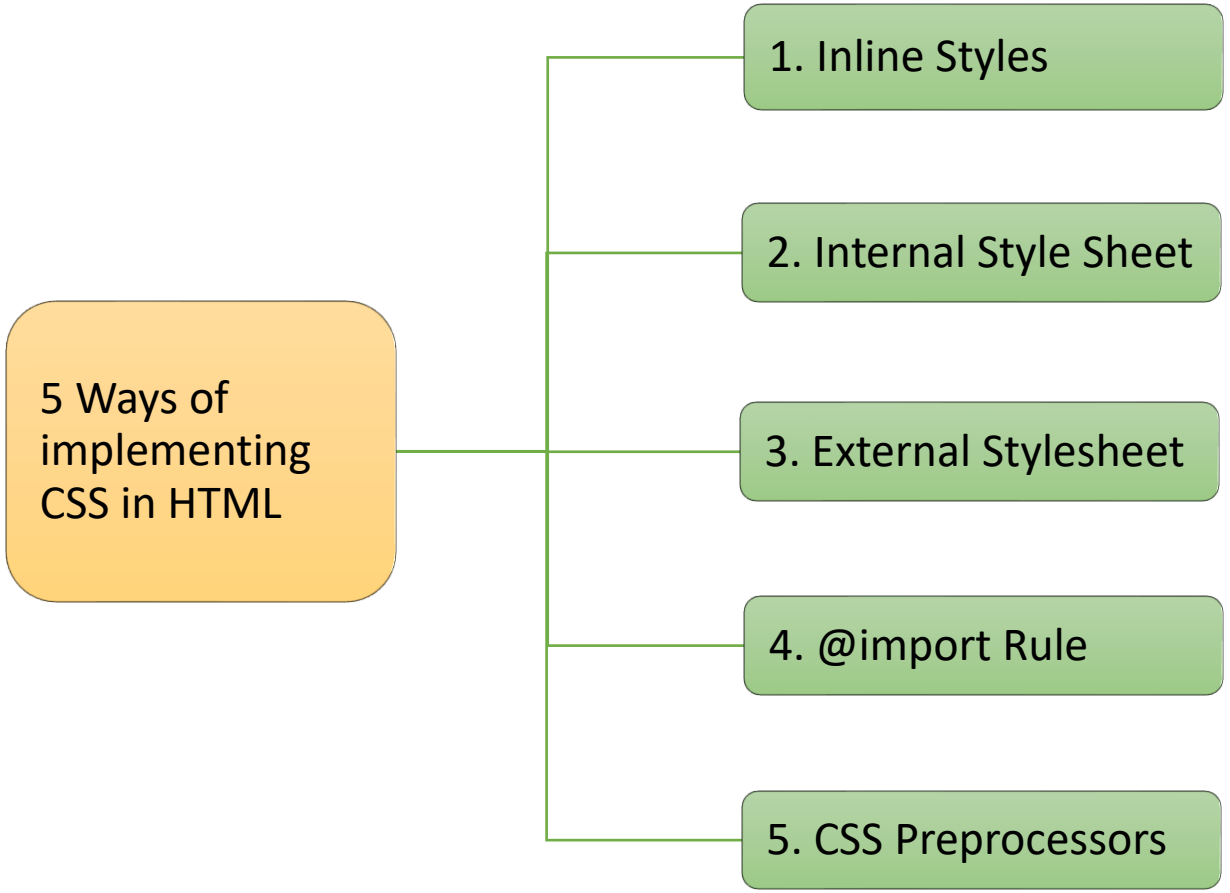

```
<html>
  <head>
    <meta charset="UTF-8" />
    <title>Inline CSS Example</title>
  </head>
  <body>
    <div style="background-color: #f0f0f0;">
      <h1 style="color: blue">Interview</h1>
      <p style="font-size: 30px">Happy</p>
    </div>
  </body>
</html>
```

Interview

Happy

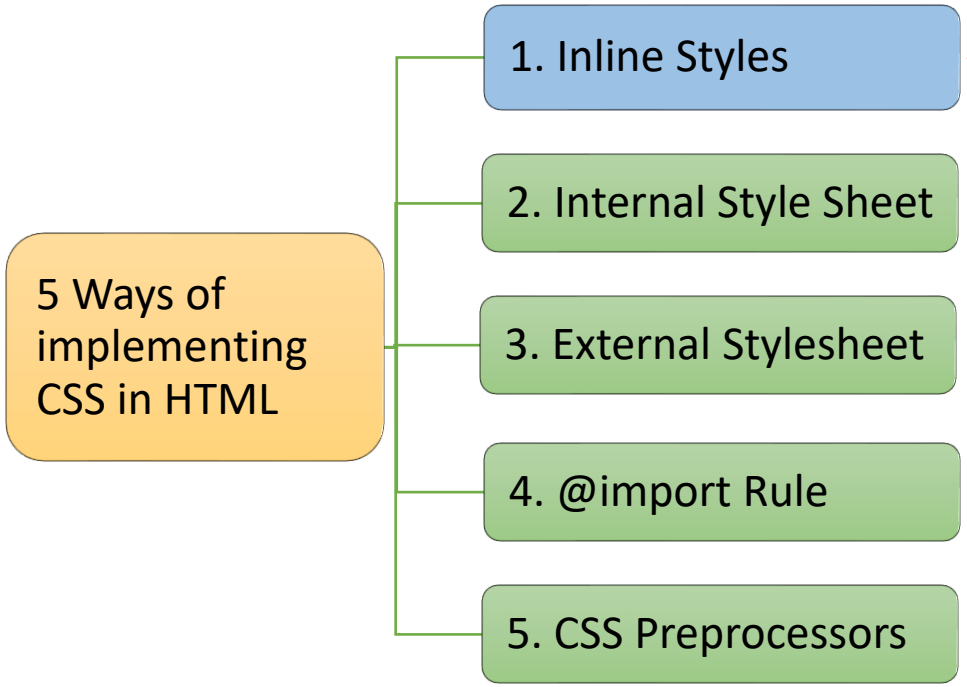
CSS

Q. What is CSS? What are the 5 ways to Implement CSS in HTML? V. IMP.



Q. What is Inline Style in CSS? When to use it in real applications?

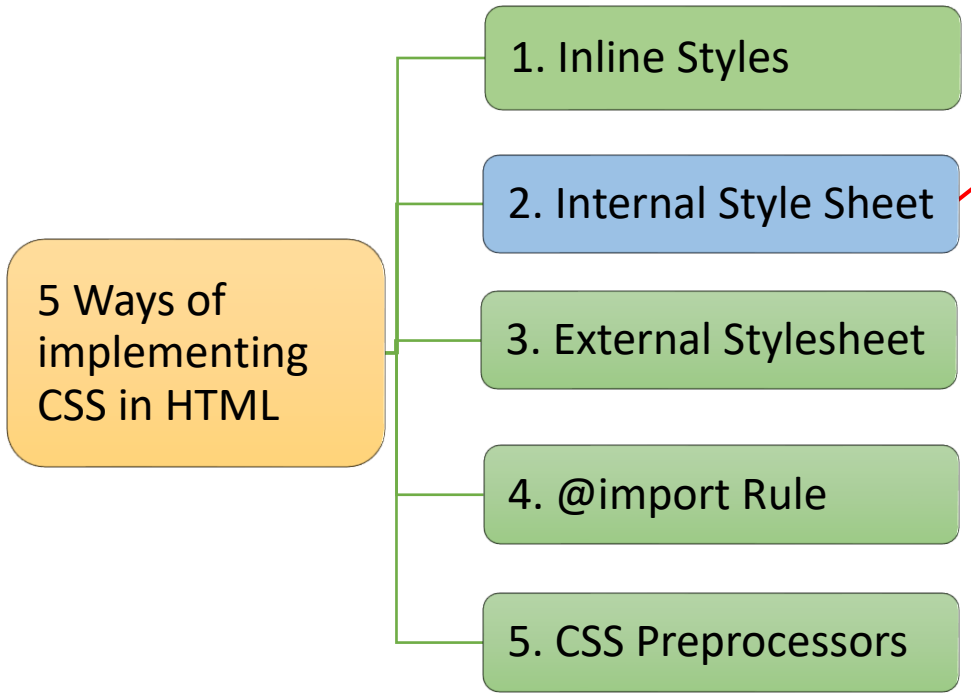
- Inline Styles apply styles directly to individual HTML elements using the **style attribute**.
- This method is suitable for applying styles to a **single element**.




```
<html>
  <head> </head>
  <body>
    <h1 style="text-align: center">Inline Styles</h1>
    <p style="color: green">Paragraph</p>
  </body>
</html>
```

Q. What is Internal Stylesheet in CSS? When to use it in real applications?

- Internal Style Sheets can be implemented by adding the **<style> element** in the <head> section of HTML.
- This method is suitable for **smaller projects.**




```
<html>
  <head>
    <style>
      div {
        text-align: left;
      }
      .para {
        color: green;
      }
    </style>
  </head>
  <body>
    <div>Internal Style Sheet</div>
    <p class="para">Paragraph</p>
  </body>
</html>
```

Q. What is External Stylesheet in CSS? When to use it in real applications? V. IMP.

- In External Stylesheet, a **separate CSS file** is created(.css extension) and link it in the HTML file using the **<link> element**.
- This is most common and **recommended**.

