



Course Levels

- Level 0 Setup & Fundamentals
- Level 1 HTML Basics
- Level 2 Must-Use HTML Tags
- Level 3 Browser Tools
- Level 4 HTML and Project Structure

- Level 5 List, Tables & Forms
- Level Bonus Github Pages & CodeSpace

Level 0

Setup & Fundamentals

1. IDE or Code Editor
 1. What is IDE
 2. Need of IDE
 3. IDE Selection

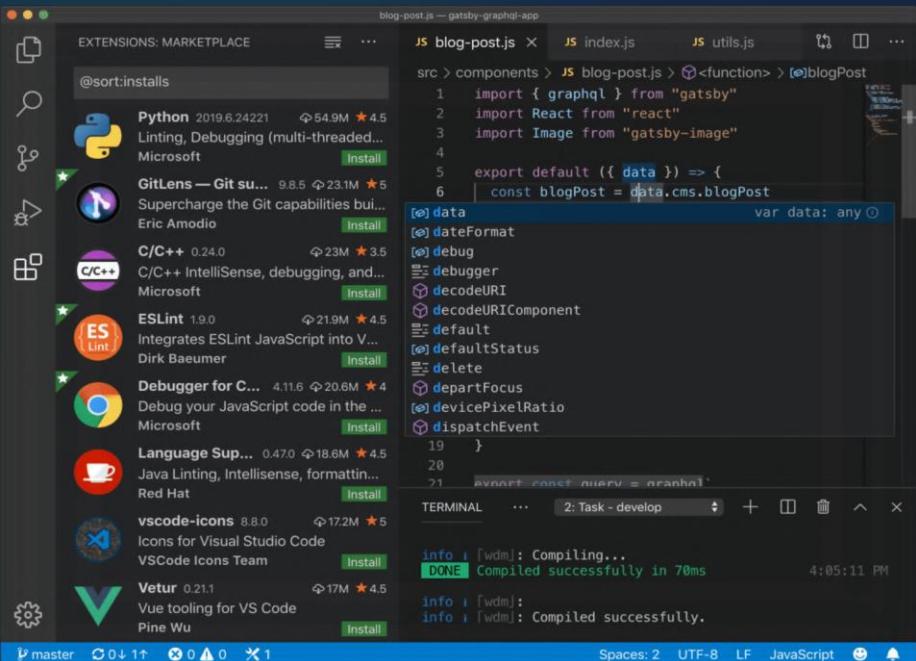
FF5F1F

- 4. Installation and Setup
- 5. VsCode Extensions
- 2. Website Components and Fundamentals
 - 1. Client Side vs Server Side
 - 2. FrontEnd / BackEnd / FullStack
 - 3. Role of Browser
 - 4. HTML
 - 5. CSS
 - 6. JS

Level 0

Setup & Fundamentals

1.1. IDE OR *Code Editor*



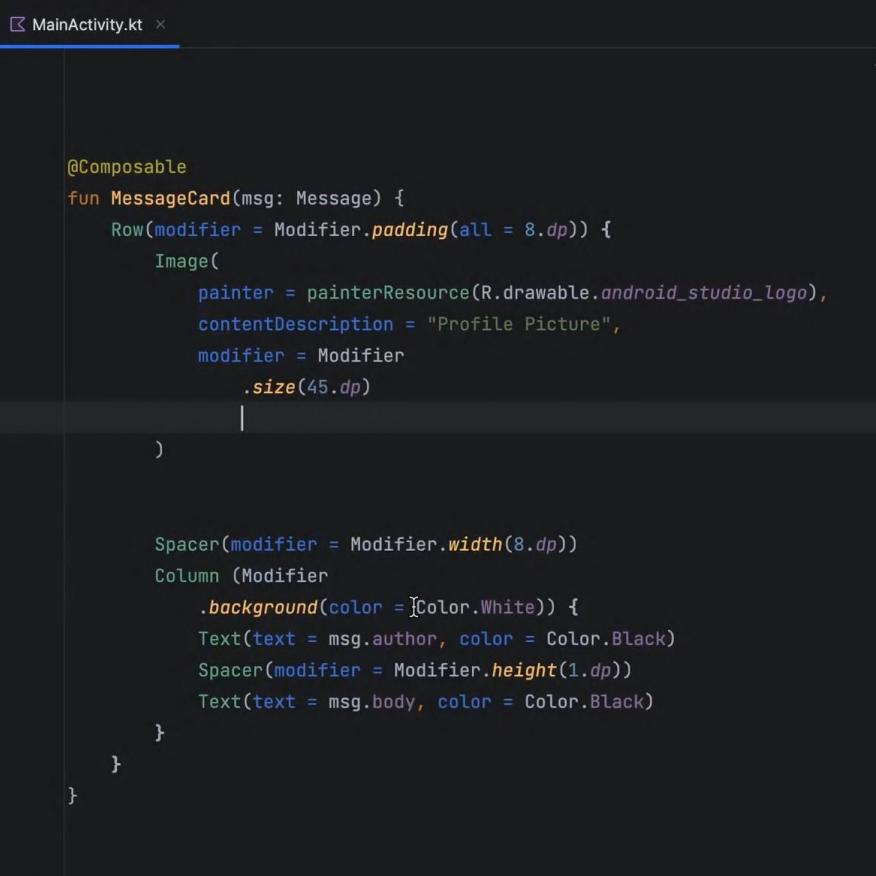
FF5F1F

1.1 What is IDE

1. IDE stands for Integrated Development Environment.
2. Software suite that consolidates basic tools required for software development.
3. Central hub for coding, finding problems, and testing.



1. Streamlines development.



The image shows a screenshot of an Android Studio code editor. The tab bar at the top has 'MainActivity.kt' selected. The code in the editor is a Kotlin function named 'MessageCard'. It uses Jetpack Compose's API to create a UI component. The code includes imports for 'Row', 'Image', 'Spacer', 'Column', and 'Text' from 'androidx.compose.ui' and 'Modifier' from 'androidx.compose.ui.layout'. The 'Image' component is set to have a painter resource of 'R.drawable.android_studio_logo', a content description of 'Profile Picture', and a size of 45.dp. The 'Column' contains a 'Text' component with the author's name and a 'Text' component with the message body, both in black color. The entire function is enclosed in a 'fun' block with a parameter 'msg: Message'.

```
@Composable
fun MessageCard(msg: Message) {
    Row(modifier = Modifier.padding(all = 8.dp)) {
        Image(
            painter = painterResource(R.drawable.android_studio_logo),
            contentDescription = "Profile Picture",
            modifier = Modifier
                .size(45.dp)
        )
        Spacer(modifier = Modifier.width(8.dp))
        Column (Modifier
            .background(color = Color.White)) {
            Text(text = msg.author, color = Color.Black)
            Spacer(modifier = Modifier.height(1.dp))
            Text(text = msg.body, color = Color.Black)
        }
    }
}
```

2. Increases productivity.
3. Simplifies complex tasks.
4. Offers a unified workspace.
5. IDE Features
 1. Code Autocomplete
 2. Syntax Highlighting
 3. Version Control
 4. Error Checking



1.3 IDE Selection

FF5F1F

1. Sublime Text

2. Atom

3. VS Code



A screenshot of the Visual Studio Code interface. The left sidebar shows the 'EXTENSIONS: MARKETPLACE' section with various extensions listed, such as Python, GitLens, C/C++, ESLint, Debugger for Chrome, Language Support for Java, vscode-icons, and Vetur. The main code editor area displays a file named 'blog-post.js' with some JavaScript code. The bottom status bar shows the file path 'src > components > JS blog-post.js', the line number '19', and the terminal output: 'info i [wdm]: Compiling...', 'DONE Compiled successfully in 70ms', and 'info i [wdm]: Compiled successfully.' The status bar also indicates 'Spaces: 2' and 'JavaScript'.

4. Github CodeSpaces



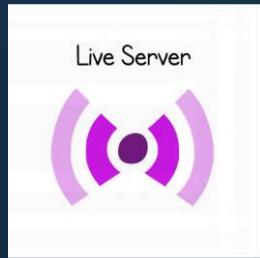
1.4 Installation & Setup

1. Search VS Code



1.5 VsCode Extensions

1. Live Server
2. Prettier



Level 0

Setup & Fundamentals

FF5F1F

2. Website Components And Fundamentals

2.1 Client Side vs Server Side

	Client Side	Server Side
Execution Location	Executes on user's device.	Executes on a remote machine.
Languages	Primarily JavaScript, HTML, CSS.	PHP, Python, Java, Node.js, etc.

2.2 FrontEnd / BackEnd / FullStack



**Client Side / Front-End
Web Development**



**Server Side
Back-End**



M E R N



express



React



Full Stack

2.3 Role of Browser



1. **Displays Web Page:** Turns HTML code into what you see on screen.
2. **User Clicks:** Helps you interact with the web page.

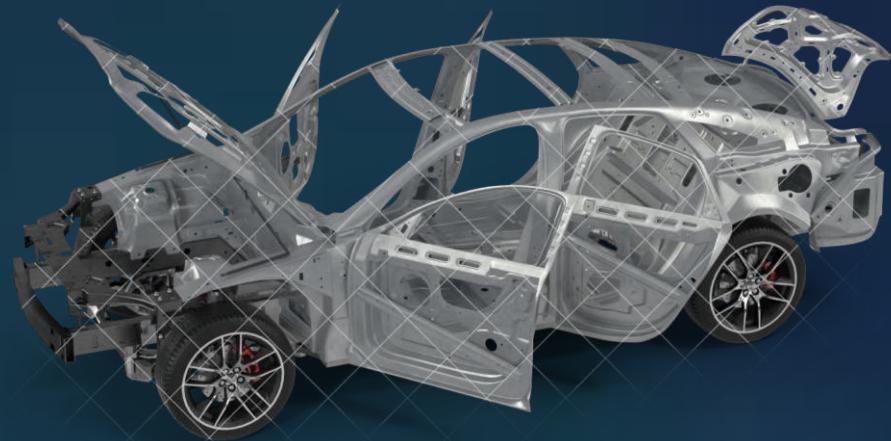
3. **Updates Content:** Allows changes to the page using **JavaScript**.
4. **Loads Files:** Gets **HTML**, images, etc., from the server.



2.4 HTML

(Hypertext Markup Language)

- 1. Structure:** Sets up the layout.
- 2. Content:** Adds text, images, links.
- 3. Tags:** Uses elements like `<p>`, `<a>`.
- 4. Hierarchy:** Organizes elements in a tree.





2.5 CSS

(Cascading Style Sheets)

1. **Style:** Sets the look and feel.
2. **Colors & Fonts:** Customizes text and background.
3. **Layout:** Controls position and size.



4. Selectors: Targets specific HTML elements.



2.6 JS

(Java Script)

1. **JavaScript has nothing to do with Java**
2. **Actions: Enables interactivity.**
3. **Updates: Alters page without reloading.**



4. **Events: Responds to user actions.**
5. **Data: Fetches and sends info to server.**

- 1. IDE or Code Editor**
 - 1. What is IDE**
 - 2. Need of IDE**
 - 3. IDE Selection**
 - 4. Installation and Setup**
 - 5. VsCode Extensions**
- 2. Website Components and Fundamentals**
 - 1. Client Side vs Server Side**
 - 2. FrontEnd / BackEnd / FullStack**
 - 3. Role of Browser**
 - 4. HTML**
 - 5. CSS**
 - 6. JS**



Level 1

HTML Basics

1. Starting up
 1. First File using Text Editor
 2. File Extension
 3. Opening the project in VsCode
 4. Index.html
2. Basics of HTML
 1. What are Tags
 2. Using Emmet ! to generate code
 3. Basic HTML Page

4. **MDN Documentation**
5. **Comments**
6. **Case Sensitivity**

Level 1

HTML Basics

FF5F1F



1.1. Starting Up

1.1 First file using Text Editor

1. Create a folder with name First Project on your Desktop.
2. Open Notepad.
3. Create a file and save it as index.html
4. Copy Sample code
5. Open Browser and Check.



1.2 File Extension

HTML

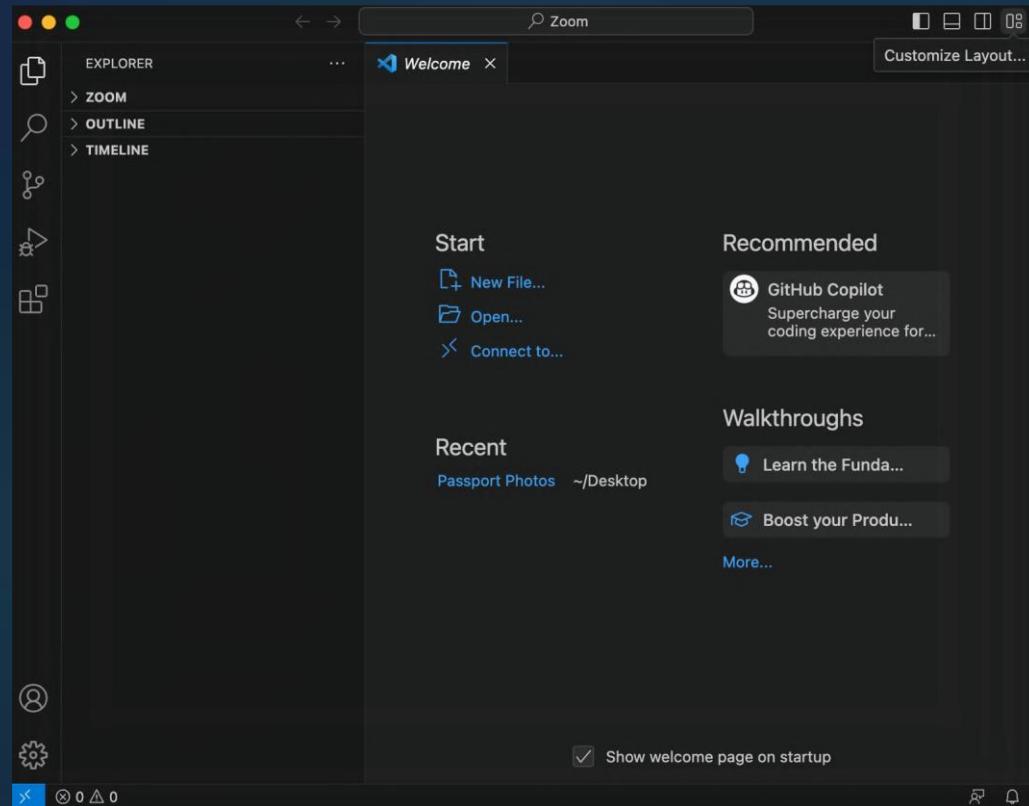
1. Most commonly used.
2. Works across all browsers.
3. Widely recognized and supported.
4. Typically saved as .html.



HTM

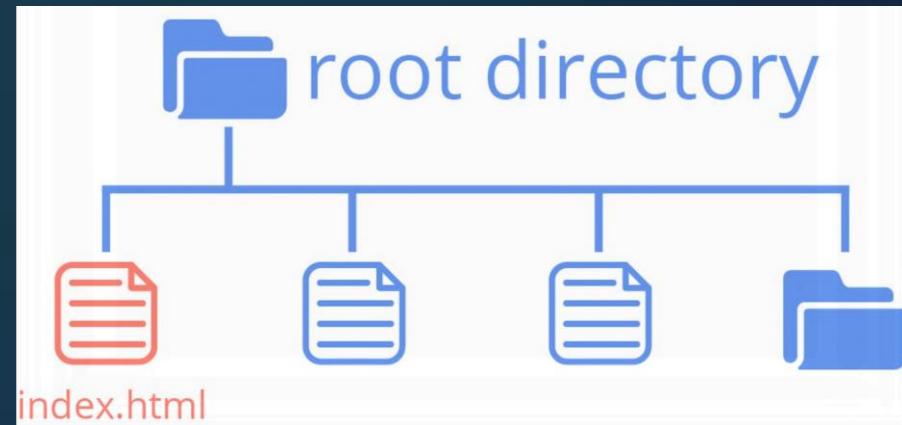
1. Less commonly used.

2. **Originated for compatibility with older systems.**
3. **Works same as .html.**
4. **Typically saved as .htm.**



1.4 Importance of index.html

1. Default name of a website's homepage.
2. First page users see when visiting a website
3. Important for SEO (Search Engine Optimization)
4. Provides uniform starting point across servers



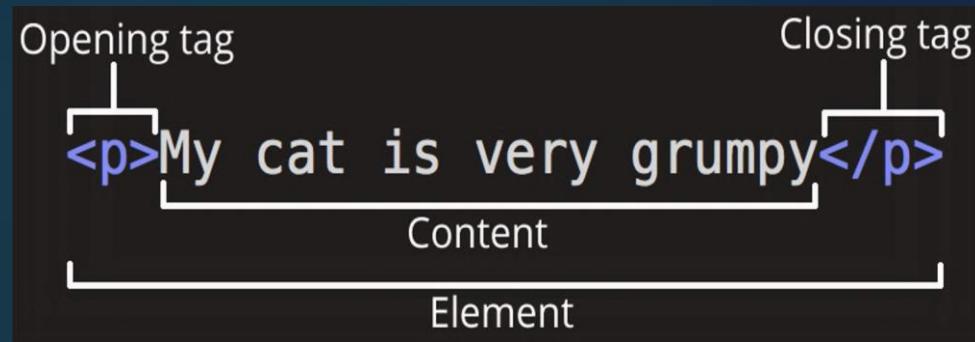
1.2.