5 Ways of implementing CSS in HTML

1. Inline Styles
2. Internal Style Sheet
3. External Stylesheet
4. @import Rule
5. CSS Preprocessors

2-CssTypes3.css

```css
h1 {
    text-align: center;
}
p {
    color: green;
}
```

```html
<html>
    <head>
        <link rel="stylesheet" href="2-CssTypes3.css">
    </head>
    <body>
        <h1>Internal Style Sheet</h1>
        <p>Paragraph</p>
    </body>
</html>
```

Q. What are the 5 advantages of External Stylesheet? V. IMP.

5 Advantages of CSS:

1. **Separation** of content(HTML) and presentation(Style).
2. **Reusability** of CSS classes in multiple elements.
3. Keeps things organized and **structured**.
4. Adapts to Different **Devices**.
5. Improves Website **Speed.**

1-Css2.css > ...

```css
.container {
    background-color: #f0f0f0;
    padding: 20px;
}
.heading {
    color: #333;
}
.paragraph {
    font-family: Arial, sans-serif;
}
```

Q. How do you Include CSS in a webpage or HTML?

- 1. External Stylesheet: By using the **<link> element** in the <head> section of the HTML document.

```
<head>
  <title>Sass Example</title>
  <link rel="stylesheet" href="4-PreProcessor.css" />
</head>
```

- 2. Internal Style Sheet: By using the **<style> element** in the <head> section.

```
<head>
  <style>
    h1 {
      text-align: center;
    }
  </style>
</head>
```

- 3. Inline Styles: Apply styles directly to individual HTML elements using the **style attribute**.

```
<body>
  <h1 style="text-align: center">Inline Styles</h1>
  <p style="color: green">Paragraph</p>
</body>
```

Q. How to implement CSS using @import rule?

- @import rule is to include **external CSS** file to main CSS file.
- This method is less commonly used due to potential **performance issues**.

```css
/* 3-Main.css */
@import url("3-External.css");

p {
  font-size: 18px;
}
```

```css
/* 3-External.css */
h1 {
  color: blue;
}
```

5 Ways of implementing CSS in HTML

1. Inline Styles
2. Internal Style Sheet
3. External Stylesheet
4. @import Rule
5. CSS Preprocessors

```html
<html>
  <head>
    <link rel="stylesheet" href="3-Main.css" />
  </head>
  <body>
    <h1>Imported Styles</h1>
    <p>Paragraph</p>
  </body>
</html>
```

Q. What is CSS Preprocessors? What is SCSS? V. IMP.

- CSS preprocessors are **scripting languages** that provide additional syntax and features that are not available in traditional CSS. For example, variables, loops, and conditional statements.
- SCSS is a type of CSS preprocessor.

4-PreProcessor.scss > ...

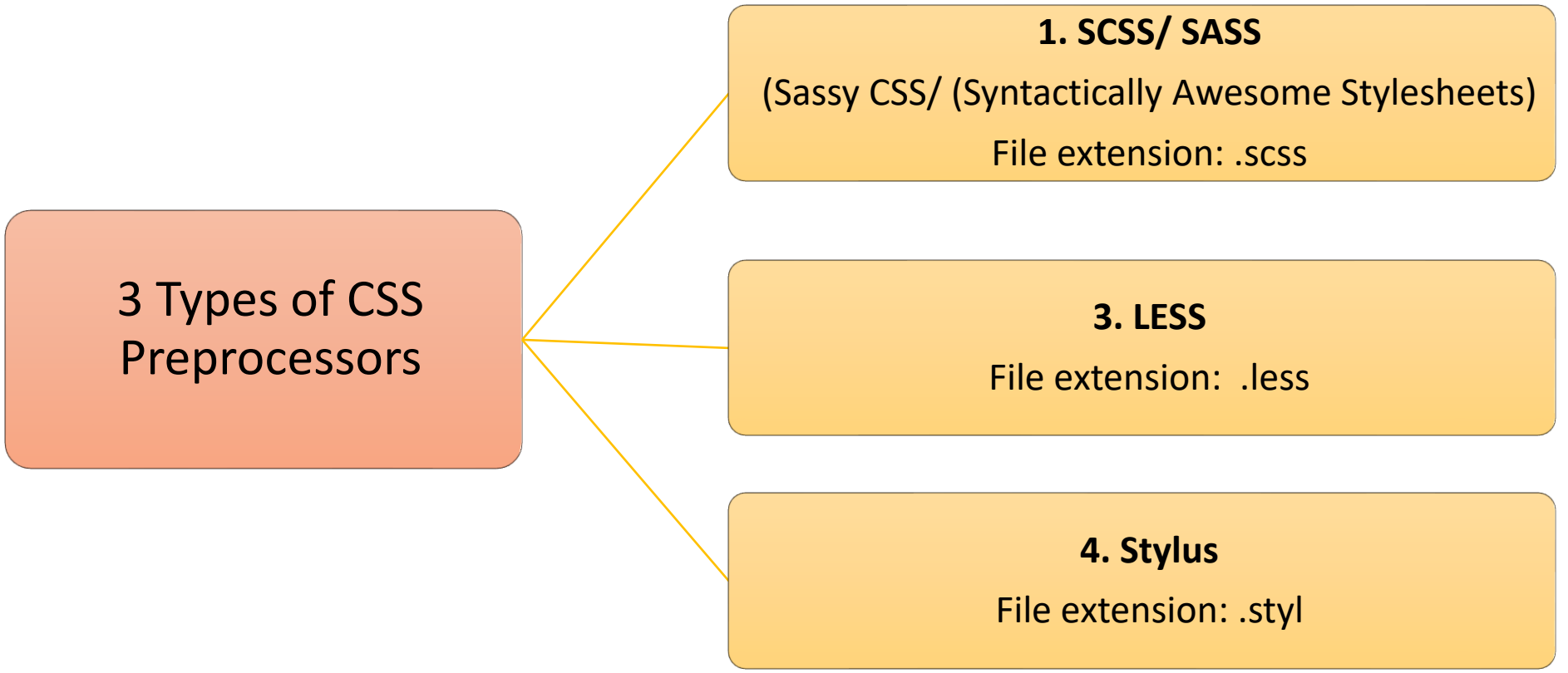
```scss

$primary-color: blue;
$base-font-size: 16px;

.heading {
    font-size: $base-font-size * 1.2;
    color: $primary-color;
}
```

4-PreProcessor.css > ...

```css

/* Generated css from scss */

.heading {
  font-size: 19.2px;
  color: blue;
}
```

```html
<html>
  <head>
    <title>Sass Example</title>
    <link rel="stylesheet" href="4-PreProcessor.css" />
  </head>
  <body>
    <h1 class="heading">Interview Happy</h1>
    <p class="paragraph">Crack Interviews</p>
  </body>
</html>
```

Q. What are the 3 Types of CSS Preprocessors?



2. Selectors in CSS

Q. What are Selectors in CSS? How many types of selectors are there?

Q. Difference between ID, Element & Class selector? When to use which selector?

Q. What are Universal selectors?

Q. What are the Descendant Selectors in CSS?

Q. What are Attribute Selectors in CSS? What are it's 5 Types?

Q. What are Child Selectors in CSS?

Q. How Child Selector is different from Descendent selectors?

Q. What are Pseudo-class Selector & Pseudo-element Selector?

Q. What are Selectors in CSS? How many Types of selectors are there? V. IMP.

- In CSS, selectors are patterns that are used to select and style HTML elements.

Types of Selectors

1. ID Selector(#)
2. Element Selector(no symbol)
3. Class Selector(.)
4. Universal Selector(*)
5. Descendant selector
6. Child Selector(>)

```html
<html>
  <head>
    <link rel="stylesheet" href="3-Main.css" />
  </head>
  <body>
    <h1>Imported Styles</h1>
    <p>Paragraph</p>
  </body>
</html>
```

```css
h1 {
  color: blue;
}
```

Q. Difference between ID, Element & Class selector? When to use which selector? V. IMP.

```html
<div id="my-id">
  <p>This background will be yellow.</p>
</div>
```

```css
/* # ID Selector */
#my-id {
  background-color: yellow;
}
```

This background will be yellow.

```html
<h2>This text font size will be 20px.</h2>
```

```css
/* Element Selector */
h2 {
  font-size: 20px;
}
```

This text font size will be 20px.

```html
<div class="my-class">
  <p>This text will be blue.</p>
</div>
```

```css
/* . Class Selector */
.my-class {
  color: blue;
}
```

This text will be blue.

Q. Difference between ID, Element & Class selector? When to use which selector? V. IMP.

1. ID Selector (#)

- ID selectors are used for unique elements on a page, and they should only be assigned to one element.

```
/* # ID Selector */
#my-id {
  background-color: yellow;
}
```

This background will be yellow.

2. Element Selector

- Element selectors are used when you want to apply a style to all elements of a specific type.

```
/* Element Selector */
h2 {
  font-size: 20px;
}
```

This text font size will be 20px.

3. Class Selector (.)

- Class selectors are used when you want to apply a style to multiple elements with the same class.

```
/* . Class Selector */
.my-class {
  color: blue;
}
```

This text will be blue.

Q. What are Universal selectors?

- Universal selector applied the specified styles to all elements, regardless of their type or attributes.

```
<body>
  <div id="my-id">
    <p>Interview</p>
  </div>

  <h2>Happy</h2>
</body>
```

Interview

Happy

```
/* * Universal Selector */
* {
  color: blue;
  font-size: 24px;
}
```

Q. What are the Descendant Selectors in CSS?

- A descendant selector in CSS is a selector that targets an element that is a descendant of another specified element(**nested elements**).

```html
<body>
  <div class="container">
    <p>Paragraph 1</p>
    <p>Paragraph 2</p>
    <h4>No css applied</h4>
  </div>

  <ul>
    <li>Item 1</li>
    <li>Item 2</li>
  </ul>
</body>
```

```css
.container p {
  color: blue;
  font-size: 16px;
}

ul li {
  background-color: red;
  padding: 5px;
  margin: 5px;
}
```

Paragraph 1

Paragraph 2

- Item 1
- Item 2

Q. What are Attribute Selectors in CSS? What are it's 5 Types?

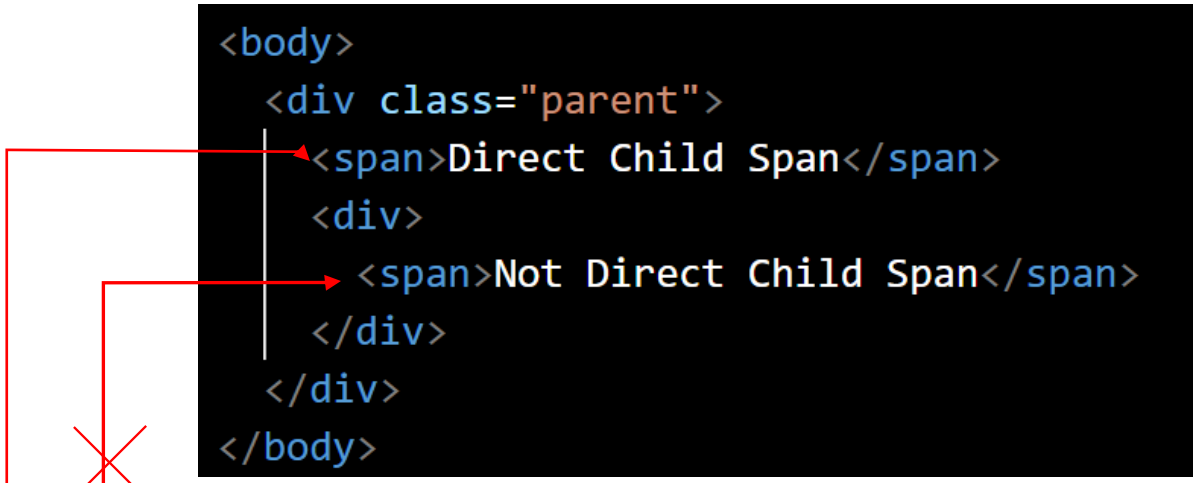
- Attribute selectors in CSS target HTML elements based on their attributes and their values.

```html
<input type="text"/>

<input type="password"/>

<a href="https://www.example.com">Link 1</a>

<a href="http://www.example.pdf">Link 2</a>
```


```css
/* Attribute Exists Selector */
input[type] {
  border: 1px solid black;
}
/* Attribute Equals Selector */
a[href="https://www.example.com"]
{
  color: blue;
}
/* Attribute Contains Selector */
input[type*="text"] {
  background-color: #f0f0f0;
}
/* Attribute Starts With Selector */
a[href^="https://"]
{
  color: green;
}
/* Attribute Ends With Selector */
a[href$=".pdf"] {
  text-decoration: underline;
}
```

Q. What are Child Selectors in CSS?

- A child selector targets an element that is a **immediate child** of specified parent element.
- It's denoted by the **>** symbol.



```html
<body>
  <div class="parent">
    <span>Direct Child Span</span>
    <div>
      <span>Not Direct Child Span</span>
    </div>
  </div>
</body>
```

```css
/* Style applied to direct children
of class "parent" that are spans */
.parent > span {
  color: green;
}
```

Direct Child Span

Not Direct Child Span

Q. How Child Selector is different from Descendant selectors?

- The child selector targets **immediate children**, while the descendant selector targets all descendants, regardless of depth.

Q. What are Pseudo-class Selector and Pseudo-element Selector?

- A pseudo-class selector is used to define the **special state** of an HTML element, like when a user hovers over an element.
- A pseudo-element selector targets a **specific part** of an element, such as ::before to insert content before an element.

```html
<body>
  <div class="box">
    <p>Hover over me</p>
    <span>Content added before me</span>
  </div>
</body>
```

```css
/* Pseudo-class Selector */
p:hover {
  color: blue;
}
```

Hover over me

```css
/* Pseudo-element Selector */
span::before {
  content: "Yes: ";
  font-weight: bold;
}
```

Yes: Content added before me

3. Layout and Positioning

Q. Explain the concept of the Box Model in CSS.

Q. What is the difference between Padding & Margin?

Q. What is the difference between display: inline & display: block?

Q. What is the difference between display: inline & display: inlineblock?

Q. What is z-index property in CSS? How same z-index elements will be displayed?

Q. What is the role of float property in CSS?

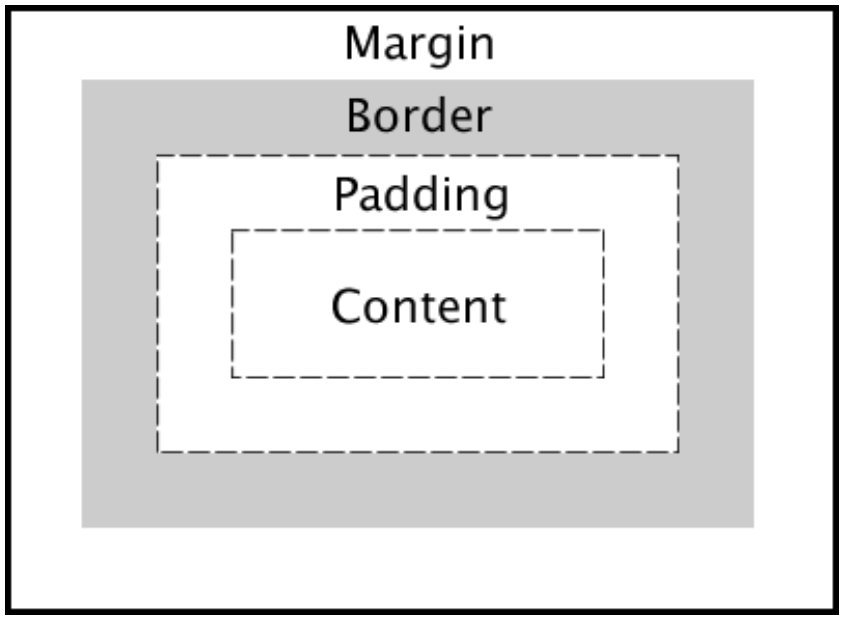
Q. What is the difference between position: relative/ absolute/ fixed?

Q. Explain the concept of the Box Model in CSS. V. IMP.

- The "box model" describes the **layout of elements** on a webpage in terms of their dimensions and spacing.

1. **Content**: This is the actual content of the element, such as text, images, videos, etc.
2. **Padding**: This is the space between the content and the border.
3. **Border**: This is the line that separate the content from the surrounding elements.
4. **Margin**: This is the space between the borders of adjacent elements. Margins help control the space between elements.

- In CSS, you can control each of these components using various properties like width, height, padding border, margin.



- Without box model

elem ent1 elem ent2 ele ment3

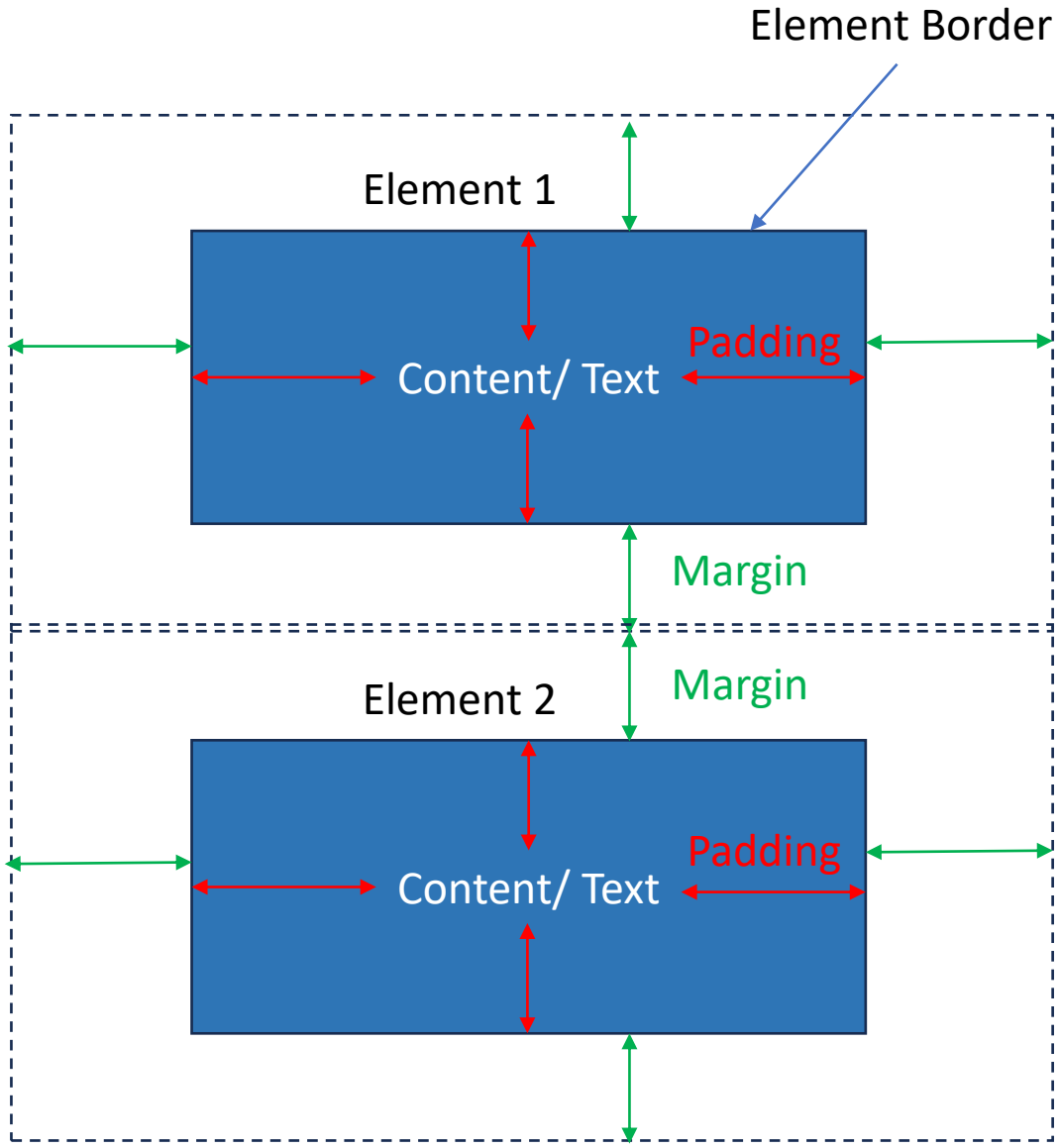
- With box model

elem ent1 | elem ent12 | ele ment3

Q. What is the difference between Padding and Margin? V. IMP.

- Padding is the space between the content of an element and its border.
- Margin is the space outside an element, between the element and its surrounding elements.

```
.h1 {
  border: 1px solid black;
  margin-top: 20px;
  margin-right: 30px;
  margin-bottom: 40px;
  margin-left: 50px;
  padding-top: 10px;
  padding-right: 20px;
  padding-bottom: 30px;
  padding-left: 40px;
}
```