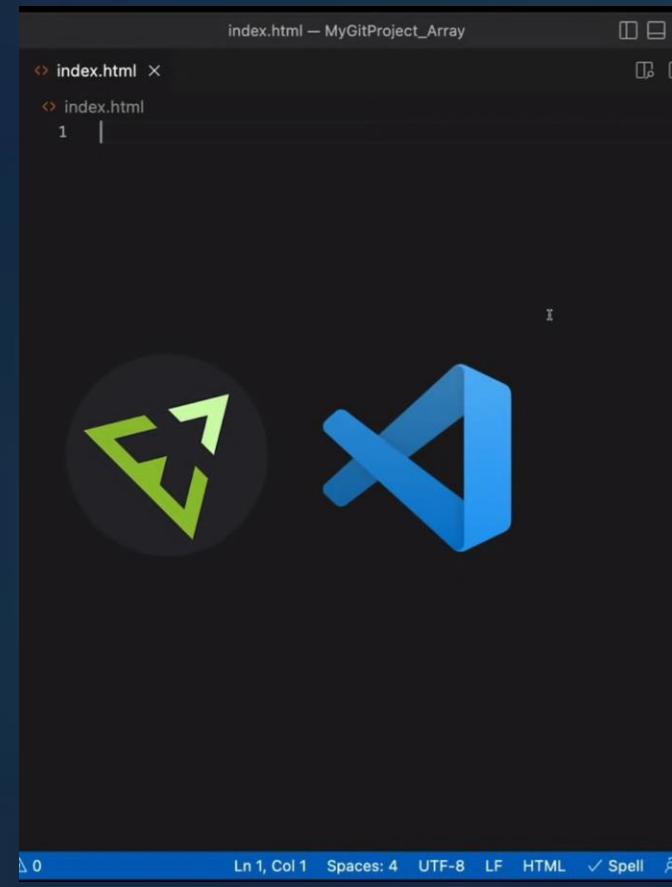


Basics of HTML

2.1 What are Tags

1. Elements that are used to create a website are called HTML Tags.
2. Tags can contain content or other HTML tags.
3. Define elements like text, images, links





1. Type ! and wait for suggestions.

2.3 Basic HTML Page

```
html lang="en">          Parent of all HTML tags / Root element
```

```
<!DOCTYPE html>      Defines the HTML Version
```

```
<
```

```
<head>
```

Parent of
meta data

```
title>My First Webpage</title> Title of the web page
```

tags

```
<  
  </head>  
  Parent of content  
  <body>          tags  
    <h1>Hello World!</h1> Heading tag</body>  
</html>
```

2.4 MDN Documentation

1. Visit <https://developer.mozilla.org/>
2. Official resource for HTML
3. Offers comprehensive guides and tutorials
4. Includes examples for real-world use
5. Updated with latest HTML features
6. Trusted by developers worldwide

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      The body element represents the content of the
7      document.
8  <MDN Reference
9  <body>
10
11 </body>
12 </html>
```



2.5 Comments

1. Used to add notes in HTML code
2. Not displayed on the web page
3. Syntax: <!-- Comment here -->
4. Helpful for code organization

Writing comments in HTML

Single-line Comment

```
1 <!--This is a single line  
comment in HTML. You cannot  
see it on a webpage. Click  
on view-source to see a  
message I left just for you.  
-->
```

1 <!-- This is a multi-line
comment in HTML.

2 You cannot see it on a
webpage.

3 If you view-source on the
browser you can see the
comment there.-->

Multi-line Comment

2.6 Case Sensitivity

1. HTML is case-insensitive for tag names
2. Attribute names are also be case-insensitive
3. Best practice: use lowercase for consistency

```
<html> = <HTML>
<p>      = <P>
<head> = <HEAD>
<body> = <BODY>
```


1. **First File using Text Editor**
2. **File Extension**
3. **Opening the project in VsCode**
4. **Index.html**
2. **Basics of HTML**
 1. **What are Tags**
 2. **Using Emmet ! to generate code**
 3. **Basic HTML Page**
 4. **MDN Documentation**
 5. **Comments**
 6. **Case Sensitivity**



Project Level 1

HTML Basics

1. Create a new project with **Index.html**
2. Generate boilerplate code using **Emmet**



3. Write “I am learning with Prashant sir”
4. Use comments
5. Also use Case insensitive tags

Level 2

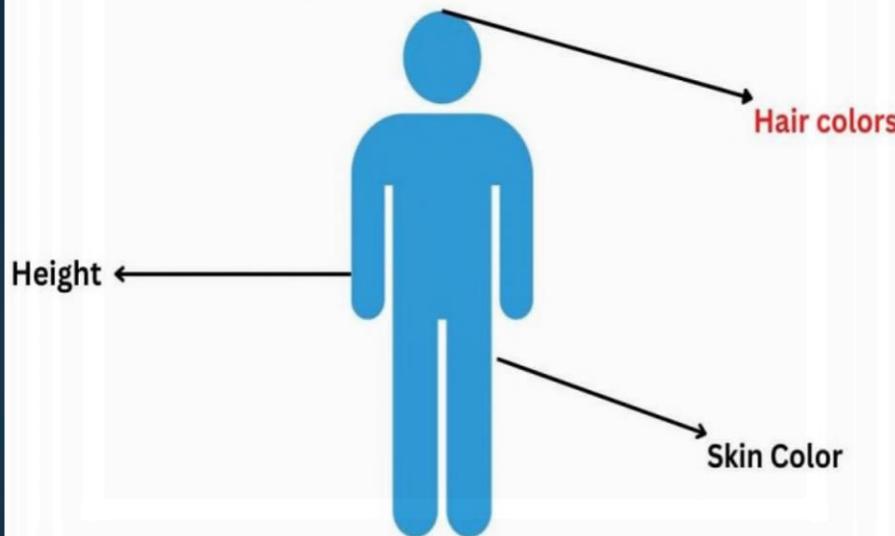
Must-Use HTML Tags

1. HTML Attributes
 1. What are HTML Attributes
 2. Id Property
2. HTML Tags
 1. Heading Tag
 2. Paragraph Tag

3.
 <HR> tags
 4. Image Tag
 5. Video Tag
 6. Anchor Tag
 7. Bold / Italic / Underline / Strikethrough
 8. Pre Tag
 9. Big / Small Tag
 10. Superscript / Subscript
3. Character Entity Reference
1. What are Character Entity References

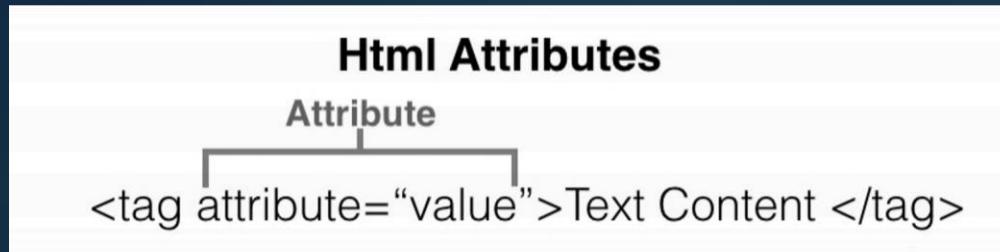
Must **Level-2** Use HTML Tags

Attribute



1.1. HTML Attributes

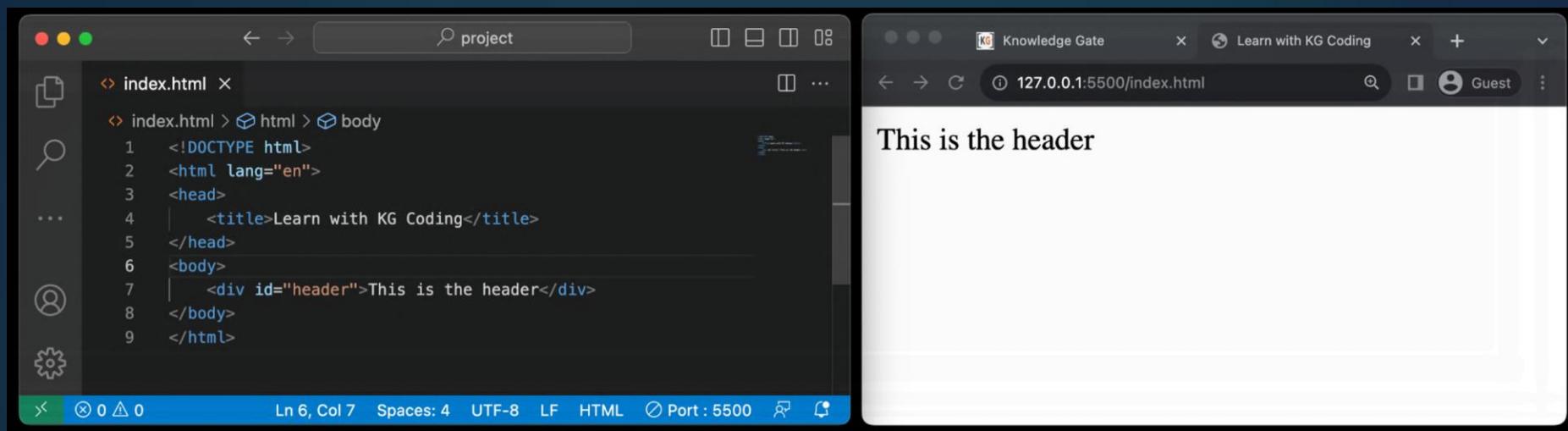
1.1 What are HTML Attributes?



1. Provides additional information about elements
2. Placed within opening tags
3. Common examples: href, src, alt
4. Use name=value format

5. Can be single or multiple per element

1.2 id property



The image shows a code editor on the left and a browser window on the right. The code editor displays the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <div id="header">This is the header</div>
</body>
</html>
```

The browser window shows the rendered output: "This is the header".



1.2.

HTML

Tags

2.1 Heading Tag

The image shows a comparison between a code editor and a web browser. On the left, a code editor window displays the file `index.html` with the following content:

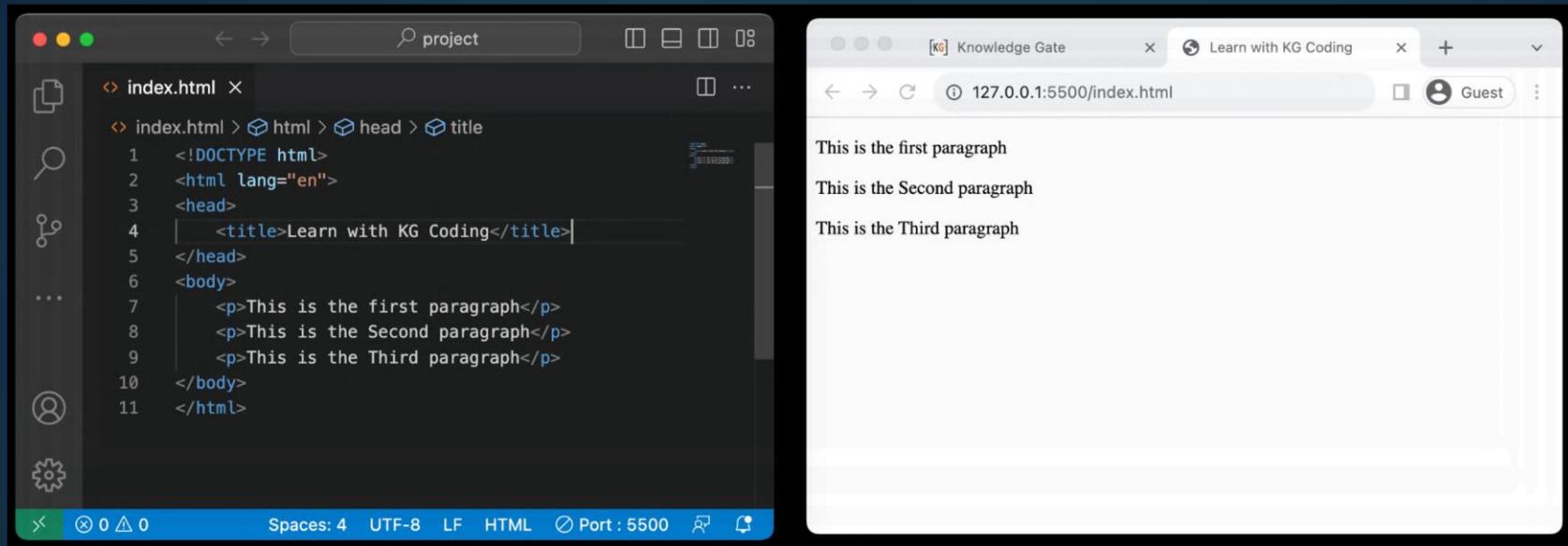
```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <h1>Heading 1</h1>
    <h2>Heading 2</h2>
    <h3>Heading 3</h3>
    <h4>Heading 4</h4>
    <h5>Heading 5</h5>
    <h6>Heading 6</h6>
</body>
</html>
```

On the right, a browser window titled "Knowledge Gate" shows the rendered HTML with the following headings:

Heading 1
Heading 2
Heading 3
Heading 4
Heading 5
Heading 6

1. Defines headings in a document
2. Ranges from <h1> to <h6>
3. <h1> is most important, <h6> is least
4. Important for SEO
5. Helps in structuring content

2.2 Paragraph Tag



The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

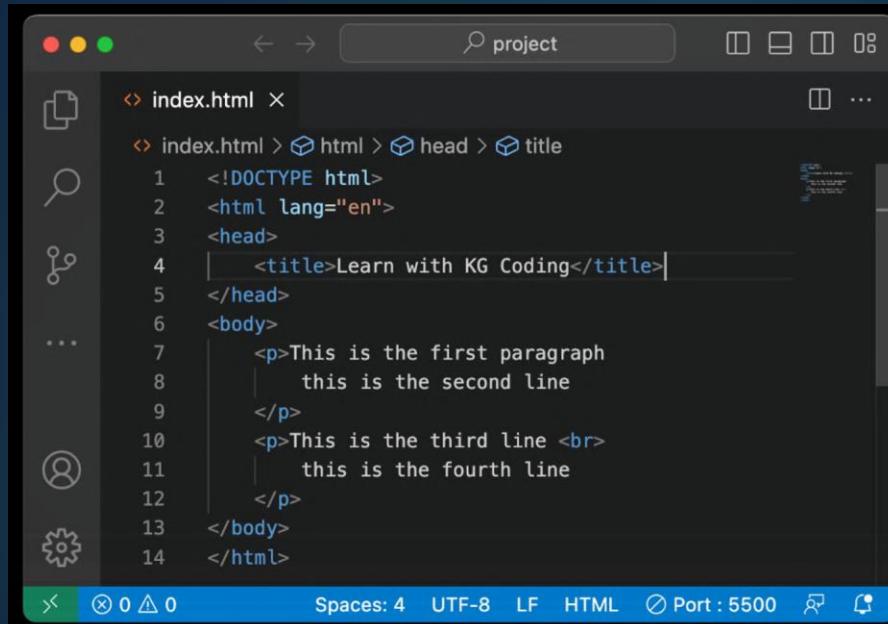
```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <p>This is the first paragraph</p>
    <p>This is the Second paragraph</p>
    <p>This is the Third paragraph</p>
</body>
</html>
```

The browser window shows the rendered HTML with three paragraphs:

This is the first paragraph
This is the Second paragraph
This is the Third paragraph

1. Used for defining paragraphs
2. Enclosed within <p> and </p> tags
3. Adds automatic spacing before and after
4. Text wraps to next line inside tag
5. Common in text-heavy content

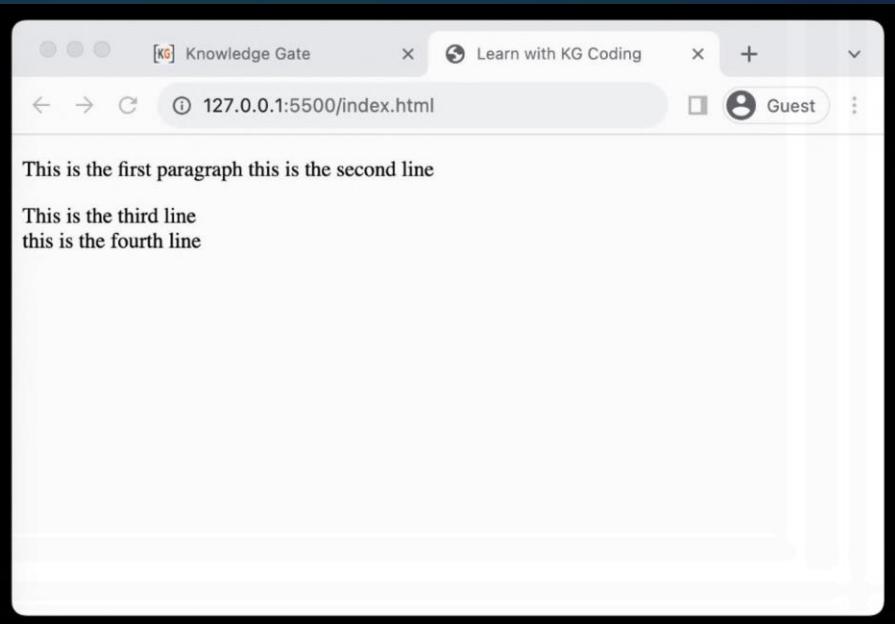
2.3
 Tag



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following code:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>Learn with KG Coding</title>
5  </head>
6  <body>
7  |   <p>This is the first paragraph
8  |       this is the second line
9  |   </p>
10 |   <p>This is the third line <br>
11 |       this is the fourth line
12 |   </p>
13 </body>
14 </html>
```

The code editor has a dark theme with light-colored syntax highlighting. The status bar at the bottom shows "Spaces: 4" and "Port : 5500".

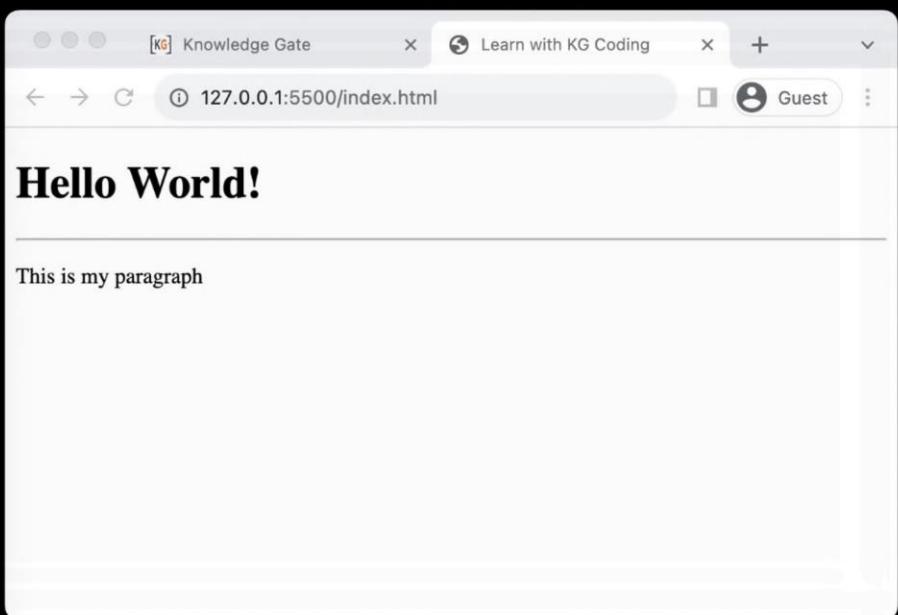


A screenshot of a web browser window titled "Knowledge Gate". The address bar shows "127.0.0.1:5500/index.html". The page content is:

This is the first paragraph
this is the second line

This is the third line
this is the fourth line

1. **
 adds a line break within text**
2. **
 is empty, no closing tag needed**
3. **
 and
 are both valid**



The image shows a split-screen view. On the left is a code editor window titled "index.html" with the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <h1>Hello World!</h1>
    <hr>
    <p>This is my paragraph</p>
</body>
</html>
```

On the right is a browser window titled "Knowledge Gate" showing the rendered HTML. The page displays "Hello World!" in a large font, followed by a horizontal line (hr), and the text "This is my paragraph".