Q. What is the difference between display: inline & display: block? V. IMP.

- Block elements take up the **full width** of their parent container by default
- Inline elements take up as much width as per the **width of content only.**

```html
<body>
  <h3>Display Properties</h3>
  <div class="block">Block Element</div>
  Text1
  <div class="inline">Inline Element</div>
  Text2
</body>
```

```css
.block {
  display: block;
  border: 1px solid black;
  padding: 20px;
  margin: 20px;
}
```

```css
.inline {
  display: inline;
  border: 1px solid black;
  padding: 20px;
  margin: 20px;
}
```

Block Element

Text1 Inline Element Text2

Q. What is the difference between display: inline and display: inlineblock? V. IMP.

- Inline elements do not accept width and height values whereas inline-block accept width and height also.

```
<body>
  <div class="inline">Inline Element</div>Text2<br />
  <div class="inline-block">Inline-Block</div>Text3
</body>
```

```
.inline {
  display: inline;
  border: 1px solid black;
  height: 50px;
  width: 50px;
  margin: 50px;
}
```

Inline Element Text2

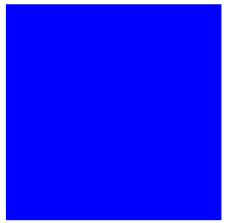
```
.inline-block {
  display: inline-block;
  border: 1px solid black;
  height: 50px;
  width: 50px;
  margin: 50px;
}
```

Inline-Block Text3

Q. What is z-index property in CSS? How same z-index elements will be displayed?

- z-index property in CSS is used to control the **stacking order** of positioned elements.
- Elements with a higher z-index value will be displayed on top of elements with a lower z-index value.
- If two elements have the same z-index, the one that comes later in the HTML source will appear on top.

```html
<body>
  <div class="container">
    <div class="box1"></div>
    <div class="box2"></div>
  </div>
</body>
```



```css
.box1 {
  position: absolute;
  width: 100px;
  height: 100px;
  background-color: red;
  z-index: 1;
}

.box2 {
  position: absolute;
  width: 100px;
  height: 100px;
  background-color: blue;
  z-index: 2;
}
```

Q. What is the role of float property in CSS?

- The float property in CSS is used to specify how an element should align or "float" within its parent container.

```html
<body>
  <div class="sidebar">
    <h2>Sidebar</h2>
  </div>

  <div class="main-content">
    <h1>Main Content</h1>
  </div>
</body>
```

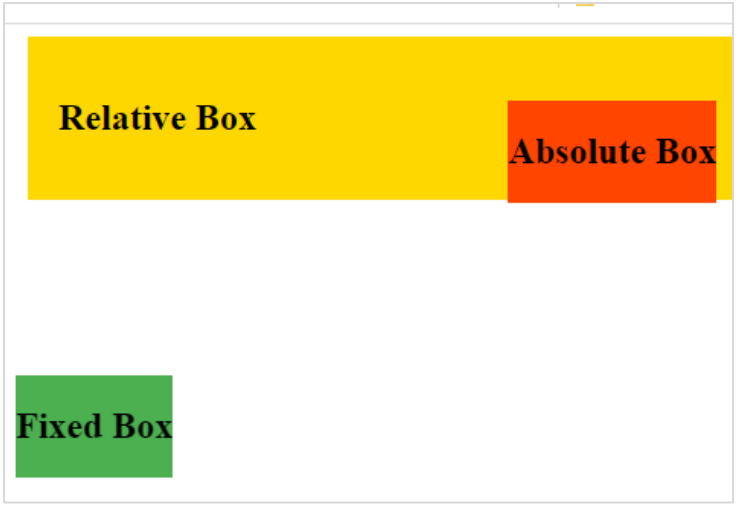
```css
.sidebar {
  width: 30%;
  float: right;
  background-color: lightblue;
  padding: 20px;
  box-sizing: border-box;
}

.main-content {
  width: 70%;
  padding: 20px;
  box-sizing: border-box;
  background-color: lightgreen;
}
```

Q. What is the difference between position: relative/ absolute/ fixed? V. IMP.

- position: relative positions an element relative to its normal position of the page.
- position: absolute is positioned as per the parent element.
- position: fixed will fix the position of element even if the page is scrolled.

```html
<body>
  <div class="relative-box">
    <h2>Relative Box</h2>
  </div>

  <div class="absolute-box">
    <h2>Absolute Box</h2>
  </div>

  <div class="fixed-box">
    <h2>Fixed Box</h2>
  </div>
</body>
```

```css
.relative-box {
  position: relative;
  left: 20px;
  padding: 20px;
  background-color: #ffd700;
}

.absolute-box {
  position: absolute;
  top: 50px;
  right: 20px;
  background-color: #ff4500;
}

.fixed-box {
  position: fixed;
  bottom: 50px;
  left: 20px;
  background-color: #4caf50;
}
```



4. Responsive Design, Flexbox & Grid Layout

Q. What is the role of meta viewport tag?

Q. What are Media Queries in CSS?

Q. What is display: flex in CSS?

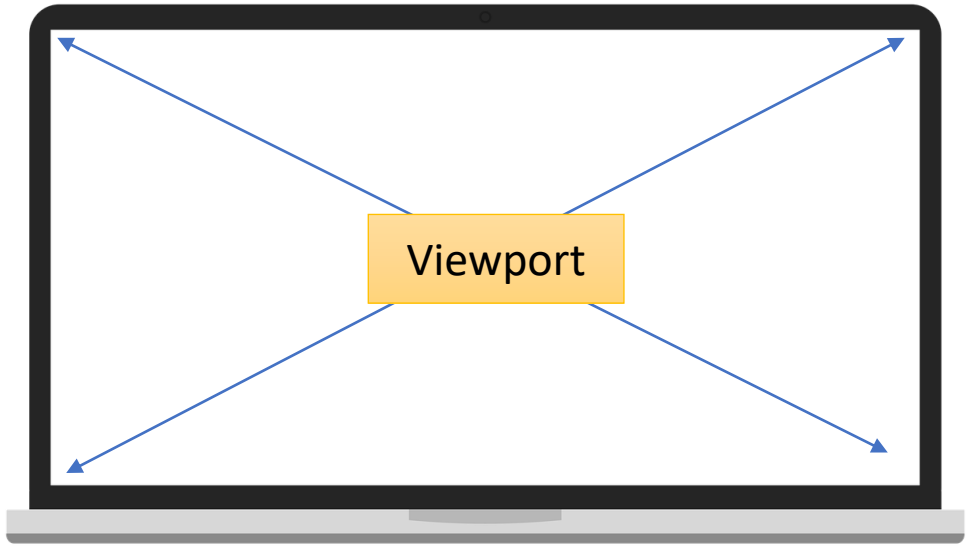
Q. What is CSS Grid Layout in CSS?

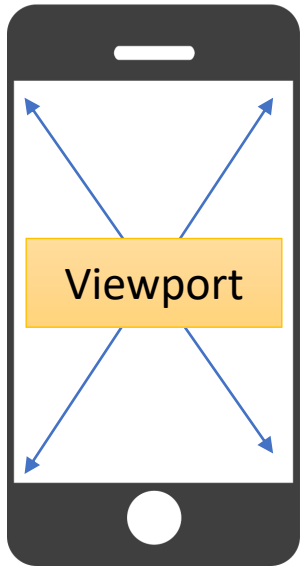
Q. How can you create a Sticky Header in CSS?

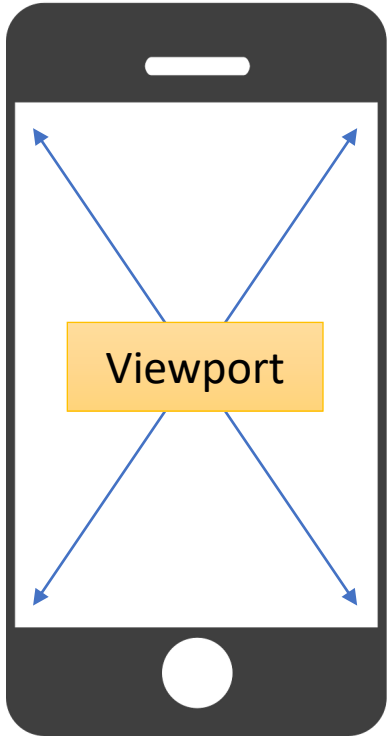
Q. What are the 5 best practices for creating a Responsive Design using CSS?

Q. What is the role of meta viewport tag? V. IMP.

- In web design, the "viewport" is the portion of the webpage that the user can see on their screen at any given time, without scrolling.







Q. What is the role of meta viewport tag? V. IMP.

- The <meta name="viewport"> tag in HTML is used to adjust the webpage on different devices and screen sizes.

- Name attribute specifies that the meta tag is a meta viewport.

- Content width property tells the browser to set the width of the viewport to the width of the device's screen.

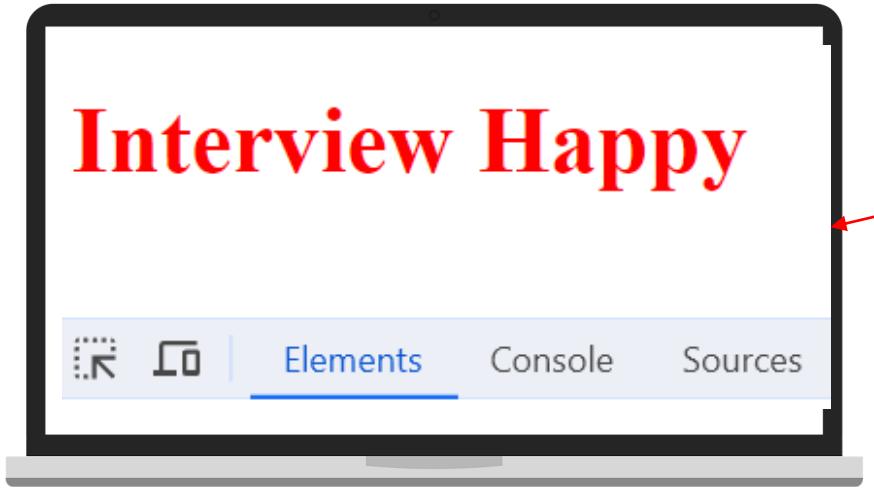
- Content initial-scale property sets the initial zoom level when the page is first loaded. A value of 1.0 means that the page is displayed at its actual size without any zooming.

```
<head>
  <meta charset="UTF-8" />

  <meta name="viewport" content="width=device-width, initial-scale=1.0" />

  <link rel="stylesheet" href="18-MediaQueries.css" />
</head>
```

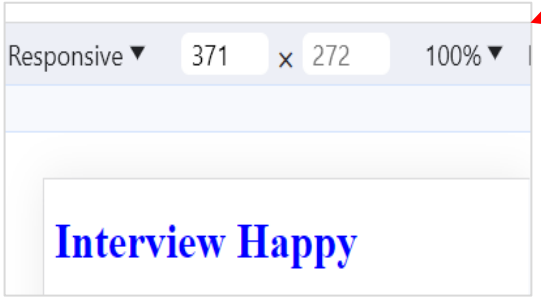
Q. What are Media Queries in CSS? V. IMP.

- Media queries in CSS allows you to apply different styles to elements based on the size of device screen.

```
<body>
  <h1>Interview Happy</h1>
</body>
```




```
h1 {
  color: red;
}
@media screen and (max-width: 768px) {
  h1 {
    color: blue;
  }
}
```



Q. What is display: flex in CSS?

- display:flex turns an element into a container that structure and align its child elements in a better way. It enables a flexbox layout for the container.
- justify-content property aligns flex items along the main axis of the flex container. It controls the spacing between and around items.

Home About Services Contact

Home About Services Contact

```html
<body>
  <nav class="nav-container">
    <a href="#">Home</a>
    <a href="#">About</a>
    <a href="#">Services</a>
    <a href="#">Contact</a>
  </nav>
</body>
```

```css
.nav-container {
  display: flex;
  justify-content: space-around;
  background-color: #333;
  padding: 10px;
}

.nav-container a {
  color: white;
}
```

Q. What is CSS Grid Layout in CSS?

- The CSS Grid Layout Module offers a grid-based layout system, with rows and columns.

```
<body>
  <div class="grid-container">
    <div>1</div>
    <div>2</div>
    <div>3</div>
    <div>4</div>
    <div>5</div>
    <div>6</div>
  </div>
</body>
```

Without grid css

1
2
3
4
5
6

With grid css

```
.grid-container {
  display: grid;
  grid-template-columns: auto auto auto;
}
```

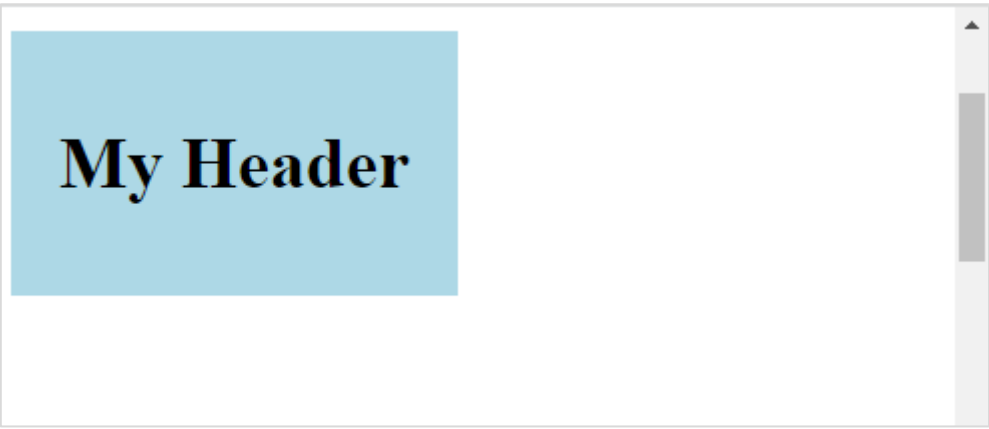
1	2	3
4	5	6

Q. How can you create a Sticky Header in CSS?

- Position: fixed can be used to implement sticky header in CSS.

```
<body>
  <div class="header">
    <h2>My Header</h2>
  </div>
  <div class="content"></div>
</body>
```

```
.header {
  position: fixed;
  padding: 10px 16px;
  background: lightblue;
}
.content {
  height: 2000px;
}
```



Q. What are the 5 best practices for creating a Responsive Design using CSS? V. IMP.

5 Best practices for creating responsive design

1. Use Responsive Units(%)
2. Use Media Queries
3. Flexible Layout
4. Images and Media size
5. Viewport Meta Tag

```css
.container {
  width: 80%;
}

@media screen and (min-width: 768px) {
  .header {
    font-size: 24px;
  }
}

.container {
  display: flex;
}

img {
  max-width: 100%;
  height: auto;
}
```

Q. What are the 5 best practices for creating a Responsive Design using CSS? V. IMP.

1. **Use Responsive Units**:

Use relative units like percentages (%) or em units for sizes (width, height, padding, margin, etc.) instead of fixed units like pixels (px).

2. Use Media Queries:

Media queries allow you to apply different styles based on the characteristics of the device, such as screen width, height, and orientation. They are essential for creating a responsive design.

3. Flexible Grid Systems:

Use CSS Grid or Flexbox to create flexible layouts that can adapt to different screen sizes and orientations.

4. Images and Media:

Use max-width: 100%; on images to ensure they don't overflow their containers. This allows images to scale down proportionally on smaller screens.

5. Viewport Meta Tag:

Include the viewport meta tag in the <head> of your HTML to control how the page is displayed on mobile devices. It sets the initial scale, width, and zoom behavior.

1. Bootstrap

Q. What is Bootstrap? What are the other 5 responsive design frameworks?

Q. What are the 5 Advantages of using Bootstrap?

Q. What are the 2 ways to include Bootstrap framework for your website.

Q. Explain the Grid System in Bootstrap?

Q. What is the difference between col-xs, col-sm, col-md, col-lg & col-xl?

Q. What are Bootstrap Components? What are Top 10 bootstrap components?

Q. What is a Bootstrap Modal component?

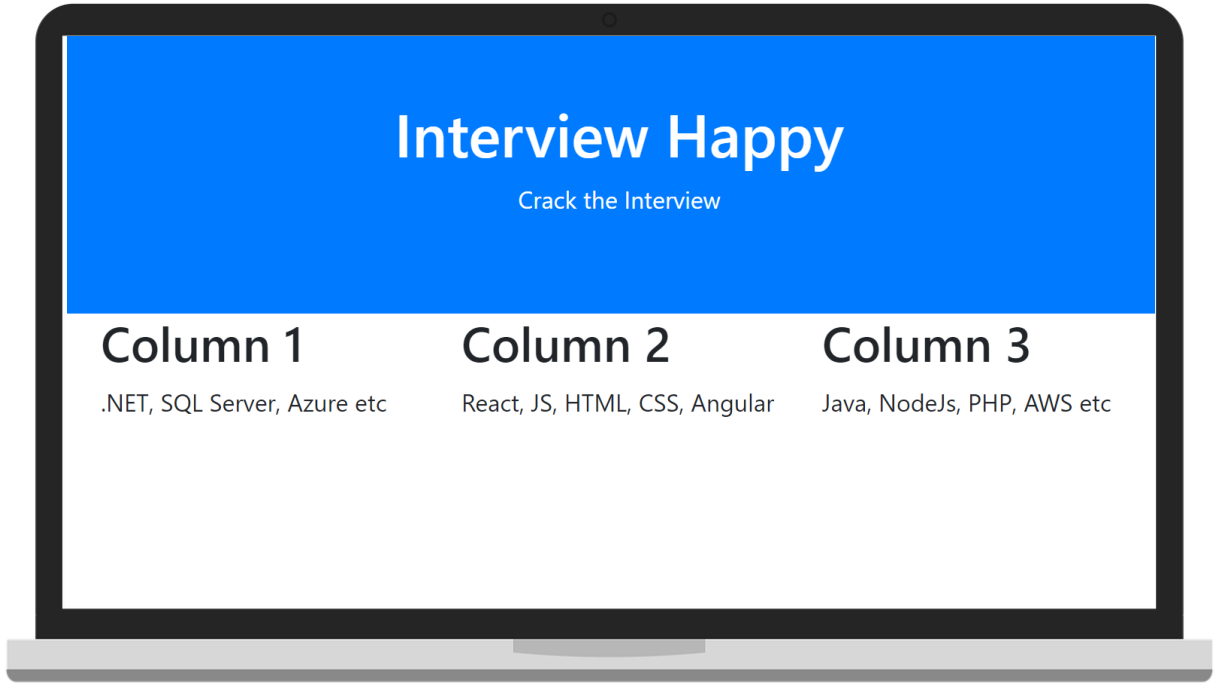
Q. What is a Bootstrap Navigation component?

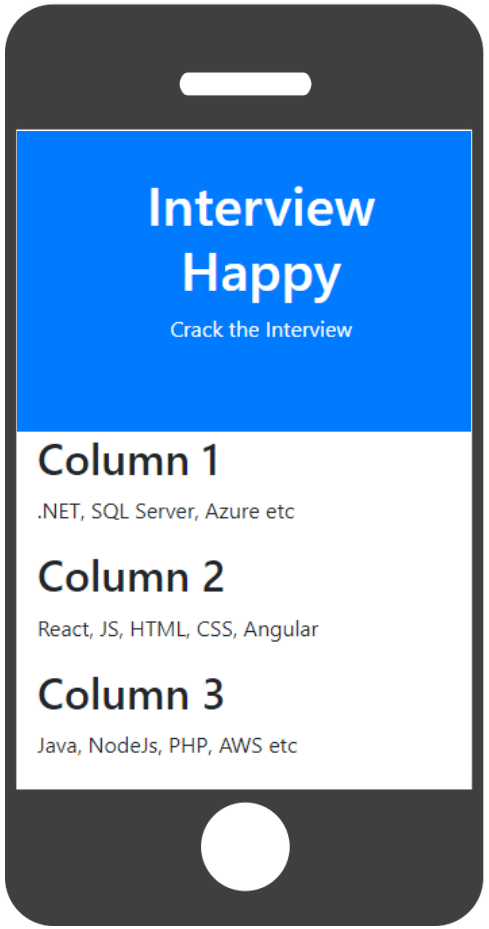
Q. What is Bootstrap Carousel component?