1. **<hr> creates a horizontal rule or line**

2. <hr> also empty, acts as a divider

2.4 Image Tag

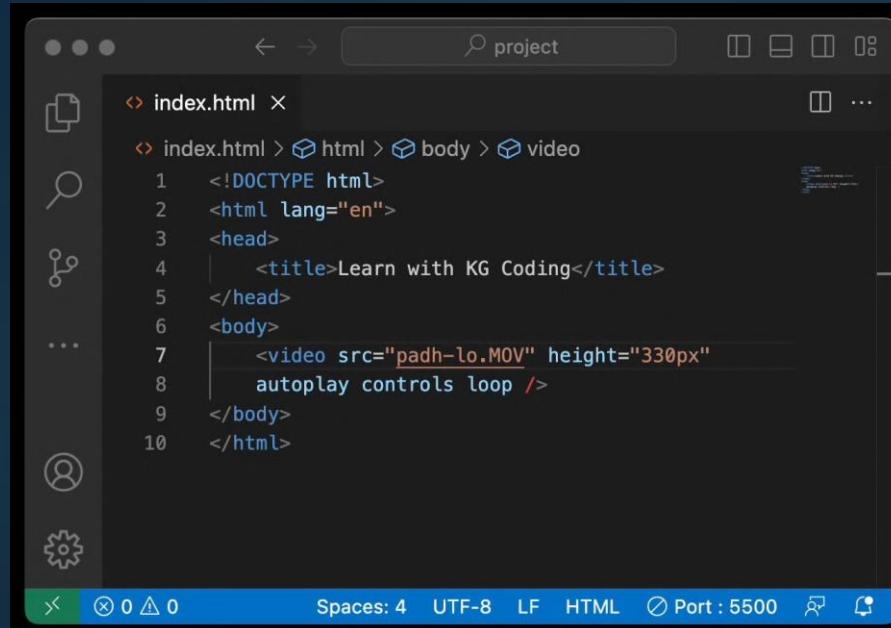
The image shows a code editor on the left and a web browser on the right. The code editor displays the contents of index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <image src="sanchitsir.png"
        alt="Sanctit Sir Passport photo"
        height="300px">
</body>
</html>
```

The browser window shows a portrait of a man with glasses and a blue polo shirt, identified as Sanctit Sir. The browser title bar reads "Knowledge Gate" and the address bar shows "127.0.0.1:5500/index.html".

1. Used to embed images
2. Utilizes the `src` attribute for image URL
3. `alt` attribute for alternative text
4. Can be resized using width and height
5. Self-closing, doesn't require an end tag

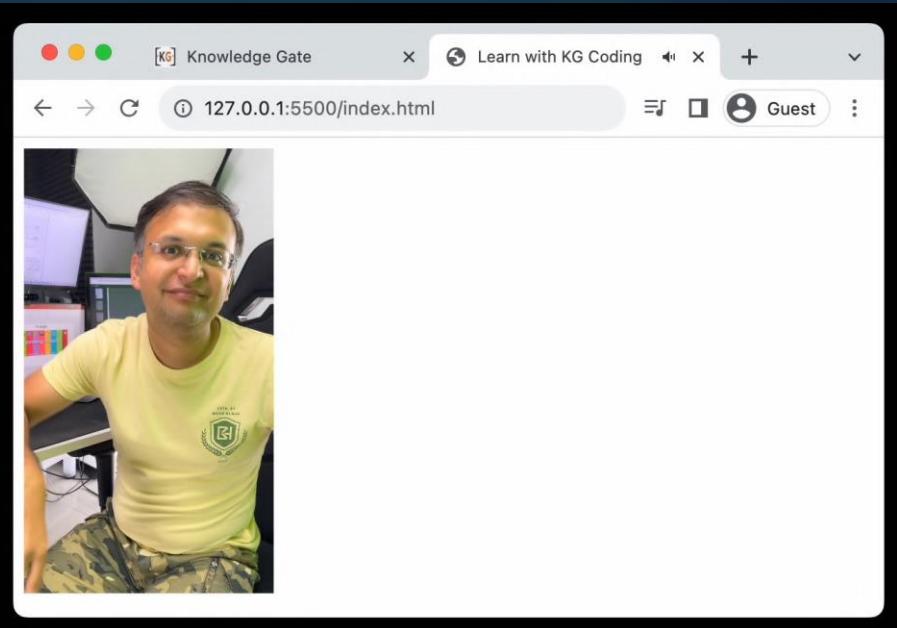
2.5 Video Tag

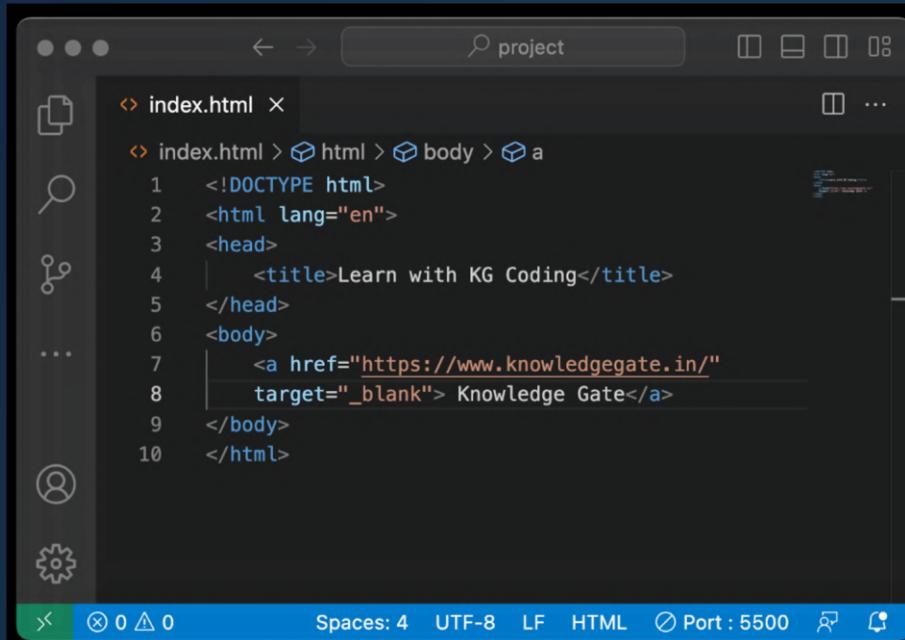


A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <video src="padh-lo.MOV" height="330px"
        autoplay controls loop />
</body>
</html>
```

The code editor has a dark theme with light-colored syntax highlighting. The bottom status bar shows "Spaces: 4" and "Port : 5500".

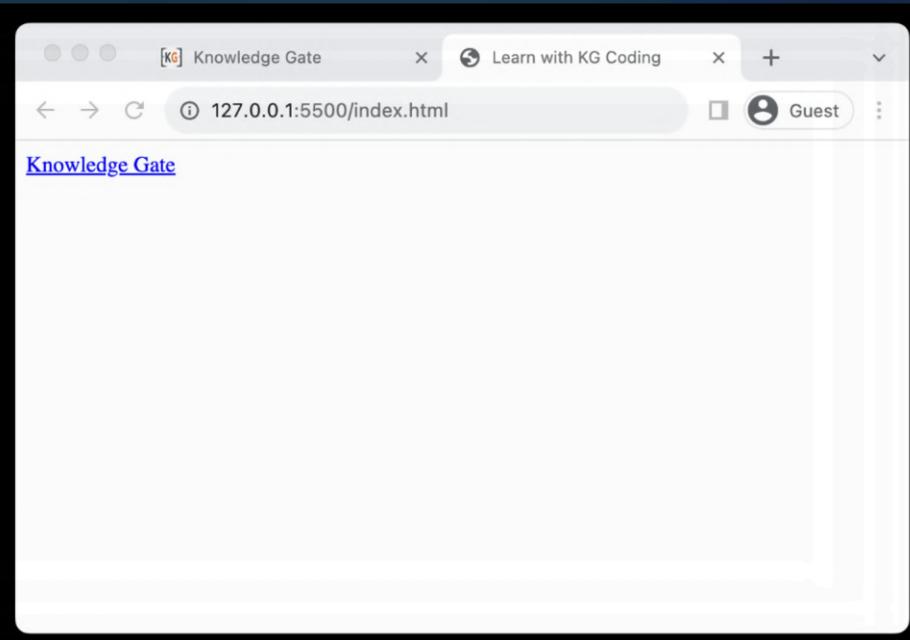




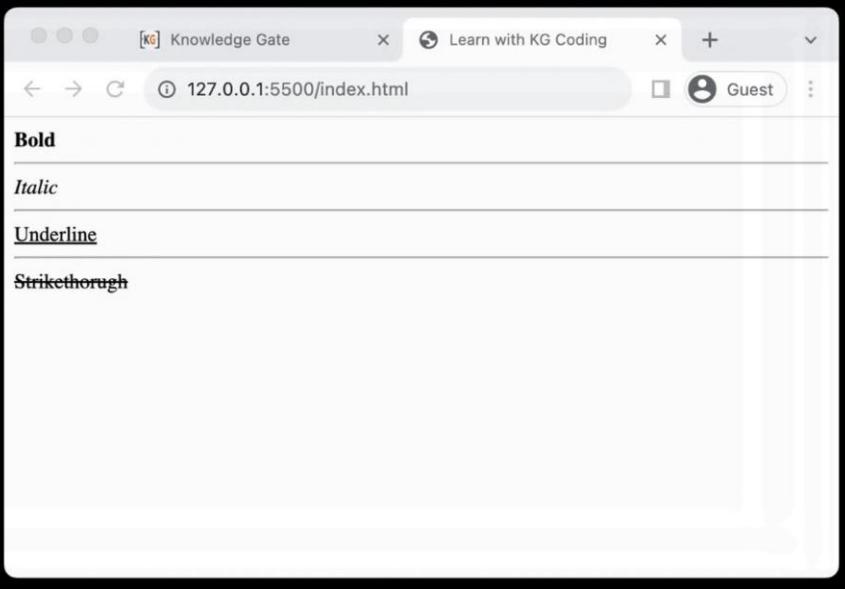
A screenshot of a dark-themed code editor. The title bar says "project". The left sidebar has icons for file, search, and user. The main area shows the file "index.html" with the following code:

```
index.html X
index.html > html > body > a
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>Learn with KG Coding</title>
5  </head>
6  <body>
7  |   <a href="https://www.knowledgegate.in/" target="_blank"> Knowledge Gate</a>
8
9 </body>
10 </html>
```

The status bar at the bottom shows "Spaces: 4" and "Port : 5500".



1. **Used for creating hyperlinks**
2. **Requires href attribute for URL**



The image shows two side-by-side screenshots. On the left is a code editor window titled 'project' containing the file 'index.html'. The code is as follows:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>Learn with KG Coding</title>
5  </head>
6  <body>
7  |   <b>Bold</b> <hr>
8  |   <i>Italic</i> <hr>
9  |   <u>Underline</u> <hr>
10 |   <s>Strikethrough</s>
11 </body>
12 </html>
```

The right screenshot shows a web browser window titled 'Knowledge Gate' with the URL '127.0.0.1:5500/index.html'. The page displays four horizontal lines with text above them: 'Bold', 'Italic', 'Underline', and 'Strikethrough'. Each line is preceded by its corresponding HTML tag: **, *, , and ~~.~~***

1. ** makes text bold**
2. **<i> makes text italic**
3. **<u> underlines text**

4. **<s> or <strike> applies strikethrough**
5. **Primarily used for text styling and emphasis**

2.8 Pre Tag

The image shows a code editor on the left and a web browser on the right, demonstrating the use of the `<pre>` tag.

Code Editor (VS Code):

- Project name: project
- File: index.html
- Code structure:
 - index.html > html > body > pre
 - Content of the pre tag:

```
first line
more spaces
<pre>
    first line
    more spaces
</pre>
```
- Code content:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>Learn with KG Coding</title>
5  </head>
6  <body>
7  |   first line
8  |   more   spaces
9  |   <pre>
10 |     first line
11 |     more   spaces
12 |   </pre>
13 </body>
14 </html>
```
- Bottom status bar: Spaces: 4, UTF-8, LF, HTML, Port: 5500

Browser:

- Title: Learn with KG Coding
- URL: 127.0.0.1:5500/index.html
- User: Guest
- Content:

first line more spaces

first line
more spaces

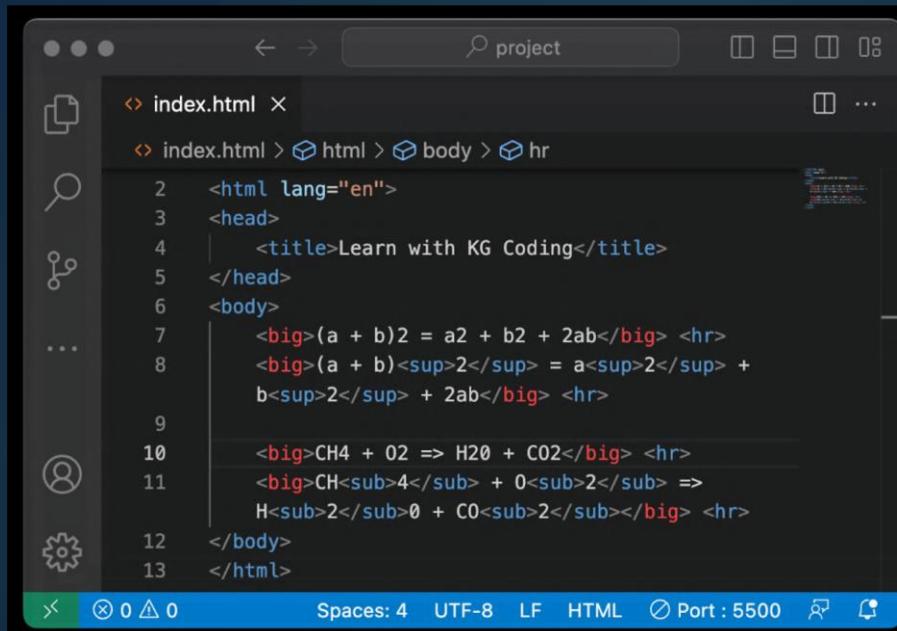
The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <big>This is Big</big> <hr>
    <small>This is Small</small>
</body>
</html>
```

The browser window shows the rendered HTML. The text "This is Big" is displayed in a large font size, indicating it is styled by the `<big>` tag. Below it, there is a horizontal line (`<hr>`) and the text "This is Small" in a smaller font size, indicating it is styled by the `<small>` tag.

1. **<big> increases text size**
2. **<small> decreases text size**

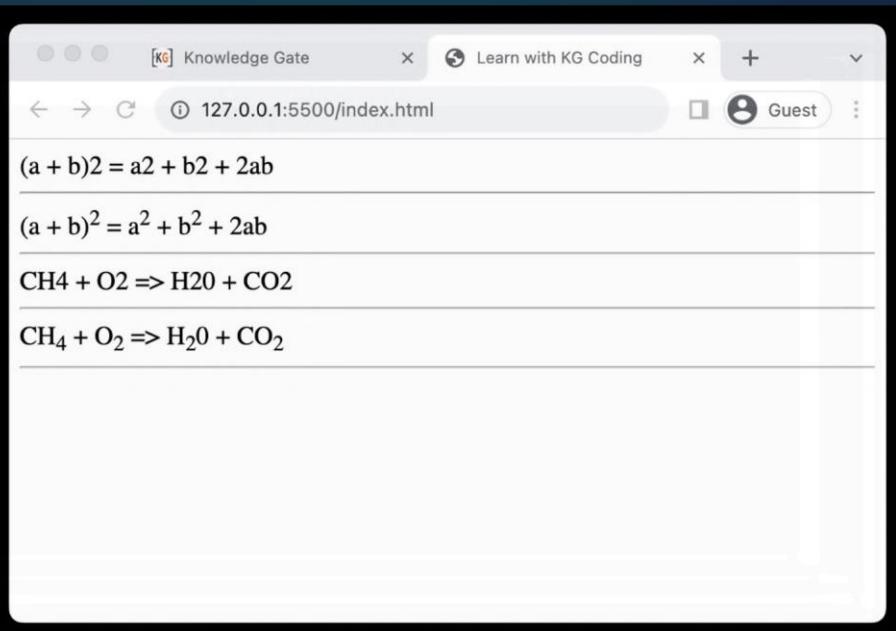
2.10 Superscript/Subscript Tag



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following code:

```
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <big>(a + b)2 = a2 + b2 + 2ab</big> <hr>
    <big>(a + b)2 = a2 + b2 + 2ab</big> +
    b2 + 2ab</big> <hr>
    <big>CH4 + O2 => H2O + CO2</big> <hr>
    <big>CH4 + O2 => H2O + CO2</big>
</body>
</html>
```

The code uses the `<big>` tag to wrap mathematical expressions and chemical equations, which are displayed with superscripts and subscripts.



A screenshot of a web browser window titled "Knowledge Gate". The address bar shows "127.0.0.1:5500/index.html". The page content displays the rendered output of the `index.html` file:

(a + b)² = a² + b² + 2ab

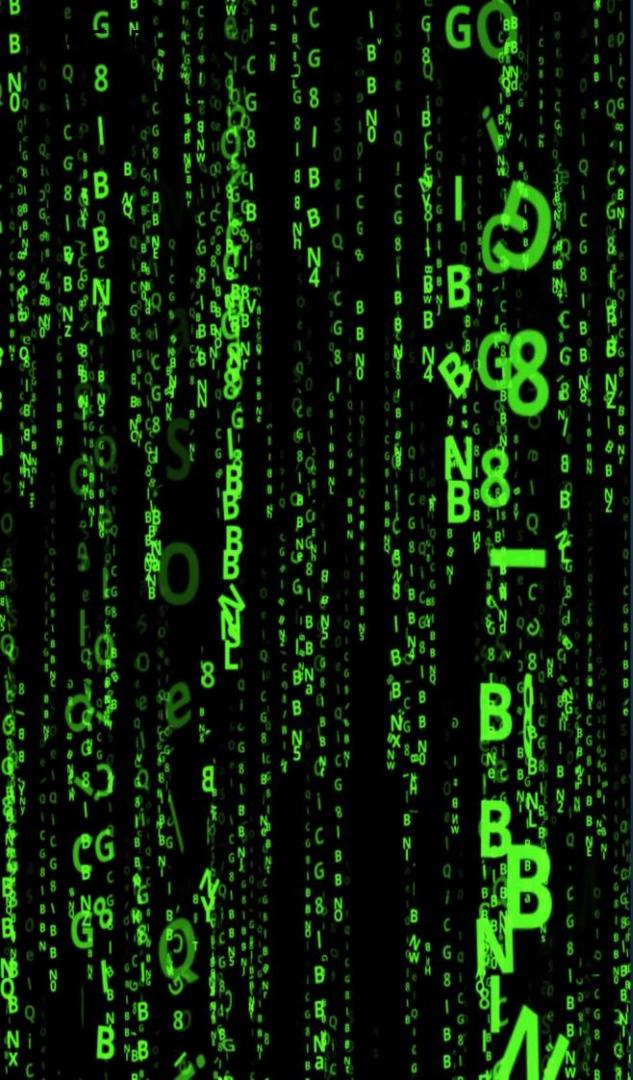
(a + b)² = a² + b² + 2ab

CH₄ + O₂ => H₂O + CO₂

CH₄ + O₂ => H₂O + CO₂

The browser shows two identical renditions of the same content, likely due to a caching issue or a bug in the browser's rendering of the `<big>` tag.

1.3.Character Entity Reference



3.1 Character Entity

Reference

1. Used to display reserved or special characters
2. Syntax often starts with & and ends with ; (e.g., & for &)

	 	-	–	-	−	\circ	°	Δ	Δ	α	&al;
ϵ	€	—	—	\pm	±	\circ	º	Λ	Λ	β	&be;
ϕ	¢	...	…	$\sqrt{}$	√	a	ª	Θ	Θ	γ	&ga;
\pounds	£	\S	§	∞	∞	1	¹	Ξ	Ξ	δ	&de;
\yen	¥	¶	¶	∞	∝	2	²	Π	Π	ε	&ep;
\circ	¤	†	†	\times	×	3	³	Σ	Σ	ζ	&ze;
f	ƒ	‡	‡	\div	÷	$\frac{1}{4}$	¼	Φ	Φ	η	&et;
\circledC	©		¡	\sim	∼	$\frac{1}{2}$	½	Ψ	Ψ	θ	&th;
\circledR	®	⌚	¿	\approx	≈	$\frac{3}{4}$	¾	Ω	Ω	\imath	&io;
TM	™	%	‰	\cong	≅	\therefore	∴	∇	∇	κ	κ

1. **HTML Attributes**
 1. **What are HTML Attributes**
 2. **Id Property**
2. **HTML Tags**
 1. **Heading Tag**
 2. **Paragraph Tag**
 3. **
 <HR> tags**
 4. **Image Tag**
 5. **Video Tag**
 6. **Anchor Tag**
 7. **Bold / Italic / Underline / Strikethrough**
 8. **Pre Tag**
 9. **Big / Small Tag**
 10. **Superscript / Subscript**
3. **Character Entity Reference**



1. What are Character Entity References

Project Level 2

Must-Use HTML Tags

1. Create a page with heading, paragraph, line breaks and separators.
2. Use an image with height 300, which is a link to another page.



3. **Use bold, italic, underline and strike through in one line.**
4. **Write third equation of motion using superscript and subscript.**

Level 3

Browser Tools

1. **Browser Tools**
 1. **View Page Source**
 2. **Inspect Element**
 3. **HTML without CSS**
2. **Responsive Design**

1. Different screen size
3. Live Edit Code
 1. Live edit HTML
 2. Live edit CSS
 3. Live edit JS
 4. Changes only happening at client
4. Validating Web pages
 1. Using validator.w3.org

Level 3

Browser Tools

1.1. Browser Tools



HTML and CSS

Back

Forward

Reload

Save As...

Print...

Cast...

Search Images with Google

Send to Your Devices

Create QR code for this page

Translate to English

 AdBlock — best ad blocker >

 Redux DevTools >

View Page Source

Inspect

FF5F1F

Level 3

2. Useful for debugging and learning
3. Shows external files like JavaScript links

Level 3

FF5F1F

editing of

Back

Forward

Reload

Save As...

Print...

Cast...

Search Images with Google

Send to Your Devices

Create QR code for this page

Translate to English

 AdBlock — best ad blocker >

 Redux DevTools >

View Page Source

Inspect

HTML/CSS

2. Useful for debugging and testing
3. Shows element hierarchy and layout
4. Includes console for JavaScript
5. Highlights selected elements on page

Browser Tools

1.2.

FF5F1F

Level 3



Responsiv

FF5F1F

Level 3

2.1 Different Screen Sizes

Responsive

- iPhone SE
- iPhone XR
- ✓ iPhone 12 Pro
- Pixel 5
- Samsung Galaxy S8+
- Samsung Galaxy S20 Ultra
- iPad Air
- iPad Mini
- Surface Pro 7
- Surface Duo
- Galaxy Fold
- Samsung Galaxy A51/71
- Nest Hub
- Nest Hub Max
- Facebook for Android v407 on Pixel 6

Edit...

Different screen sizes

FF5F1F

Level 3

2. **Flexible layouts**
3. **Optimizes images and assets**
4. **Enhances user experience on mobile and desktop**

Browser Tools

1.3. Live Edit



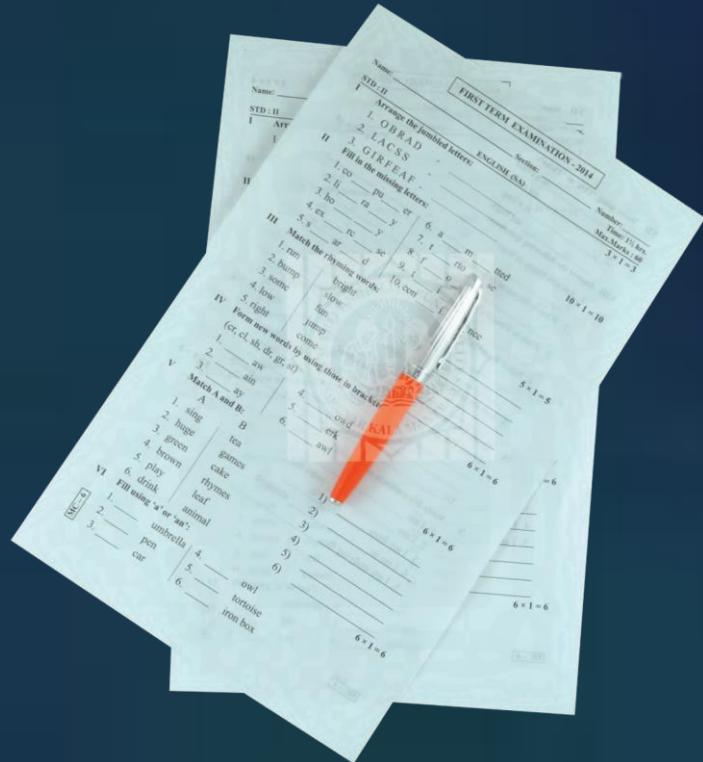
FF5F1F

Level 3

Code

3.4 Changes happening at Client

1. Changes made are temporary



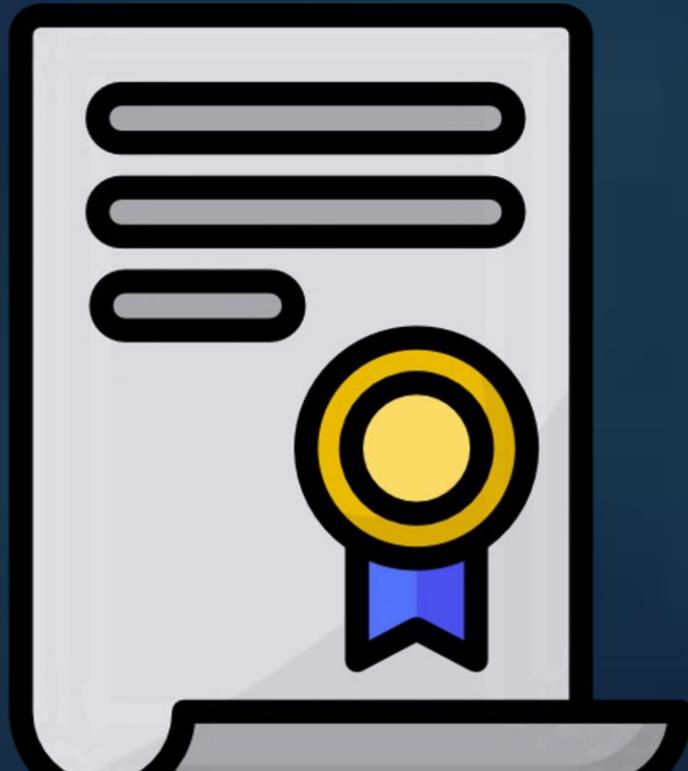
Level 3

2. Affect only the current session
3. Not saved to the server
4. Reset upon page reload
5. Useful for testing, not permanent fixes

Like: If you change the question in your question paper that has no effect on actual exam.

FF5F1F

Level 3



1.4. Validating

FF5F1F

Level 3

Nu Html Checker

This tool is an ongoing experiment in better HTML checking, and its behavior remains subject to change

Showing results for contents of text-input area

Checker Input

Show source outline image report Options...

Check by css

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>My First Webpage</title>
</head>
<body>
  <h1>Hello World!</h1>
</body>
<html>
```

Check

Use the Message Filtering button below to hide/show particular messages, and to see total counts of errors and warnings.

Message Filtering

1. Error Stray start tag `html`.

to standards

2. Minimizes cross-browser issues
3. Helps in achieving better SEO results
4. Easier to debug and maintain
5. Optimizes performance by reducing parsing errors

1. **Browser Tools**
 1. View Page Source
 2. Inspect Element
 3. HTML without CSS
2. **Responsive Design**
 1. Different screen size
3. **Live Edit Code**
 1. Live edit HTML
 2. Live edit CSS
 3. Live edit JS
 4. Changes only happening at client
4. **Validating Web pages**
 1. Using validator.w3.org



Project Level 3

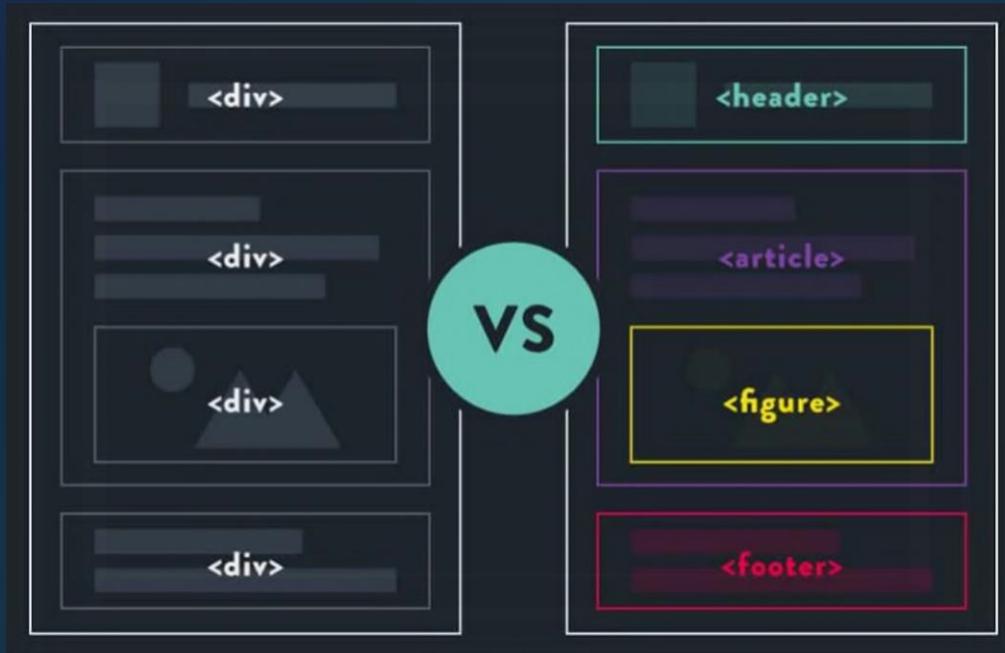
Browser Tools

1. **Save Source of Instagram in a file and check the render.**
2. **Inspect the likes element on the page and read the code to understand.**



- 3. **Aside Tag**
- 3. **Footer Tag**
- 3. **Folder Structure**
 - 1. Recommended Folder structure
- 4. **More Tags**
 - 1. Navigation tags
 - 2. Block / Inline Elements
 - 3. Div tags
 - 4. Span Tags

HTML and Project **Level 4** Structure



1.1.

Semantic Tags

1.1 Semantic/Non-Semantic Tags

Semantic Tags

- **Meaningful:** Describe content.
- **SEO:** Good for search engines.

- **Accessibility:** Useful for screen readers.
- **Examples:** <header>, <footer>, <article>, <section>, <nav>.

Non-Semantic Tags

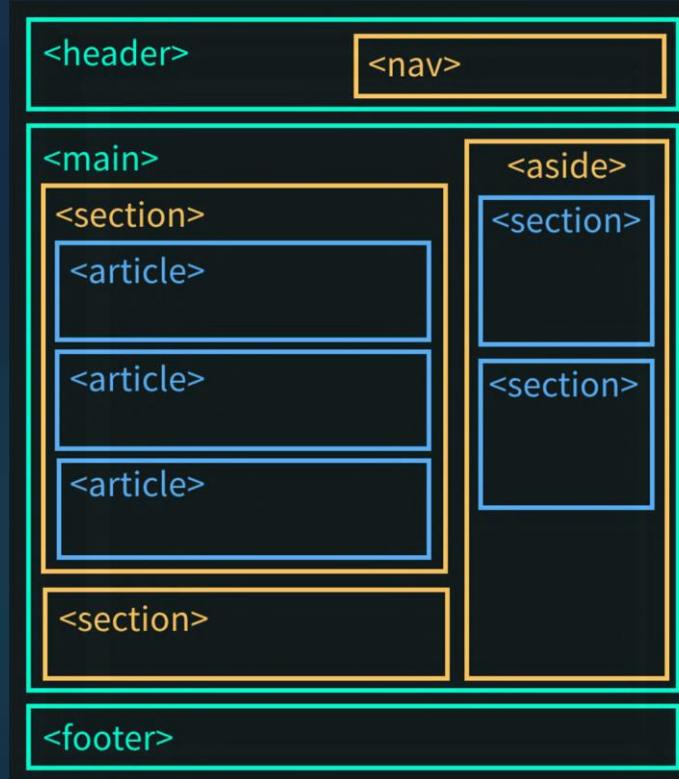
- Generic: No specific meaning.

- For Styling: Used for layout.
 - No SEO: Not SEO-friendly. •
- Examples: <div>, , <i>, .

HTML and Project Level Structure 4

FF5F1F

1.2. Body



The image shows a split-screen view. On the left is a code editor window titled 'index.html' with a file icon. The code editor displays the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <header><big>This is the Header</big></header>
</body>
</html>
```

The code editor has a dark theme with light-colored text. On the right is a web browser window titled 'Knowledge Gate' with a user icon and 'Guest'. The browser's address bar shows '127.0.0.1:5500/index.html'. The page content is a large, bold black text: 'This is the Header'.

1. **Purpose:** Used to contain introductory content or navigation links.
2. **Semantic:** It's a semantic tag, providing meaning to the enclosed content.

2.2 Main Tag

The image shows a code editor on the left and a browser window on the right. The code editor displays the file `index.html` with the following content:

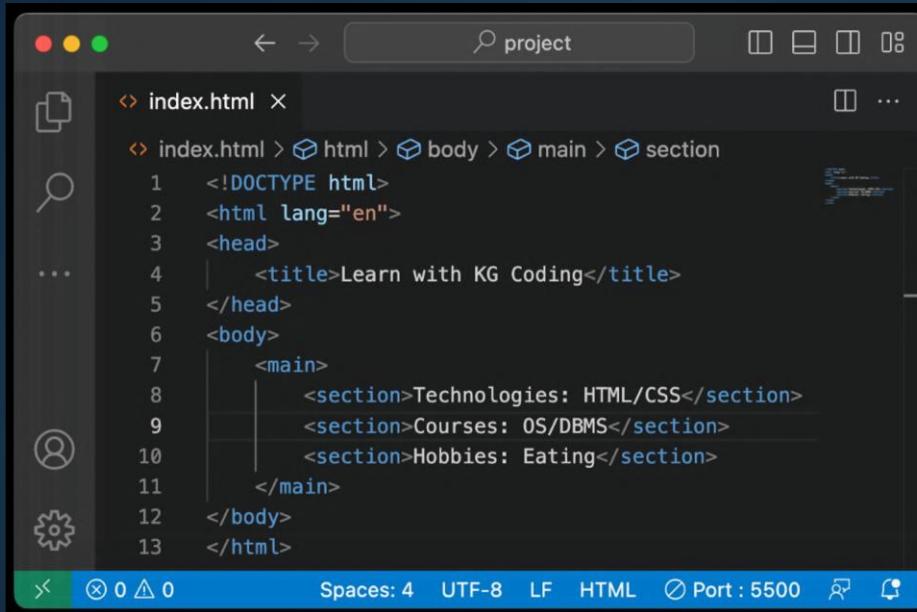
```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <header><b>This is the Header</b></header>
    <hr><main>This is main space</main>
</body>
</html>
```

The browser window shows the rendered HTML. The header section contains the text "This is the Header" in bold. Below it is a horizontal line (`<hr>`). The main content section contains the text "This is main space".

1. Purpose: Encloses the primary content of a webpage.

2. **Semantic:** Adds meaning, indicating the main content area.
3. **Unique:** Should appear only once per page.
4. **Accessibility:** Helps screen readers identify key content.
5. **Not for Sidebars:** Excludes content repeated across multiple pages like site navigation or footer.

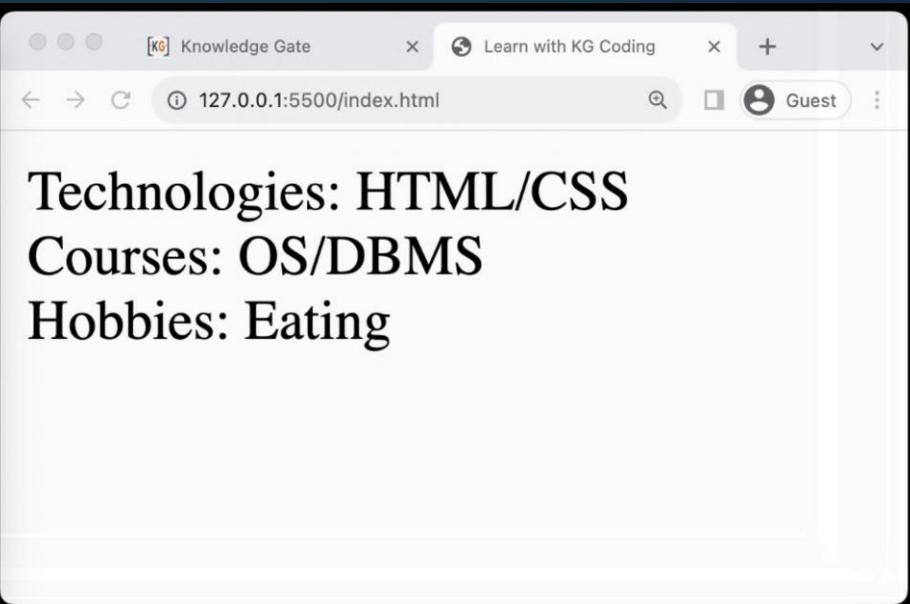
2.2.1 Section Tag



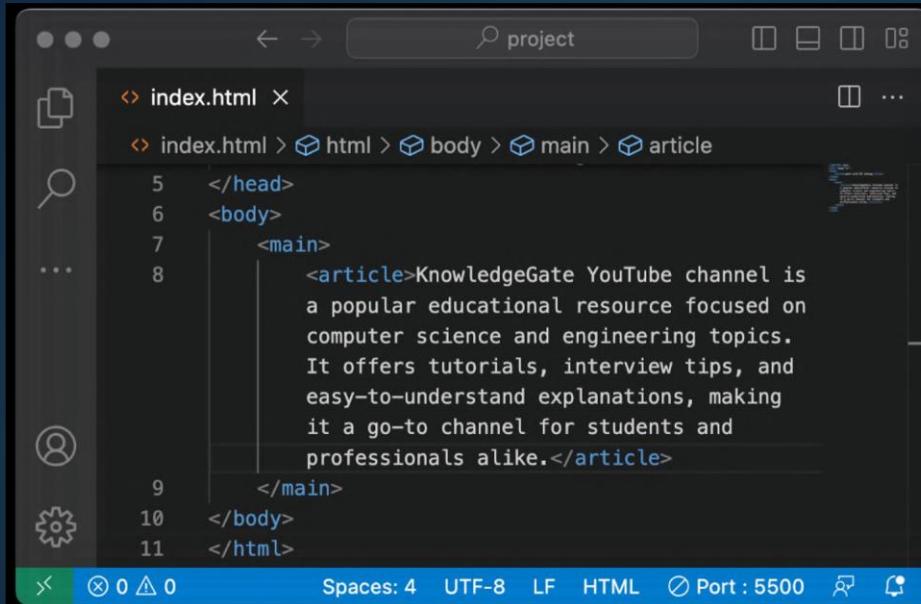
A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following code:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>Learn with KG Coding</title>
5  </head>
6  <body>
7  |   <main>
8  |   |       <section>Technologies: HTML/CSS</section>
9  |   |       <section>Courses: OS/DBMS</section>
10 |   |       <section>Hobbies: Eating</section>
11 |   </main>
12 </body>
13 </html>
```