Q. Explain the difference between Bootstrap's container & container-fluid?

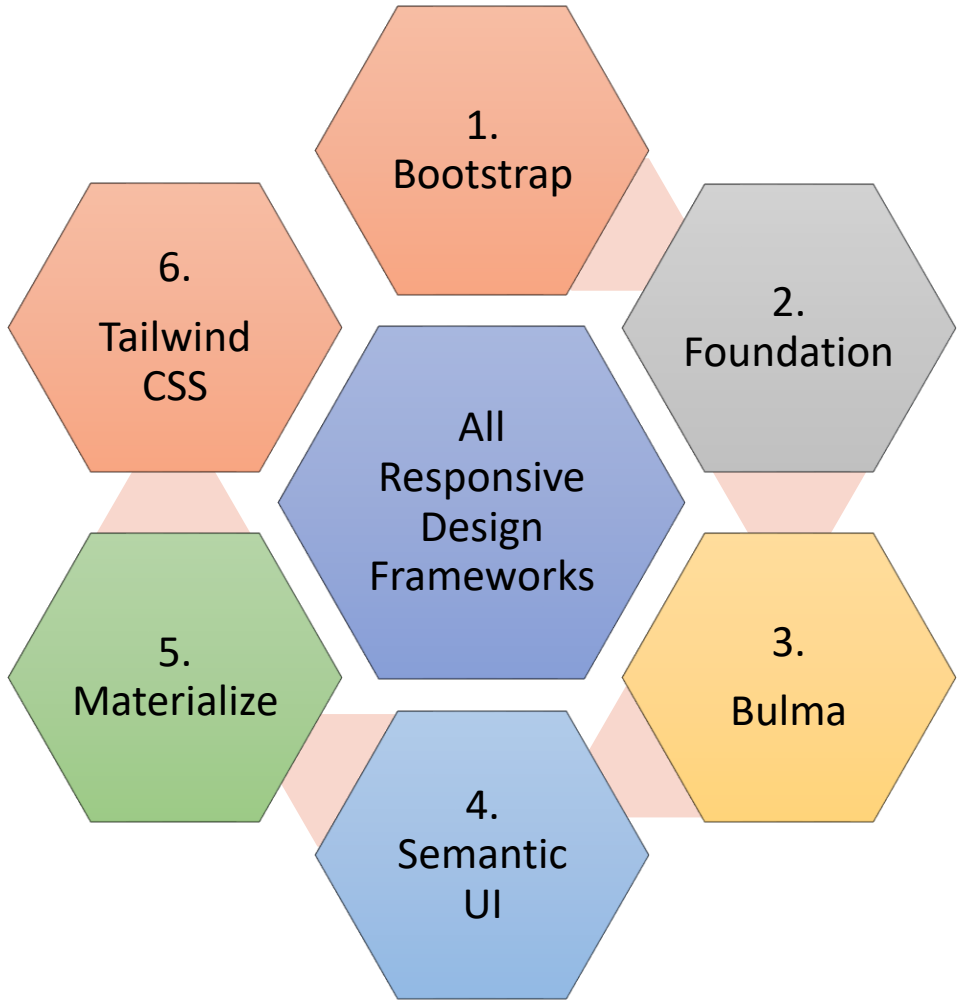
Q. What is Bootstrap? What are the other 5 responsive design frameworks? V. IMP.

- Bootstrap is a popular open-source front-end framework which provides responsive and mobile-first CSS.





Q. What is Bootstrap? What are the other 5 responsive design frameworks? V. IMP.



Q. What are the 5 Advantages of using Bootstrap?

1. Rapid Development

- Bootstrap provides a wide range of ready-to-use components like navigation bars, buttons, forms, modals, and more. This saves time and effort in writing code from scratch.

2. Responsive Design

- Bootstrap is built with a mobile-first approach, meaning it prioritizes designing for smaller screens first and then scales up for larger screens. This ensures websites with Bootstrap is responsive for every device.

3. Cross-Browser Compatibility

- Bootstrap provides a consistent set of styles and components across different browsers, which helps in reducing cross-browser compatibility issues.

4. Large Community and Support

- This means there are plenty of resources, documentation, tutorials, and forums available for assistance.

5. Accessibility

- Bootstrap incorporates best practices for web accessibility, making it easier to create websites that are usable by people with disabilities.

Q. What are the 2 ways to include Bootstrap framework for your website.

2 ways to include Bootstrap:

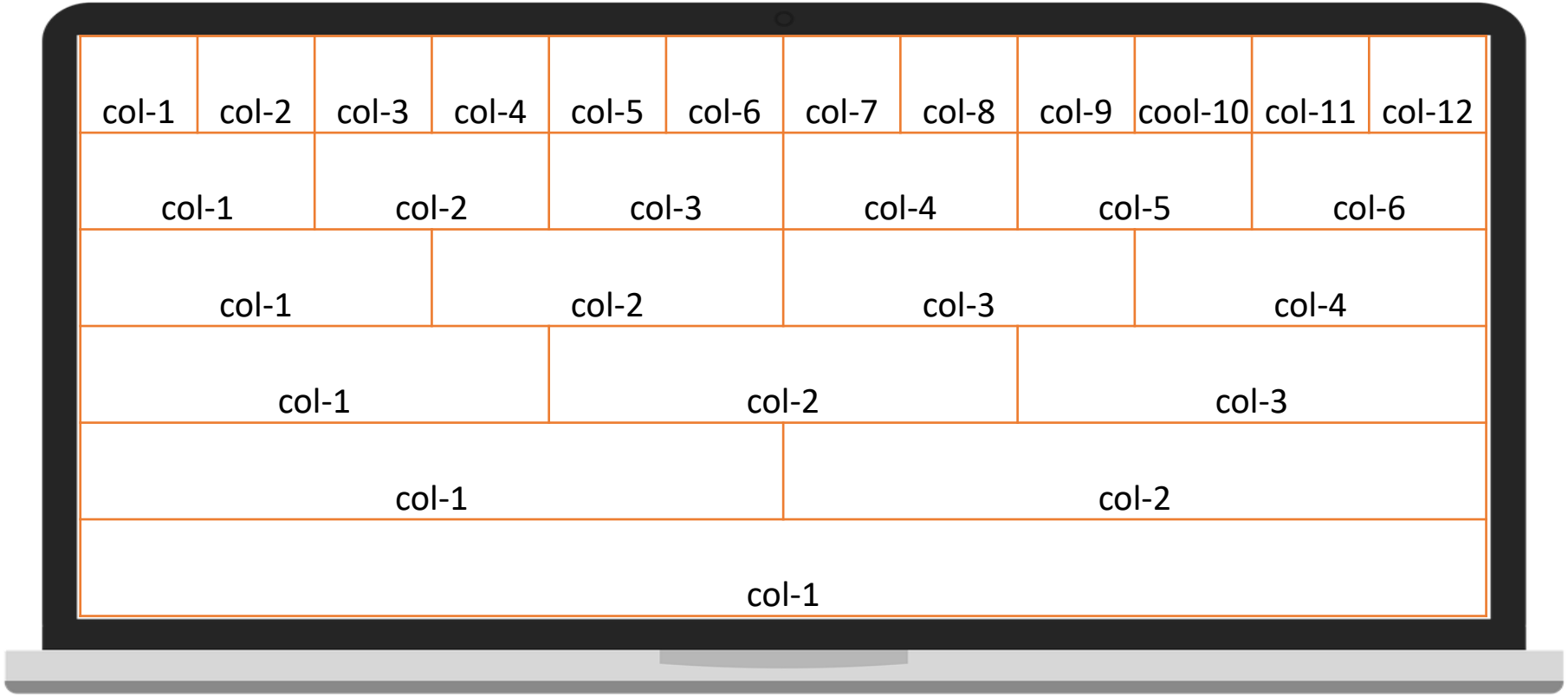
1. Put the link of Bootstrap CDN (Content Delivery Network) in head section.

2. Download the Bootstrap files and add them locally in your HTML document.

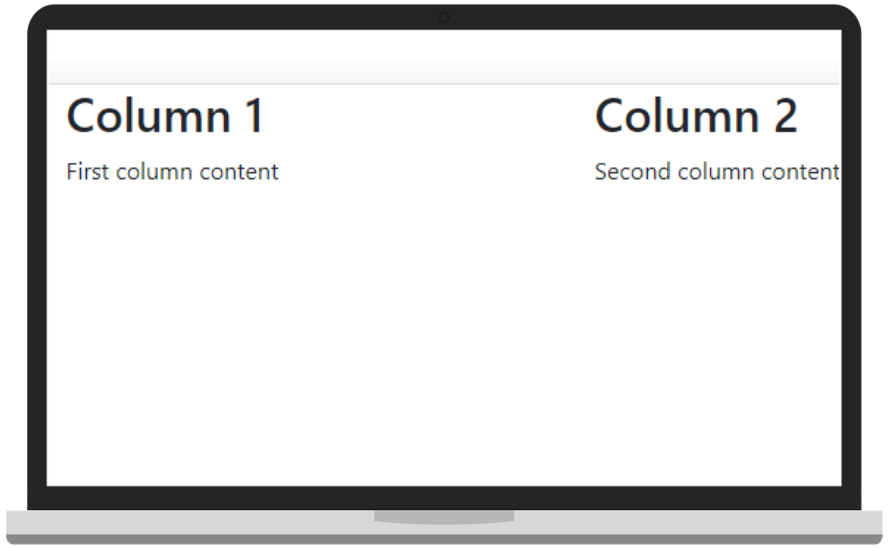
```
<head>
  <title>Bootstrap</title>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1" />
  <!-- Add Bootstrap CSS via CDN -->
  <link rel="stylesheet"
    href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
  />
</head>
```

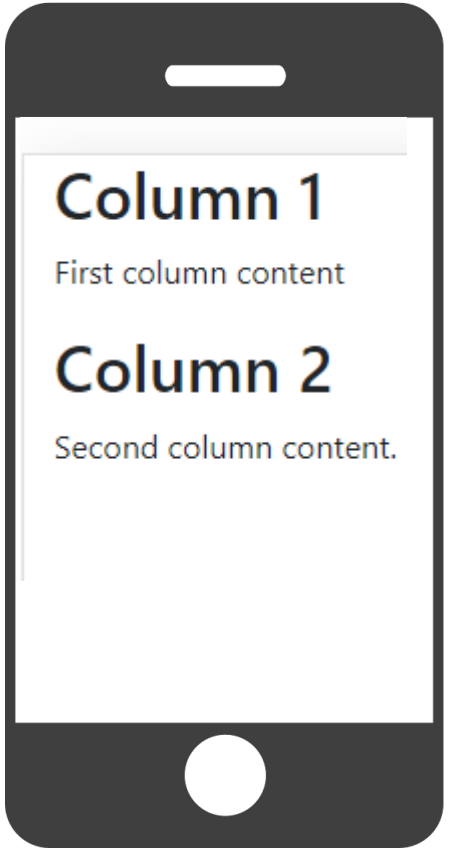
Q. Explain the Grid System in Bootstrap? V. IMP.