The code editor has a dark theme with light-colored syntax highlighting. The status bar at the bottom shows "Spaces: 4" and "Port : 5500".



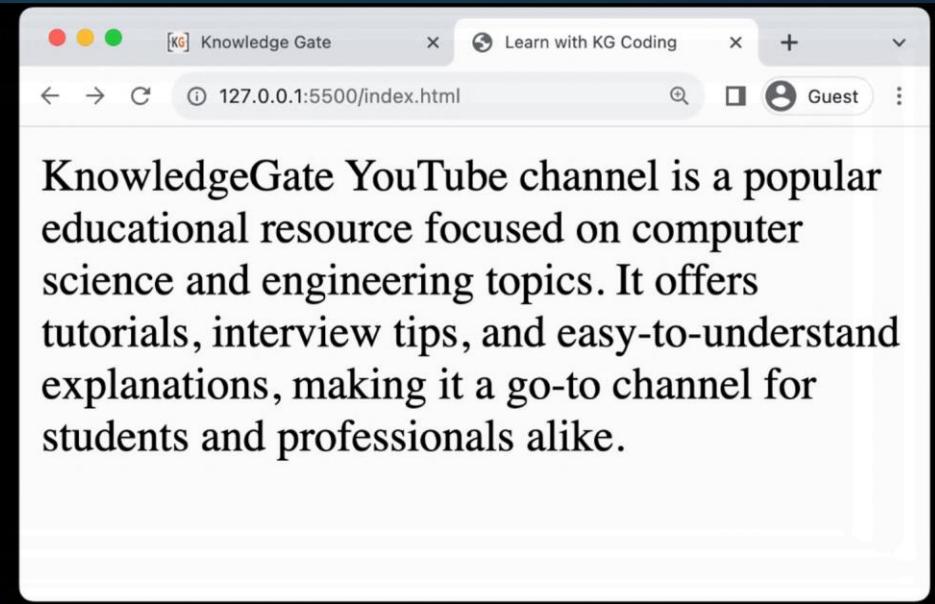
2.2.2 Article Tag



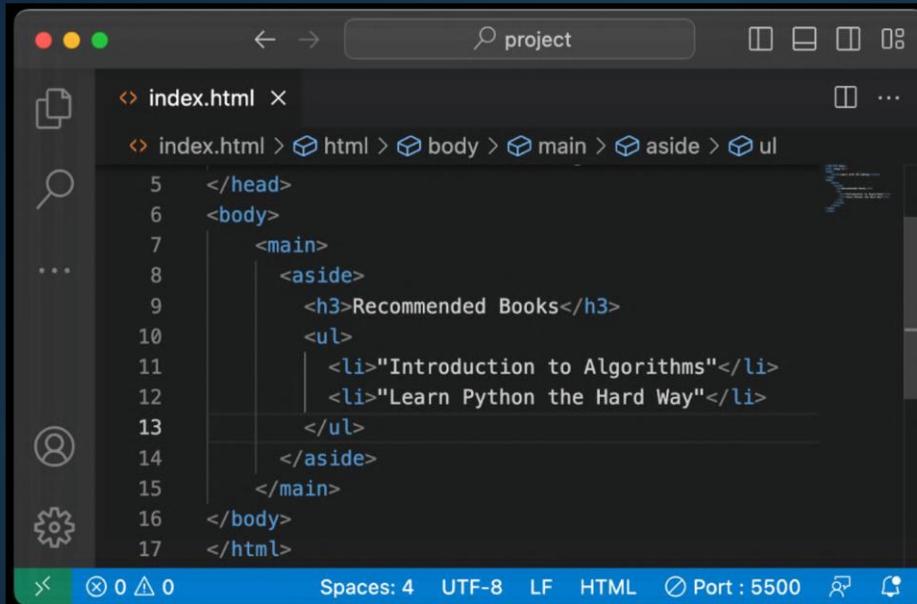
A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following HTML code:

```
<!-- index.html -->
<html>
  <head>
    ...
  </head>
  <body>
    <main>
      <article>KnowledgeGate YouTube channel is a popular educational resource focused on computer science and engineering topics. It offers tutorials, interview tips, and easy-to-understand explanations, making it a go-to channel for students and professionals alike.</article>
    </main>
  </body>
</html>
```

The code editor has a dark theme with light-colored syntax highlighting. The status bar at the bottom shows "Spaces: 4" and "Port : 5500".



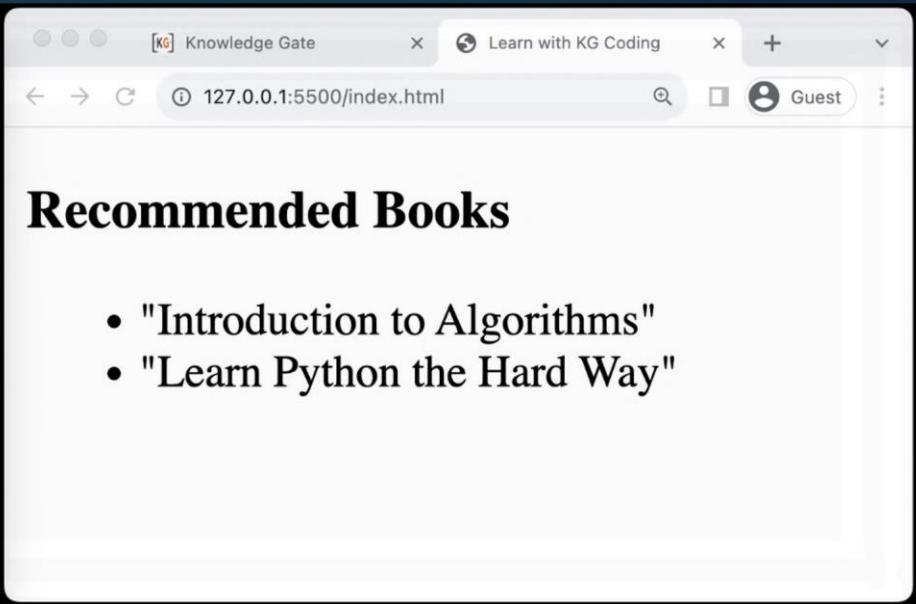
2.2.3 Aside Tag



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following HTML code:

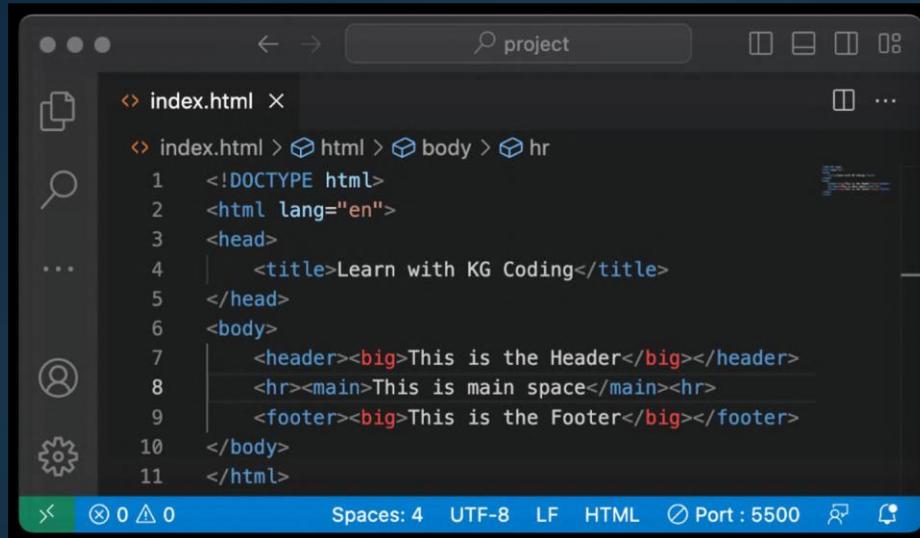
```
5  </head>
6  <body>
7    <main>
8      <aside>
9        <h3>Recommended Books</h3>
10       <ul>
11         <li>"Introduction to Algorithms"</li>
12         <li>"Learn Python the Hard Way"</li>
13       </ul>
14     </aside>
15   </main>
16 </body>
17 </html>
```

The code editor has a dark theme with light-colored syntax highlighting. The sidebar on the left shows icons for file, search, and user.



- "Introduction to Algorithms"
- "Learn Python the Hard Way"

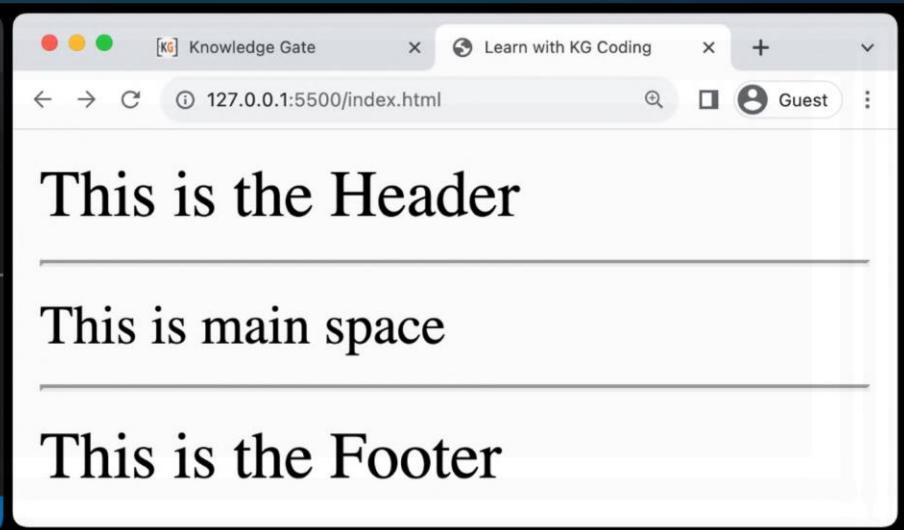
2.3 Footer Tag



A screenshot of a code editor window titled "project". The file "index.html" is open, displaying the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <header><b>This is the Header</b></header>
    <hr><main>This is main space</main><hr>
    <footer><b>This is the Footer</b></footer>
</body>
</html>
```

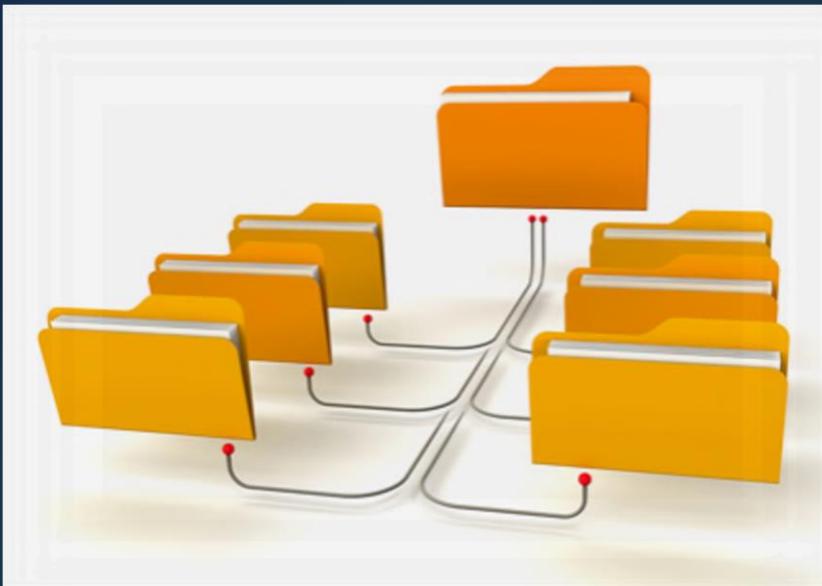
The code editor interface includes a sidebar with icons for file, search, and user, and a bottom status bar showing "Spaces: 4" and "Port : 5500".



1. Purpose: For footer content like extra info or links.
2. Semantic: Provides meaning to enclosed content.
3. Location: Typically at the bottom of pages or sections.
4. Content: Includes copyrights, contact info, and social links.
5. Multiple Instances: Can be used more than once on a page.

FF5F1F HTML and Project **Level** Structure **4**

1.3. Folder



The screenshot shows a code editor interface with the following details:

- EXPLORER** view: Shows a project structure with a folder named "PROJECT". Inside "PROJECT" are sub-folders "assets" (containing "padh-lo.MOV"), "css" (empty), "images" (containing "sanchitsir.png"), "pages" (empty), "scripts" (empty), and "index.html".
- EDITOR** view: Displays the content of "index.html". The code is as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
</body>
</html>
```

At the bottom of the editor, status bar information includes: Line 8, Column 8, Spaces: 4, UTF-8, LF, HTML, Port: 55.

- Root Directory:** Main folder containing all website files.
- HTML Files:** Store main .html files at the root level for easy access.
- CSS Folder:** Create a css/ folder for all Cascading Style Sheets.
- JS Folder:** Use a scripts/ folder for JavaScript files.
- Images Folder:** Store images in an images/ or images/ folder.
- Assets:** Other assets like fonts can go in an assets/ folder.
- Sub-directories:** For multi-page websites, use sub-folders to categorize content.



1.4. More

The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <nav>
        <a href="#home">Home</a><br>
        <a href="#about">About</a><br>
        <a href="#services">Services</a><br>
        <a href="#contact">Contact</a><br>
    </nav>
</body>
</html>
```

The browser window shows the rendered HTML with four underlined links: Home, About, Services, and Contact.

1. **Purpose:** Encloses navigation links or menus.
2. **Semantic:** Signals that the content is meant for navigating the site.

3. Common Content: Usually contains lists ``, `` of links `<a>`.
4. Accessibility: Aids screen readers in identifying site navigation.

4.2 Block / Inline Elements

Block Elements

- New Line: Start on a new line.
- Full Width: Take up all horizontal space.
- Styling: Can have margins and padding.

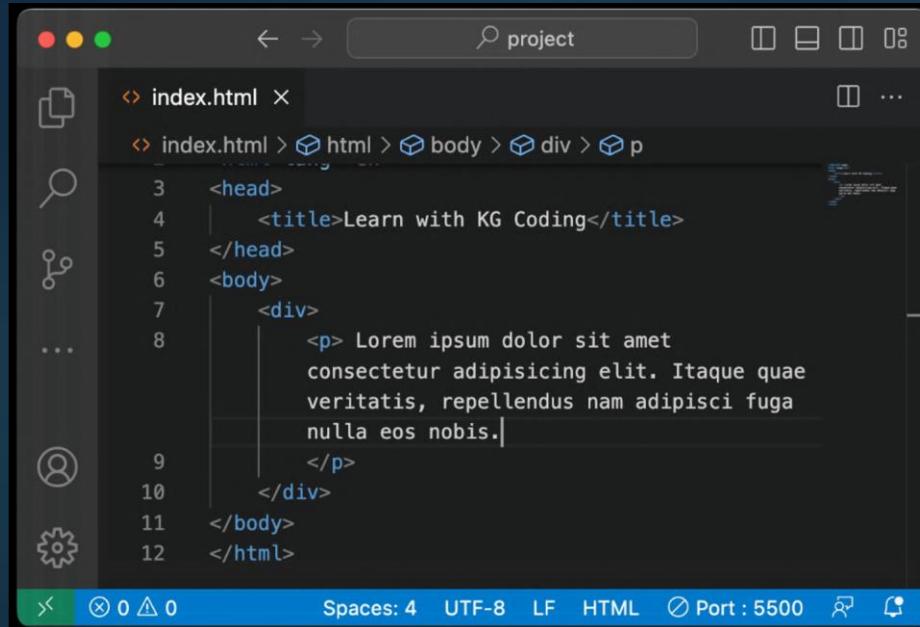
- Size: Width and height can be set.
- Examples: `<div>`, `<p>`, `<h1>`, ``, ``.

Inline Elements

- Flow: Stay in line with text.

- **Width:** Just as wide as the content.
- **No Break:** No new line between elements.
- **Limited Styling:** Can't set size easily.
- **Examples:** , <a>, , , .

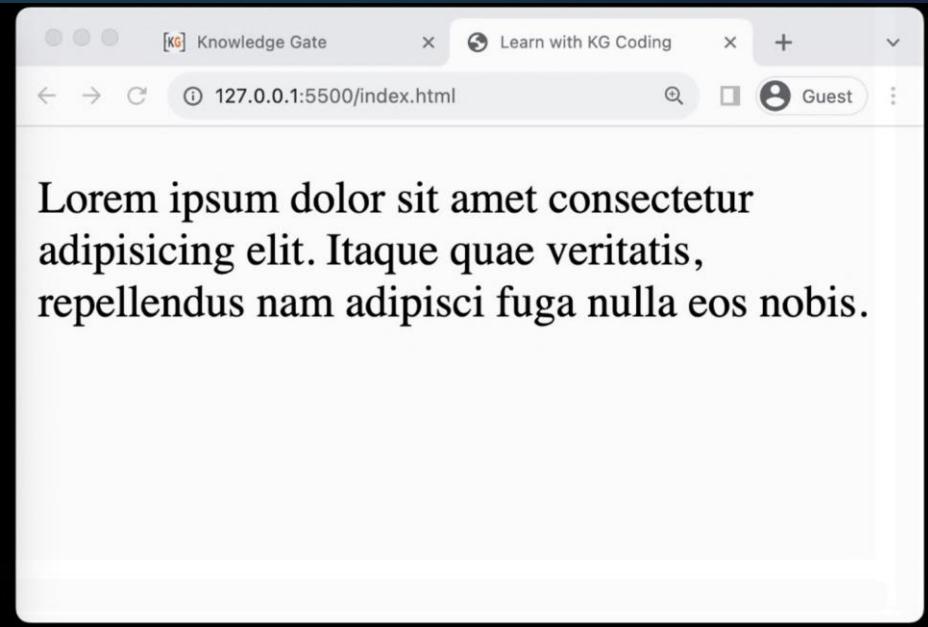
4.3 Div Tags



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following code:

```
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <div>
        <p>Lorem ipsum dolor sit amet
        consectetur adipisicing elit. Itaque quae veritatis,
        repellendus nam adipisci fuga nulla eos nobis.</p>
    </div>
</body>
</html>
```

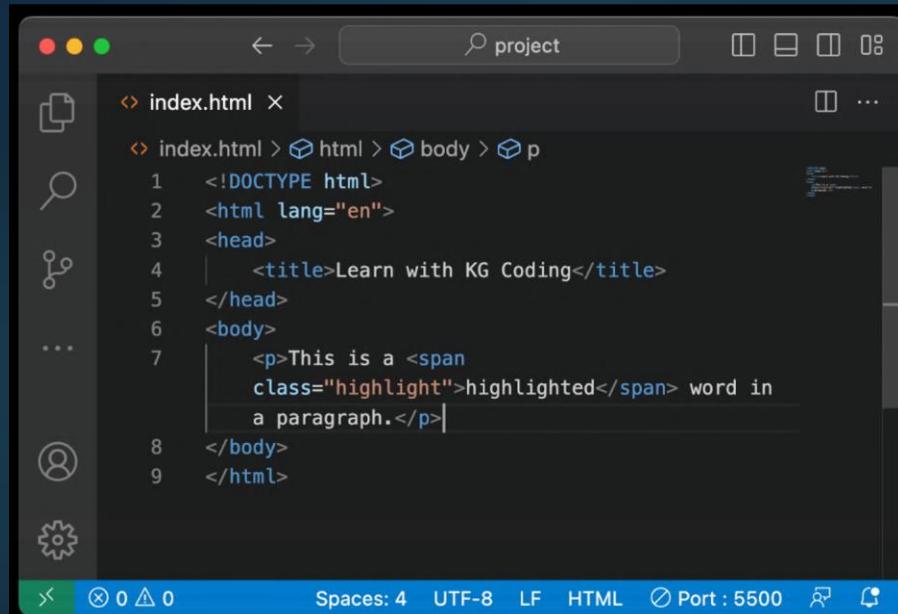
The code editor has a dark theme with light-colored syntax highlighting. A sidebar on the left contains icons for file operations like new, open, save, and search.



A screenshot of a web browser window titled "Knowledge Gate". The address bar shows "127.0.0.1:5500/index.html". The page content is:

Lore ipsum dolor sit amet consectetur
adipisicing elit. Itaque quae veritatis,
repellendus nam adipisci fuga nulla eos nobis.

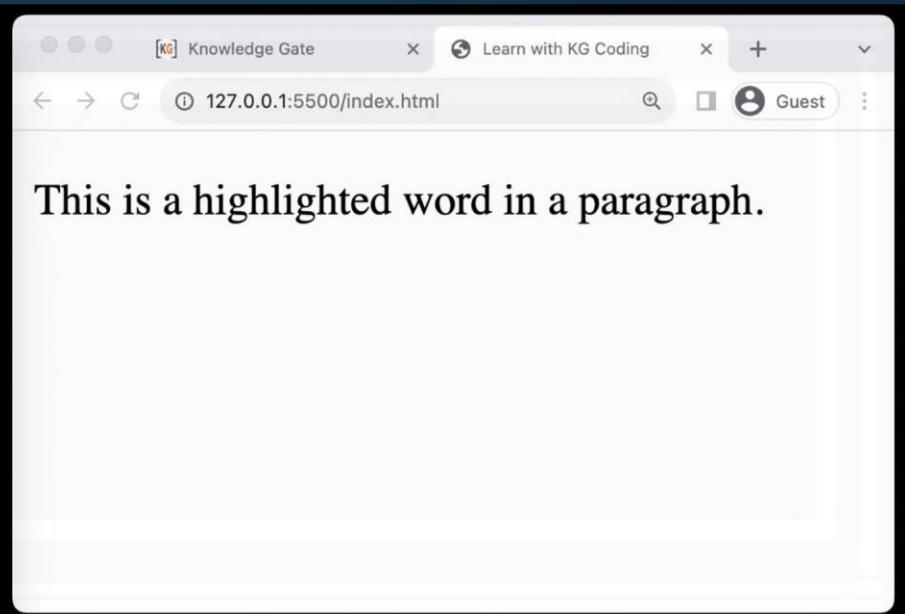
4.4 Span Tags



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <p>This is a <span
        class="highlight">highlighted</span> word in
        a paragraph.</p>
</body>
</html>
```

The word "highlighted" is highlighted with a blue background and white text, indicating it has been selected or styled.



1. Semantic / Non-Semantic Tags
2. Body Tags
 1. Header Tag
 2. Main Tag
 1. Section Tag
 2. Article Tag
 3. Aside Tag
3. Footer Tag
3. Folder Structure
 1. Recommended Folder structure
4. More Tags
 1. Navigation tags
 2. Block / Inline Elements
 3. Div tags



4. Span Tags

Project Level 4

HTML and Project Structure

1. Create a page with header, footer, main(section, article, aside tag).



2. Make sure the project from level 3 has correct folder structure.
3. Create groupings of multiple tags using div.
4. Create navigation to important sections of your page.

Level 5

List, Tables & Forms

1. List Tag
 1. Ordered Lists
 2. Types of Ordered Lists

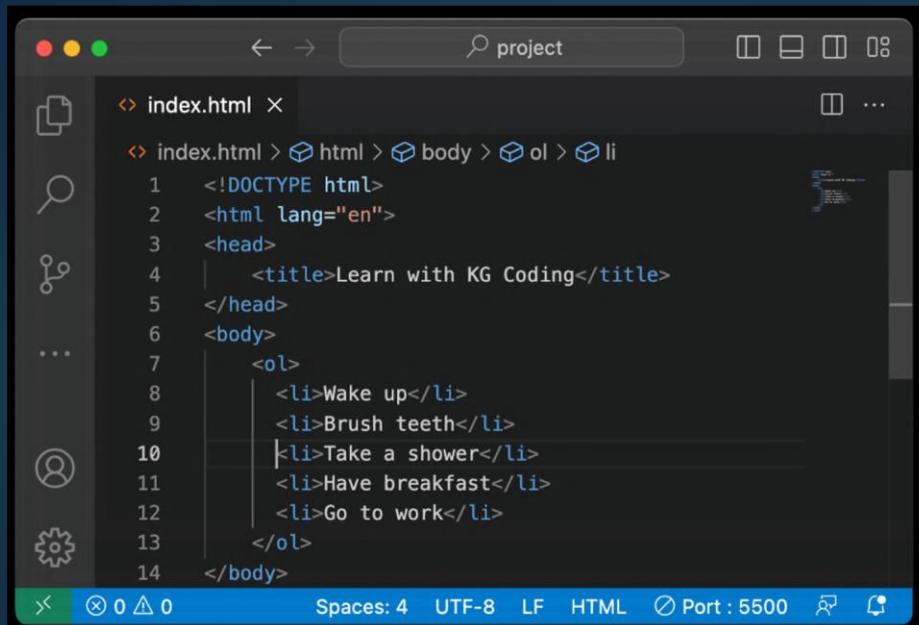
3. **Unordered Lists**
2. **Table Tag**
 1. **<tr>, <td>, <th> tags**
 2. **Captions**
 3. **Col spans**
3. **Forms**
 1. **Input tag**
 2. **Action Attributes**
 3. **Name and Value Property**
 4. **Label Tag**
 5. **Exploring Types**
4. **iFrame Tag**
 1. **Using iFrames**



1.1.

List Tag

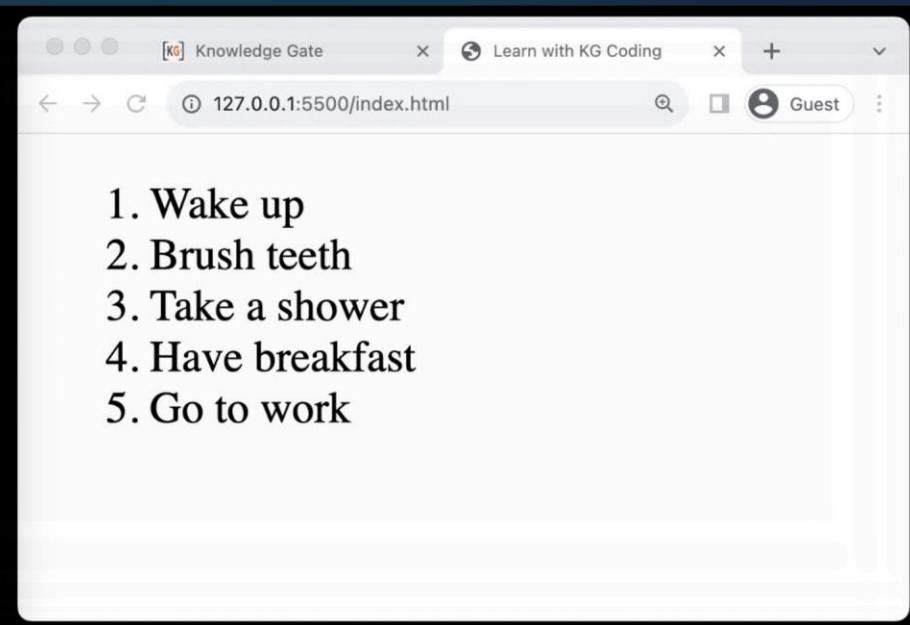
1.1 Ordered Lists



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <ol>
        <li>Wake up</li>
        <li>Brush teeth</li>
        <li>Take a shower</li>
        <li>Have breakfast</li>
        <li>Go to work</li>
    </ol>
</body>
```

The code editor has a dark theme with light-colored syntax highlighting. The status bar at the bottom shows "Spaces: 4" and "Port : 5500".



A screenshot of a web browser window titled "Knowledge Gate". The address bar shows "127.0.0.1:5500/index.html". The page content is:

1. Wake up
2. Brush teeth
3. Take a shower
4. Have breakfast
5. Go to work

Ordered Lists

- Numeric: Default type, (1, 2, 3, ...)
Attribute: type="1"
- Uppercase Letters: (A, B, C, ...)
Attribute: type="A"
- Lowercase Letters: (a, b, c, ...)
Attribute: type="a"
- Uppercase Roman: (I, II, III, ...)
Attribute: type="I"
- Lowercase Roman: (i, ii, iii, ...)
Attribute: type="i"

A. Apple
B. Banana
C. Cherry
D. Dragonfruit

a. Apple
b. Banana
c. Cherry
d. Dragonfruit

I. Apple
II. Banana
III. Cherry
IV. Dragonfruit

i. Apple
ii. Banana
iii. Cherry
iv. Dragonfruit

1.3 Unordered Lists

The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <title>Learn with KG Coding</title>
5  </head>
6  <body>
7      <ul>
8          <li>Apple</li>
9          <li>Banana</li>
10         <li>Cherry</li>
11         <li>Dragonfruit</li>
12     </ul>
13 </body>
14 </html>
```

The browser window shows the rendered output of the HTML code, displaying an unordered list with four items: Apple, Banana, Cherry, and Dragonfruit.



1.2. Table

The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

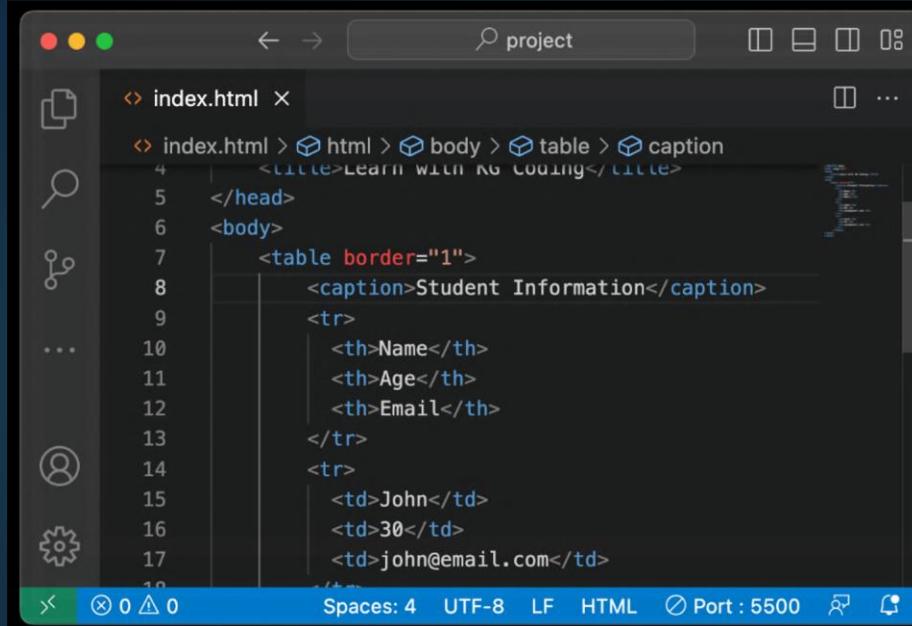
```
<!DOCTYPE html>
<html>
  <head>
    <title>Learn with KG Coding</title>
  </head>
  <body>
    <table border="1">
      <tr>
        <th>Name</th>
        <th>Age</th>
        <th>Email</th>
      </tr>
      <tr>
        <td>John</td>
        <td>30</td>
        <td>john@email.com</td>
      </tr>
    </table>
  </body>
</html>
```

The browser window on the right shows the rendered HTML at `127.0.0.1:5500/index.html`. It displays a table with three rows and three columns. The columns are labeled **Name**, **Age**, and **Email**. The data rows are:

Name	Age	Email
John	30	john@email.com
Jane	25	jane@email.com

1. **<tr> Table Row : Used to define a row in an HTML table.**

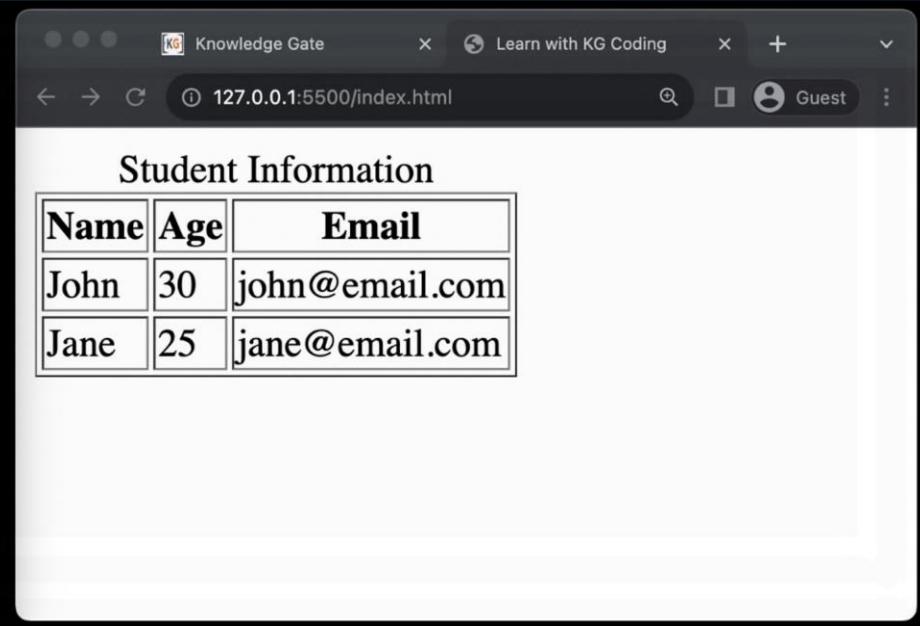
2.2 Captions



A screenshot of a code editor window titled "project". The file "index.html" is open, showing the following HTML code:

```
<!-- Learn WITH KG CODING -->
<head>
</head>
<body>
<table border="1">
<caption>Student Information</caption>
<tr>
<th>Name</th>
<th>Age</th>
<th>Email</th>
</tr>
<tr>
<td>John</td>
<td>30</td>
<td>john@email.com</td>
</tr>
</table>
</body>
</html>
```

The code editor has a dark theme with light-colored syntax highlighting. The status bar at the bottom shows "Spaces: 4", "UTF-8", "LF", "HTML", "Port : 5500", and other icons.