- Grid system is 12-column layout and is designed to adapt to various screen sizes.



Q. Explain the Grid System in Bootstrap? V. IMP.



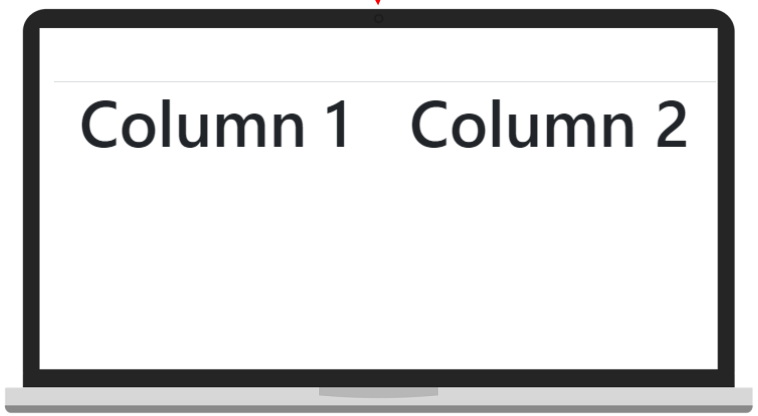



```html
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-6">
        <h2>Column 1</h2>
        <p>First column content</p>
      </div>
      <div class="col-md-6">
        <h2>Column 2</h2>
        <p>Second column content.</p>
      </div>
    </div>
  </div>
</body>
```

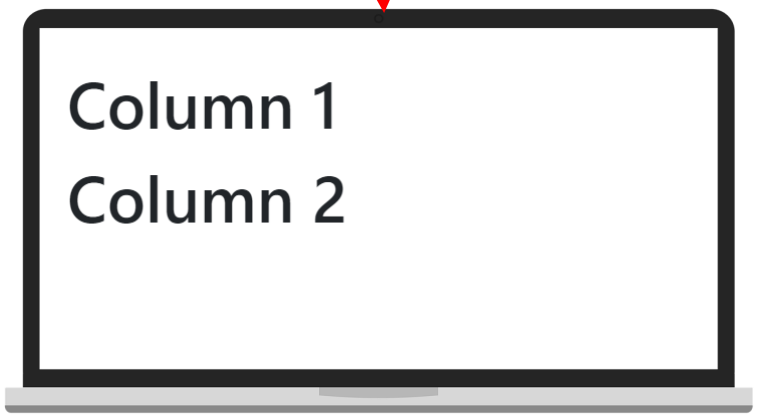
Q. What is the difference between col-xs, col-sm, col-md, col-lg & col-xl?

1. col-xs (Extra Small): Applies to extra small screens (phones).
2. col-sm (Small): Applies to small screens (tablets).
3. col-md (Medium): Applies to medium screens (laptops).
4. col-lg (Large): Applies to large screens (larger desktops).
5. col-xl (Extra Large): Applies to extra large screens (large TVs).

```
<div class="container">
  <div class="row">
    <div class="col-md-6">
      <h2>Column 1</h2>
    </div>
    <div class="col-md-6">
      <h2>Column 2</h2>
    </div>
  </div>
</div>
```




```
<div class="container">
  <div class="row">
    <div class="col-xl-6">
      <h2>Column 1</h2>
    </div>
    <div class="col-xl-6">
      <h2>Column 2</h2>
    </div>
  </div>
</div>
```

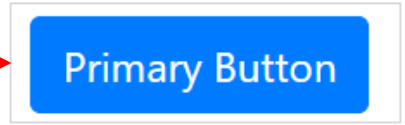


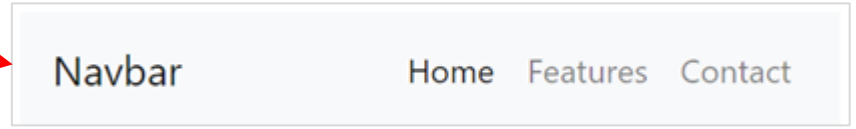
Q. What are Bootstrap Components? What are Top 10 bootstrap components? V. IMP.

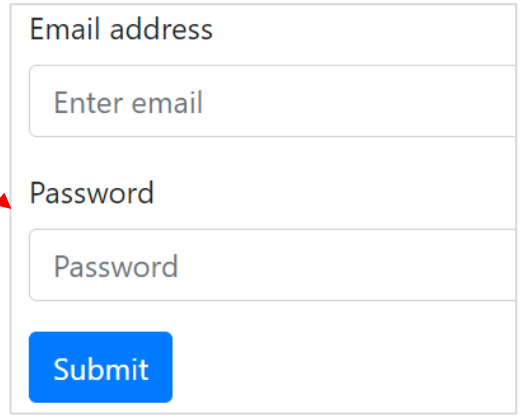
- Bootstrap components are **pre-designed elements** that can be easily integrated into a web project.

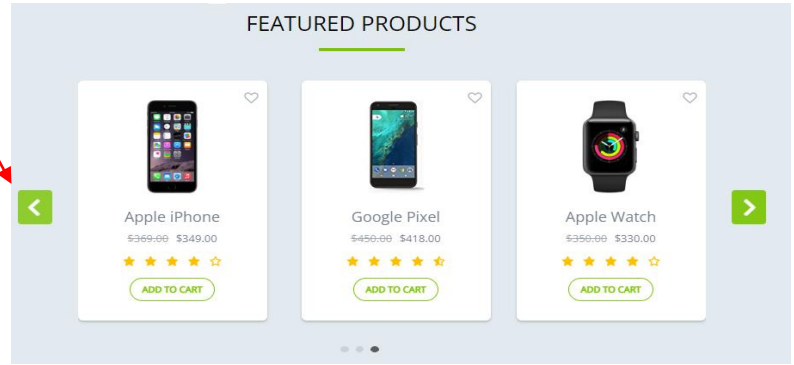
Top 10 Bootstrap components

1. Buttons
2. Navbar
3. Forms
4. Carousel
5. Cards
6. Progress Bars
7. Pagination
8. Spinners
9. Badges
10. Modals







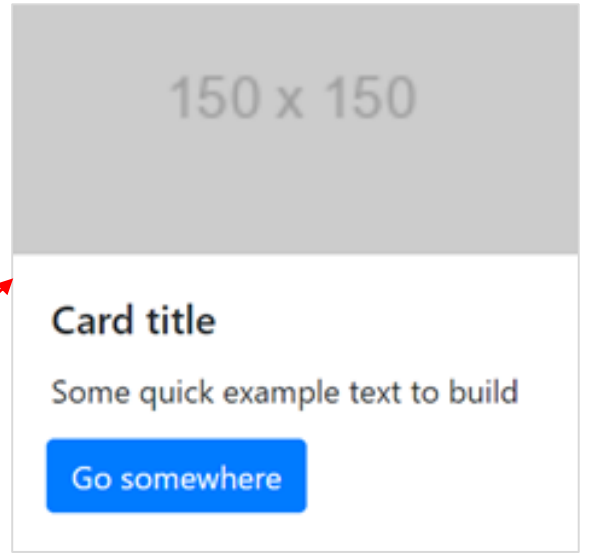


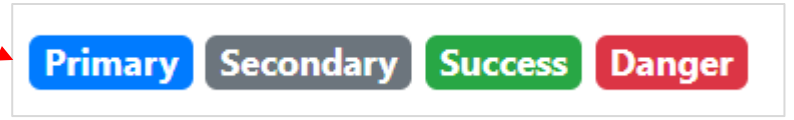
Q. What are Bootstrap Components? What are Top 10 bootstrap components? V. IMP.

- Bootstrap components are **pre-designed elements** that can be easily integrated into a web project.