A screenshot of a web browser window titled "Knowledge Gate" with the URL "127.0.0.1:5500/index.html". The page displays the following content:

Student Information

Name	Age	Email
John	30	john@email.com
Jane	25	jane@email.com

2.3 Col Spans

project

```
<index.html>
  <html>
    <body>
      <table>
        <caption>Student Information</caption>
        <thead>
          <tr>
            <th>Name</th>
            <th>Age</th>
            <th>Email</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td>John</td>
            <td>30</td>
            <td>john@email.com</td>
          </tr>
          <tr>
            <td colspan="3">This cell spans 3 columns</td>
          </tr>
        </tbody>
      </table>
    </body>
  </html>
```

Spaces: 4 UTF-8 LF HTML ⚡ Port : 5500 🔍

Knowledge Gate Learn with KG Coding Guest

127.0.0.1:5500/index.html

Student Information

Name	Age	Email
John	30	john@email.com
This cell spans 3 columns		



1.3. Forms

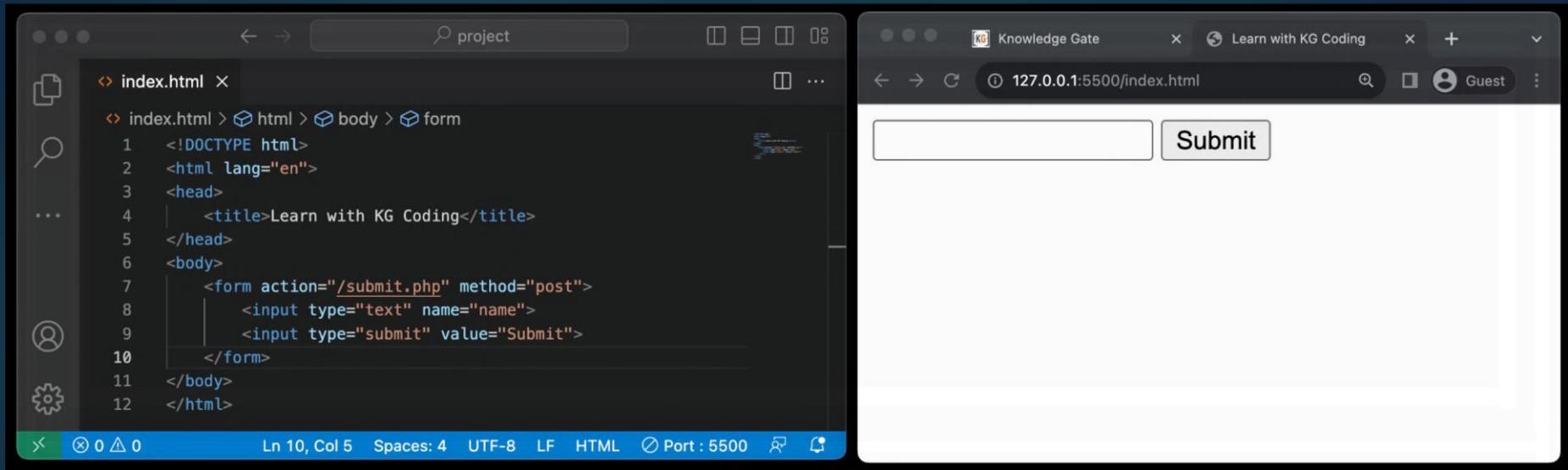
3.1 Input Tag

The image shows a code editor on the left and a web browser on the right. The code editor displays the contents of index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        Text: <input type="text" placeholder="Enter Text"><br>
        Password: <input type="password"><br>
    </form>
</body>
</html>
```

The browser window shows the rendered HTML. It has two input fields. The first input field is a text input with the placeholder "Enter Text". The second input field is a password input. The text "Hello" is entered into the text input, and the password "*****" is entered into the password input.

3.2 Action attribute

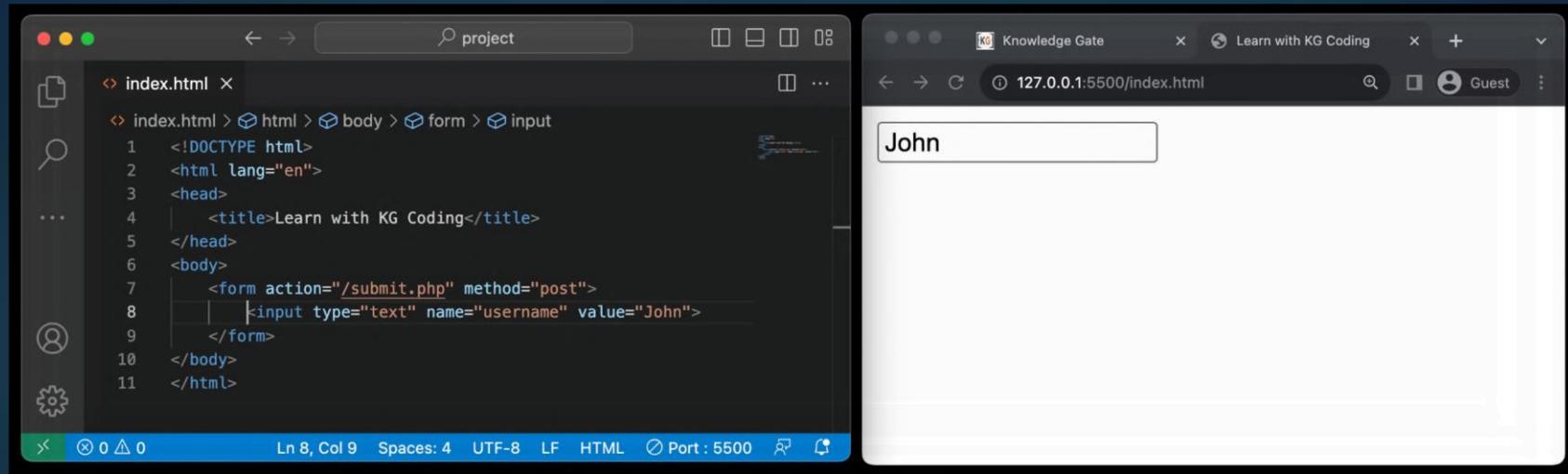


The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form action="/submit.php" method="post">
        <input type="text" name="name">
        <input type="submit" value="Submit">
    </form>
</body>
</html>
```

The browser window shows the rendered HTML. It features a text input field and a "Submit" button.

3.3 Name and Value property



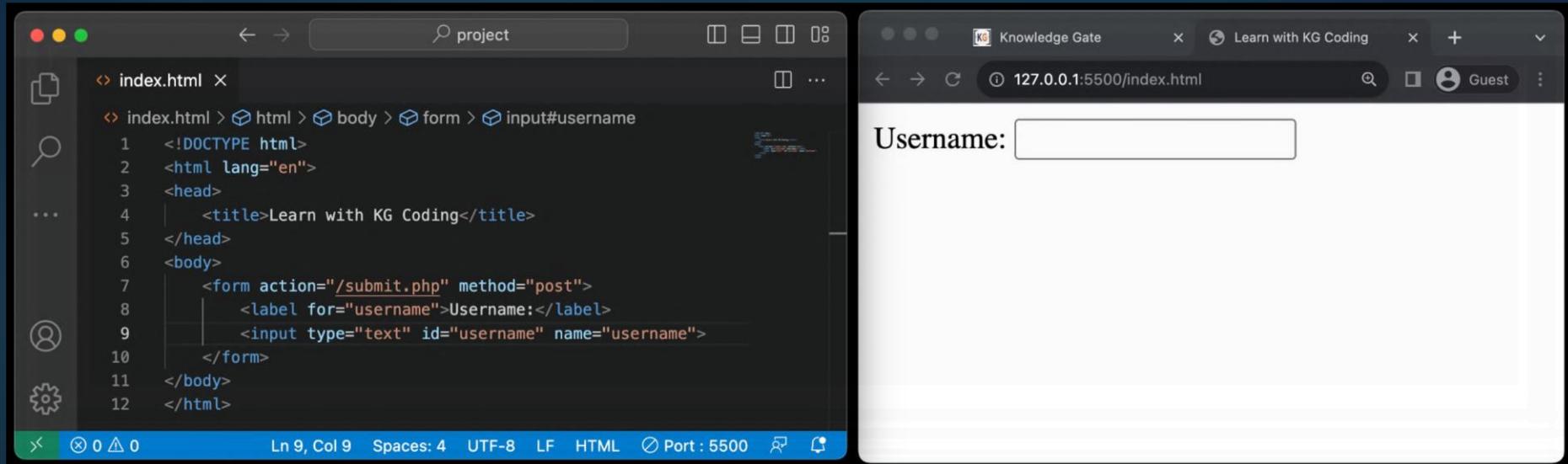
The image shows a code editor on the left and a browser window on the right. The code editor displays the file 'index.html' with the following content:

```
index.html
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form action="/submit.php" method="post">
        <input type="text" name="username" value="John">
    </form>
</body>
</html>
```

The browser window shows the URL `127.0.0.1:5500/index.html`. Inside the input field, the value "John" is displayed.

- `name` Property:
 - ID for Data: Identifies form elements when submitting.

3.4 Label Tag



The image shows a code editor on the left and a web browser on the right. The code editor displays the file `index.html` with the following content:

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <title>Learn with KG Coding</title>
5 </head>
6 <body>
7   <form action="/submit.php" method="post">
8     <label for="username">Username:</label>
9     <input type="text" id="username" name="username">
10   </form>
11 </body>
12 </html>
```

The browser window on the right shows the rendered HTML. It displays the text "Username:" followed by an empty text input field.

- Purpose: Adds a text description to form elements.

3.5 Input type: Date

The image shows a code editor on the left and a browser window on the right.

Code Editor (index.html):

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        Text: <input type="text" placeholder="Enter Text"><br>
        Password: <input type="password"><br>
        Date: <input type="date"><br>
    </form>
</body>
</html>
```

Browser Preview:

Text: hello

Password:

Date: 16/08/2023

3.5 Input type: File

The image shows a code editor on the left and a web browser on the right. The code editor displays the contents of index.html, which includes HTML code for a form with four input fields: text, password, date, and file.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        Text: <input type="text" placeholder="Enter Text"><br>
        Password: <input type="password"><br>
        Date: <input type="date"><br>
        File: <input type="file"><br>
    </form>
</body>
</html>
```

The browser window shows the rendered HTML. It has four input fields:

- Text: An input field with placeholder "Enter Text".
- Password: An input field for entering a password.
- Date: An input field for entering a date in dd/mm/yyyy format, with a calendar icon.
- File: An input field with a "Choose file" button and a message "No file chosen".

project

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        Color: <input type="color"><br>
        Text: <input type="text" placeholder="Enter Text"><br>
        Password: <input type="password"><br>
        Date: <input type="date"><br>
        File: <input type="file"><br>
    </form>
</body>
</html>
```

Ln 15, Col 8 Spaces: 4 UTF-8 LF HTML Port : 5500

Knowledge Gate Learn with KG Coding Guest

127.0.0.1:5500/index.html

Color: [Color Box]

Text: [Text Input]

Password: [Password Input]

Date: [Date Input]

File: [File Input]

R: 0 G: 0 B: 0



project

index.html X

index.html > html > body > form > br

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  |   <title>Learn with KG Coding</title>
5  </head>
6  <body>
7  |   <form>
8  |       Color: <input type="color"><br>
9  |       Text: <input type="text" placeholder="Enter Text"><br>
10 |       Password: <input type="password"><br>
11 |       Date: <input type="date"><br>
12 |       File: <input type="file"><br>
13 |       Range: <input type="range"><br>
14 |   </form>
15 </body>
16 </html>
```

x 0 △ 0 Ln 13, Col 40 Spaces: 4 UTF-8 LF HTML ⚙️ Port : 5500 🔍

Knowledge Gate Learn with KG Coding Guest

127.0.0.1:5500/index.html

Color:

Text:

Password:

Date: dd / mm / yyyy

File: Choose file No file chosen

Range:

project

index.html X

```
< index.html > ⚡ html > ⚡ body > ⚡ form > ⚡ br
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <title>Learn with KG Coding</title>
5  </head>
6  <body>
7    <form>
8      Color: <input type="color"><br>
9      Text: <input type="text" placeholder="Enter Text"><br>
10     Password: <input type="password"><br>
11     Date: <input type="date"><br>
12     File: <input type="file"><br>
13     Range: <input type="range"><br>
14     Button: <input type="button" value="Click Me"><br>
15   </form>
16 </body>
17 </html>
```

x 0 △ 0 Ln 14, Col 59 Spaces: 4 UTF-8 LF HTML ⚡ Port : 5500 🔍 🔍

Knowledge Gate Learn with KG Coding Guest ...

127.0.0.1:5500/index.html

Color:

Text:

Password:

Date: dd/mm/yyyy

File: Choose file No file chosen

Range:

Button:

← → ⌂ project ⌂ ⌂ ⌂ ⌂

index.html ×

```
< index.html > ⌂ html > ⌂ body > ⌂ form > ⌂ br
1   <!DOCTYPE html>
2   <html lang="en">
3     <head>
4       <title>Learn with KG Coding</title>
5     </head>
6     <body>
7       <form>
8         Color: <input type="color"><br>
9         Text: <input type="text" placeholder="Enter Text"><br>
10        Password: <input type="password"><br>
11        Date: <input type="date"><br>
12        File: <input type="file"><br>
13        Range: <input type="range"><br>
14        Button: <input type="button" value="Click Me"><br>
15        Submit: <input type="submit" value="Submit"><br>
16      </form>
17    </body>
18  </html>
```

x 0 △ 0 Ln 15, Col 57 Spaces: 4 UTF-8 LF HTML ⌂ Port : 5500 ⌂ ⌂

Knowledge Gate Learn with KG Coding Guest ⌂

127.0.0.1:5500/index.html

Color:

Text:

Password:

Date: dd/mm/yyyy

File: Choose file No file chosen

Range:

Button:

Submit:

project

index.html X

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        <label for="male">Male</label>
        <input type="radio" id="male" name="gender" value="Male"><br>

        <label for="female">Female</label>
        <input type="radio" id="female" name="gender" value="Female"><br>

        <label for="other">Other</label>
        <input type="radio" id="other" name="gender" value="Other"><br>
    </form>
</body>
</html>
```

Ln 12, Col 74 Spaces: 4 UTF-8 LF HTML ⚙️ 🔍

Knowledge Gate Learn with KG Coding Guest :

127.0.0.1:5500/index.html

Male

Female

Other

project

index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        <label for="apple">Apple</label>
        <input type="checkbox" id="apple" name="fruit" value="Apple"><br>

        <label for="banana">Banana</label>
        <input type="checkbox" id="banana" name="fruit" value="Banana"><br>

        <label for="cherry">Cherry</label>
        <input type="checkbox" id="cherry" name="fruit" value="Cherry"><br>
    </form>
</body>
</html>
```

Ln 6, Col 7 Spaces: 4 UTF-8 LF HTML ⚙️ 🔍

Knowledge Gate Learn with KG Coding Guest

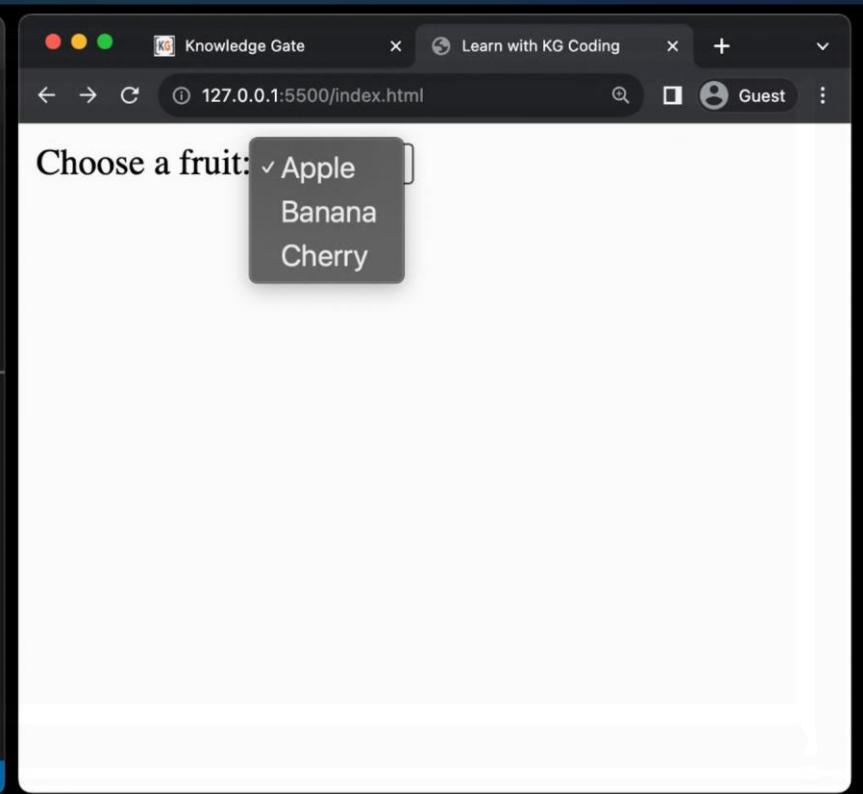
127.0.0.1:5500/index.html

Apple
Banana
Cherry

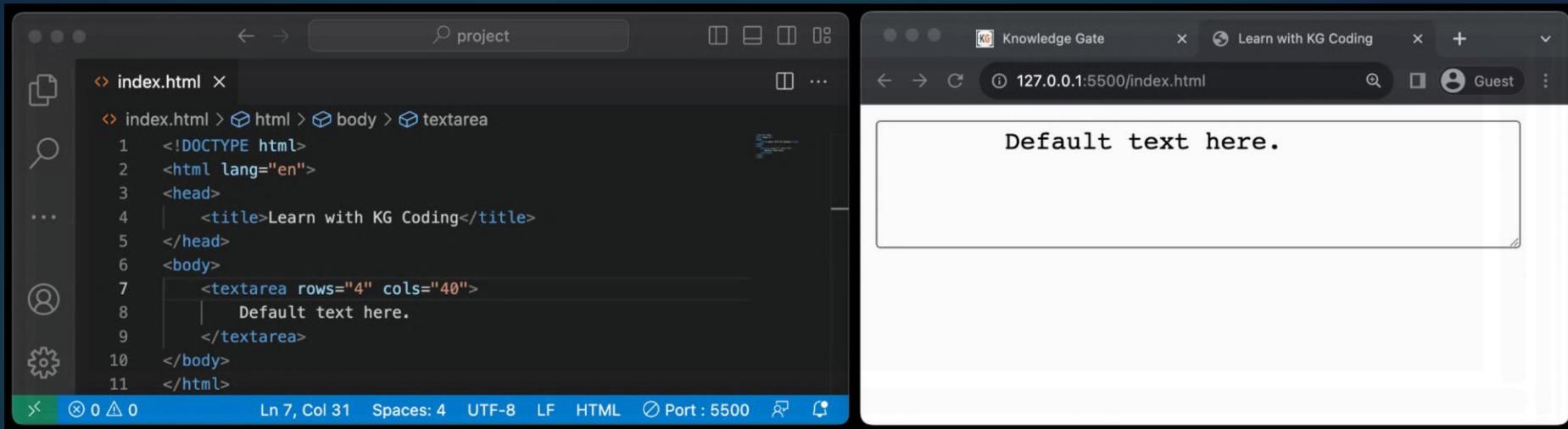
A screenshot of a code editor window titled "project". The file "index.html" is open, displaying the following HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <form>
        <label for="fruits">Choose a fruit:</label>
        <select id="fruits" name="fruits">
            <option value="apple">Apple</option>
            <option value="banana">Banana</option>
            <option value="cherry">Cherry</option>
        </select>
    </form>
</body>
</html>
```

The code editor has a dark theme with light-colored syntax highlighting. The status bar at the bottom shows "Ln 14, Col 14" and "Port : 5500".