Top 10 Bootstrap components

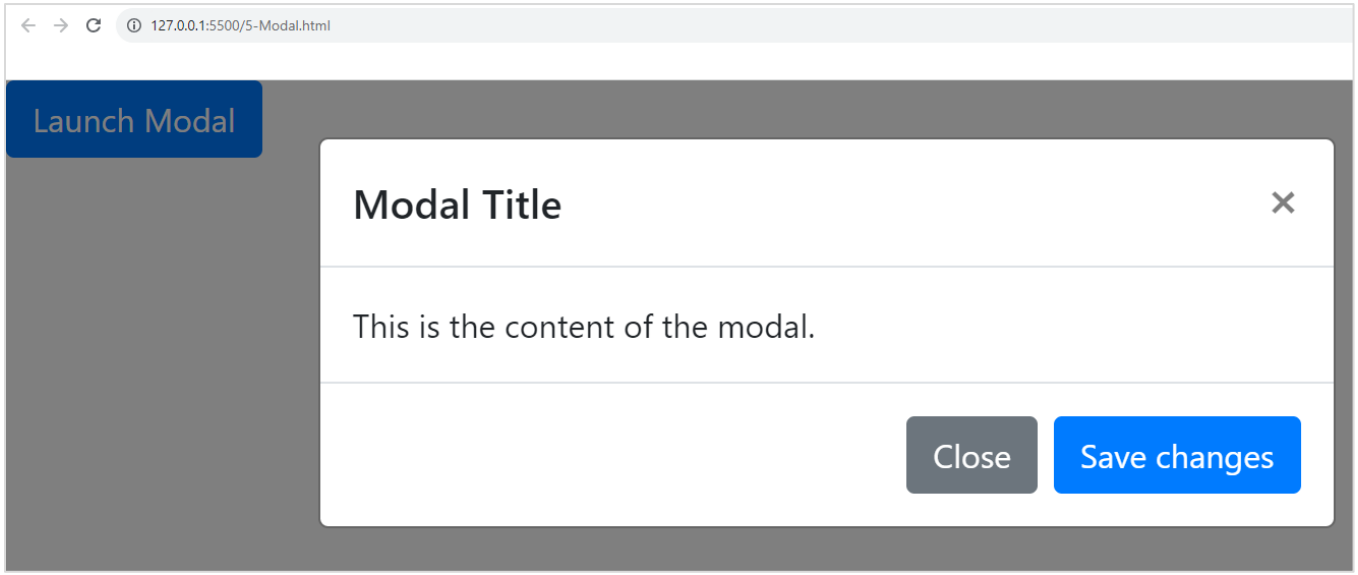
1. Buttons
2. Navbar
3. Forms
4. Carousel
5. Cards
6. Progress Bars
7. Pagination
8. Spinners
9. Badges
10. Modals





Q. What is a Bootstrap Modal component?

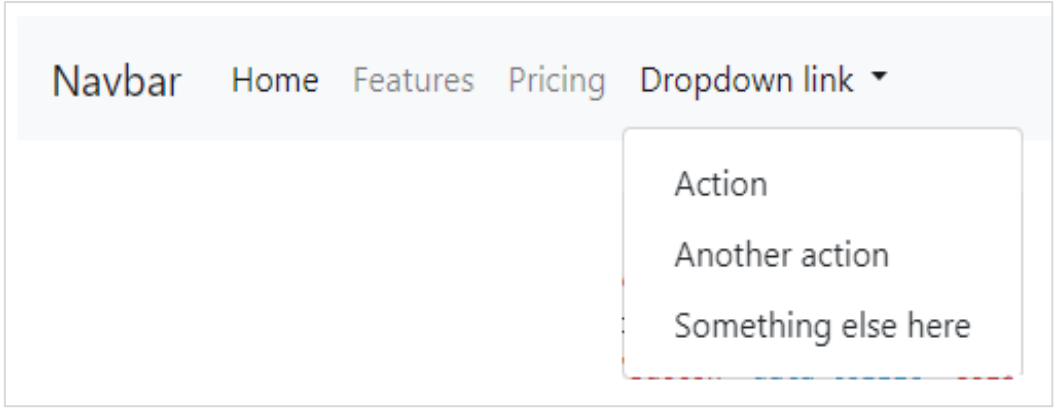
- A Bootstrap modal is a lightweight, customizable, and interactive dialog box or popup window that appears on top of the main content of a web page.




```
<body>
  <!-- Button to trigger the modal -->
  <button type="button" class="btn btn-primar
    Launch Modal
  </button>

  <!-- Modal content -->
  <div class="modal fade" id="exampleModal" t
    <div class="modal-dialog">
      <div class="modal-content">
        <div class="modal-header">
          <h5 class="modal-title" id="example
          <button type="button" class="close"
            <span aria-hidden="true">&times;<
          </button>
        </div>
        <div class="modal-body">
          This is the content of the modal.
        </div>
        <div class="modal-footer">
```

Q. What is a Bootstrap Navigation component?

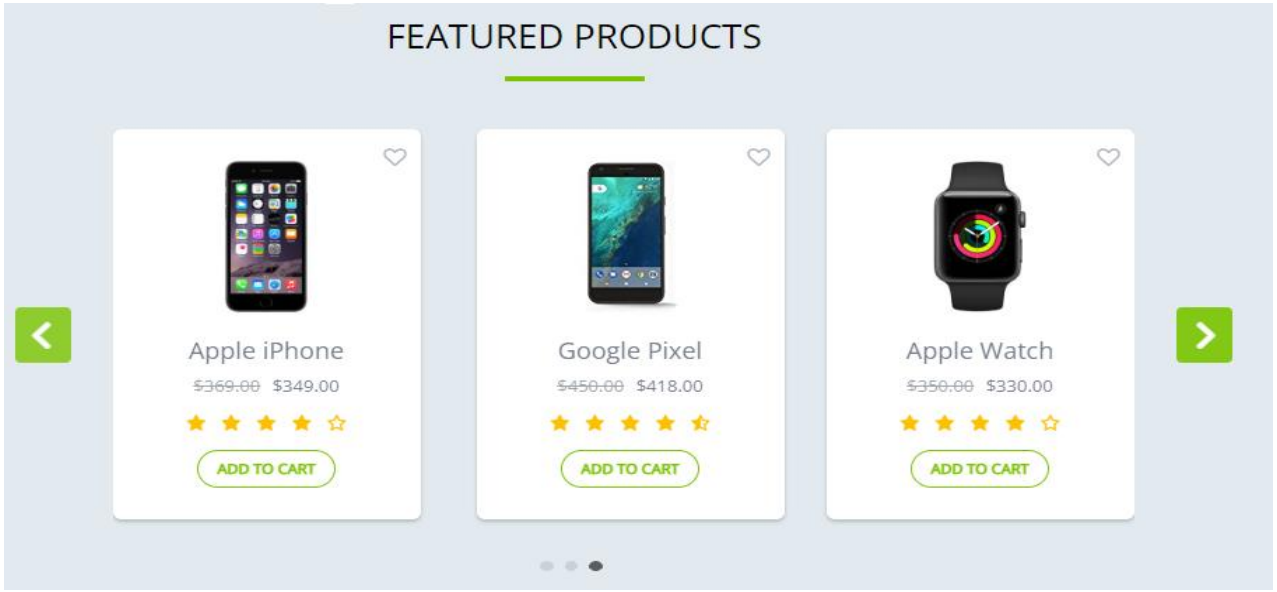
- Bootstrap's Navigation component provides a set of styles and components to create various types of navigation menus on a webpage.




```
<body>
  <nav class="navbar navbar-expand-lg navbar-dark">
    <a class="navbar-brand" href="#">Navbar</a>
    <button
      class="navbar-toggler"
      type="button"
      data-toggle="collapse"
      data-target="#navbarNav"
      aria-controls="navbarNav"
      aria-expanded="false"
      aria-label="Toggle navigation"
    >
      <span class="navbar-toggler-icon"></span>
    </button>
```

Q. What is Bootstrap Carousel component?

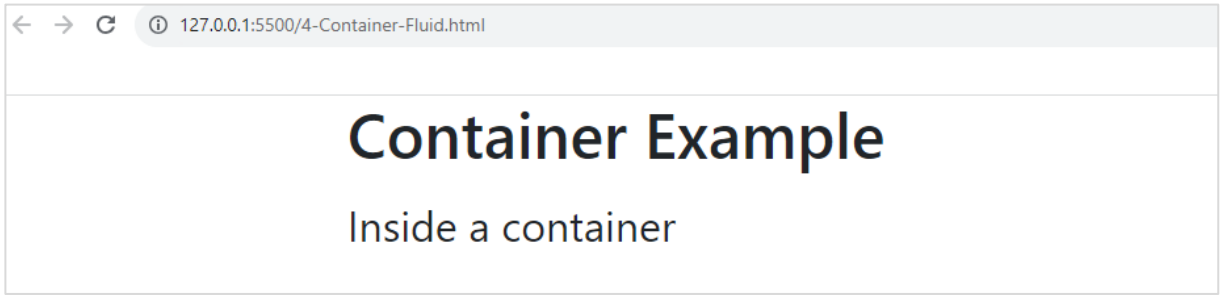
- Bootstrap's carousel component is a slideshow or image slider that allows you to display a series of images or content in a rotating manner.



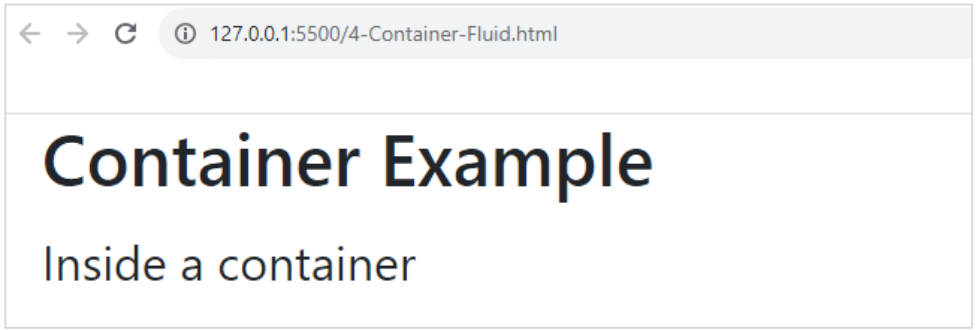

```
<div id="myCarousel" class="carousel slide"
  <!-- Indicators -->
  <ul class="carousel-indicators">
    <li data-target="#myCarousel" data-slide
    <li data-target="#myCarousel" data-slide
    <li data-target="#myCarousel" data-slide
  </ul>

  <!-- Slides -->
  <div class="carousel-inner">
    <div class="carousel-item active">
      <img src="mobile.png" alt="Image 1" />
    </div>
    <div class="carousel-item">
      <img src="screen.png" alt="Image 2" />
    </div>
  </div>
```

Q. Explain the difference between Bootstrap's container and container-fluid?

- .container class **centered the content** and add padding to the sides, ensuring that the content remains within a specified width.
- .container-fluid class provides a **full-width layout**, with the content extending to the edges of the screen.




```
<body>
  <div class="container">
    <h4>Container Example</h4>
    <p>Inside a container</p>
  </div>
</body>
```




```
<body>
  <div class="container-fluid">
    <h4>Container Example</h4>
    <p>Inside a container</p>
  </div>
</body>
```


Your fate is in your hands, not in the hands of your job interviewer.
Good Luck.