3.5 Input type: TextArea

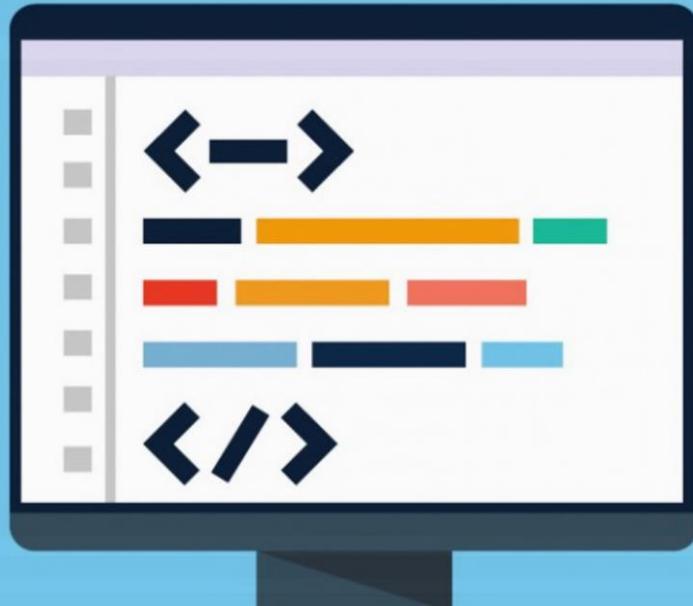


The image shows a code editor on the left and a browser window on the right. The code editor displays the file 'index.html' with the following content:

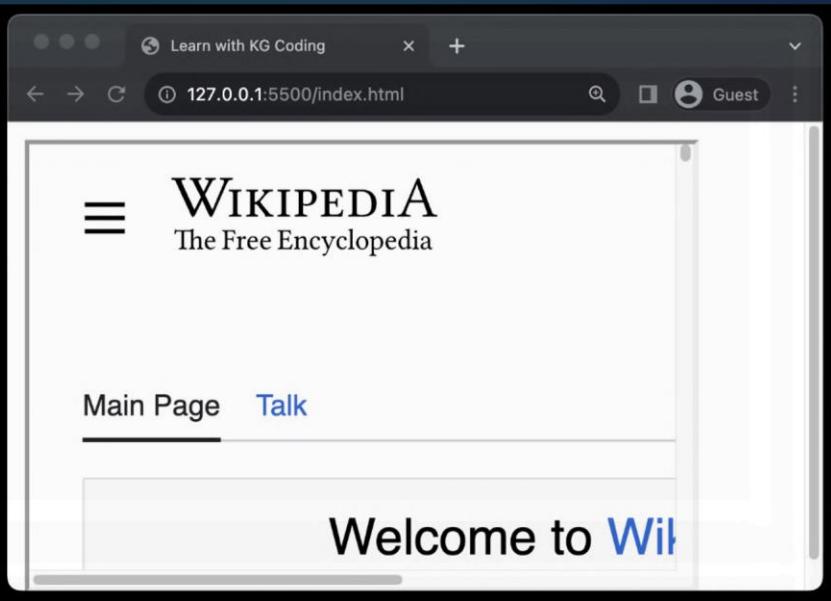
```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <textarea rows="4" cols="40">
        Default text here.
    </textarea>
</body>
</html>
```

The browser window shows the rendered HTML at the URL 127.0.0.1:5500/index.html. It displays a text area with the placeholder text "Default text here."

1. Purpose: <textarea> is used for multi-line text input in forms.



1.4. iFrame



The image shows a split-screen view. On the left is a code editor window titled "index.html" with the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Learn with KG Coding</title>
</head>
<body>
    <iframe width="300" height="200" src="https://en.wikipedia.org/wiki/Main_Page"></iframe>
</body>
</html>
```

The right side shows a browser window titled "Learn with KG Coding" displaying the Wikipedia homepage at 127.0.0.1:5500/index.html. The page features the Wikipedia logo and the text "The Free Encyclopedia". Below the logo are links for "Main Page" and "Talk". A large "Welcome to Wi" message is partially visible at the bottom.

1. **Embedded Content: Allows you to embed another webpage or multimedia content within a webpage.**

FF5F1F

2. **src Attribute:** Specifies the URL of the content to be embedded.
3. **Dimensions:** Width and height can be set using width and height attributes.

Level 5 Revision

List, Tables & Forms

- 1. List Tag**
 - 1. Ordered Lists**
 - 2. Types of Ordered Lists**
 - 3. Unordered Lists**
- 2. Table Tag**
 - 1. <tr>, <td>, <th> tags**
 - 2. Captions**
 - 3. Col spans**
- 3. Forms**
 - 1. Input tag**
 - 2. Action Attributes**
 - 3. Name and Value Property**
 - 4. Label Tag**
 - 5. Exploring Types**



Project Level 5

List, Tables & Forms

1. Create a page with all type of ordered list and one unordered list.
2. Create a table with headings, captions and a few rows. One of heading should take at least 3 columns.



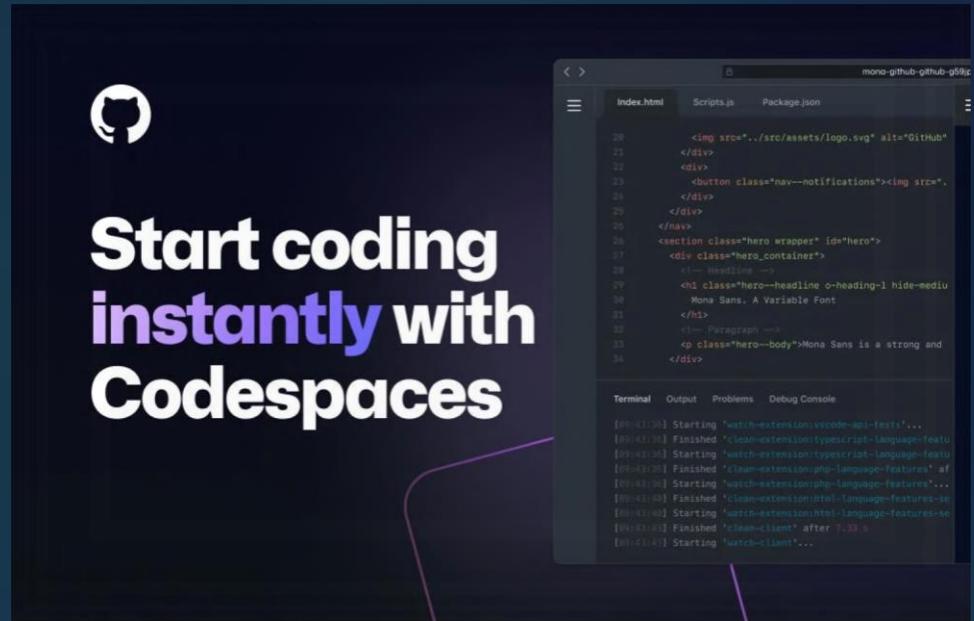
3. Create a contact me form with relevant details for your resume website.
4. Use iFrame to add this video to your page.

Level Bonus

Github Pages & CodeSpace

1. Github

1. **What is Version Control**
 2. **What is Git and GitHub**
 3. **Account Creation**
 4. **Creating a Repo**
 5. **Creating a Codespace**
 6. **Creating a Github page**
 7. **Publishing our project**
2. **FrameWorks**
1. **React**
 2. **Angular**
 3. **Vue**



FF5F1F



1.1. GitHub

1.1 What is Version Control

1. **Definition:** A system to track changes in files over time.
2. **Types:** Centralized (like SVN) and Distributed (like Git).
3. **Purpose:** Helps in teamwork and fixes mistakes.



4. **Snapshots:** Each 'commit' saves a file version.
5. **Branching:** Lets you work on different tasks separately.
6. **Merge:** Combines changes from different people.
7. **Undo:** Easy to revert to older file versions.

1.2 What is Git and GitHub

What is Git?

- **Definition:** A software tool that tracks changes in code, enabling collaboration and version control.
- **Commit:** Records a snapshot of file changes.
- **Branch:** Allows separate paths of development.

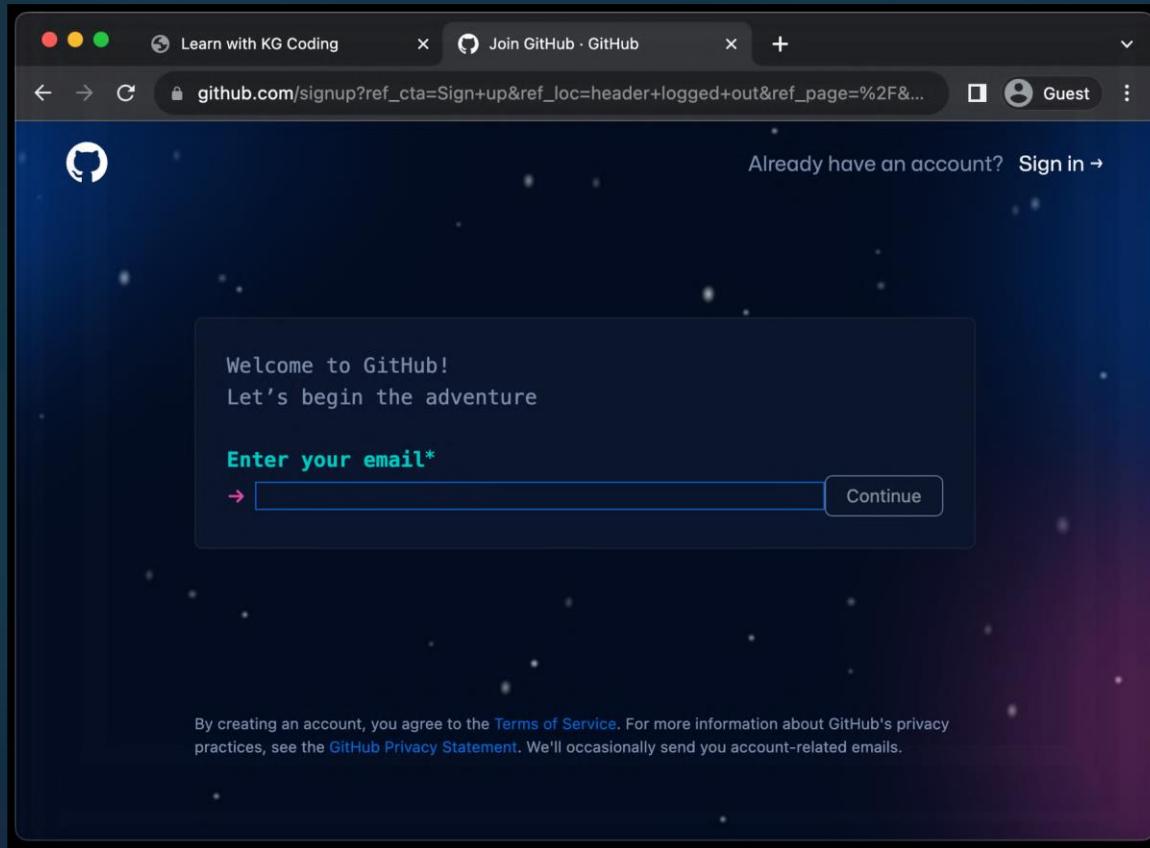


- **Merge:** Combines changes from different branches.

What is GitHub?

- **Definition:** A web service for hosting and collaborating on Git repositories.
- **Fork:** Creates a personal copy of another user's repository.
- **Pull Request:** A way to propose changes to existing code.
- **Issues:** Used for tracking bugs and feature ideas.





New repository

github.com/new

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Required fields are marked with an asterisk (*).

Owner * Repository name *

Great repository names are short and memorable. Need inspiration? How about [upgraded-parakeet](#)?

Description (optional)

Public Anyone on the internet can see this repository. You choose who can commit.

Private You choose who can see and commit to this repository.

Initialize this repository with:

Add a README file This is where you can write a long description for your project. [Learn more about READMEs](#).

Add .gitignore

.gitignore template: None

Choose which files not to track from a list of templates. [Learn more about ignoring files](#).

Choose a license

License: None

You are creating a public repository in your personal account.

Create repository

A screenshot of a GitHub repository page for 'KnowledgeGateCoding/java_for_beginner'. The page has a dark theme. At the top, there's a navigation bar with links for Code, Issues, Pull requests, Discussions, Actions, Projects, Wiki, Security, and Insights. Below the navigation bar, there's a search bar and a user profile icon. The main content area shows a file list with 'main' selected, containing files like 'LearningJava/src/in/kn...', '.gitignore', and 'README.md'. A modal window is open over the repository content, specifically for the 'Code' tab. The modal has tabs for 'Local' and 'Codespaces'. Under 'Codespaces', it says 'No codespaces' and 'You don't have any codespaces with this repository checked out'. It features a prominent green button labeled 'Create codespace on main'. Below this, there's a link 'Learn more about codespaces...'. At the bottom of the modal, it says 'Codespace usage for this repository is paid for by omprashantjain'. To the right of the modal, the repository details are shown: 'About' (No description, website, or topics provided), 'Readme', 'Activity', '2 stars', '1 watching', '0 forks', and a 'Report repository' link. There are also sections for 'Releases' (No releases published) and 'Packages' (No packages published).

KnowledgeGateCoding/java_for_beginner

Type ⌘ to search

Code Issues Pull requests Discussions Actions Projects Wiki Security Insights

KG java_for_beginner Public

main 1 branch 0 tags

Go to file Add file <> Code

omprashantjain Adding Links

LearningJava/src/in/kn... First co
.gitignore First co
README.md Adding

README.md

Complete JAVA

This is a KnowledgeGate initiative

Link to notes: Link to sample test

No codespaces

You don't have any codespaces with this repository checked out

Create codespace on main

Learn more about codespaces...

Codespace usage for this repository is paid for by omprashantjain

About

No description, website, or topics provided.

Readme

Activity

2 stars

1 watching

0 forks

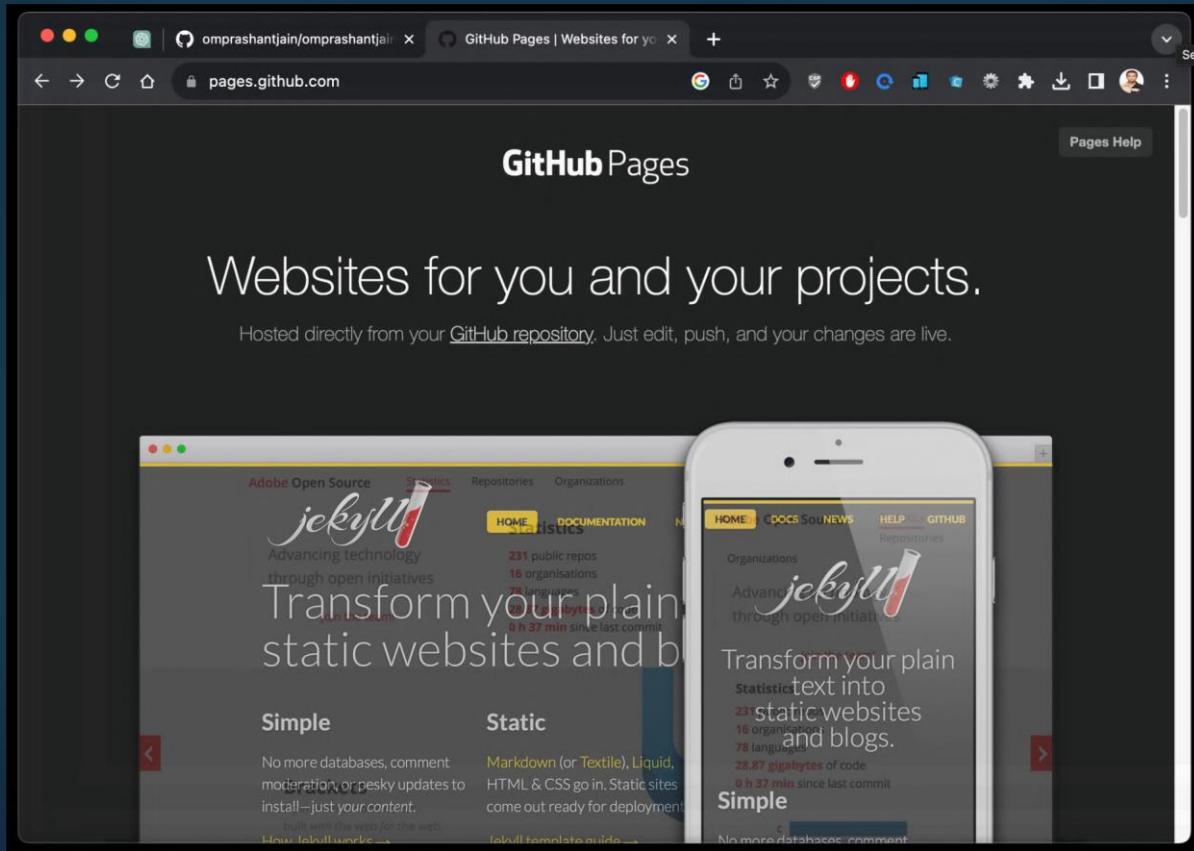
Report repository

Releases

No releases published

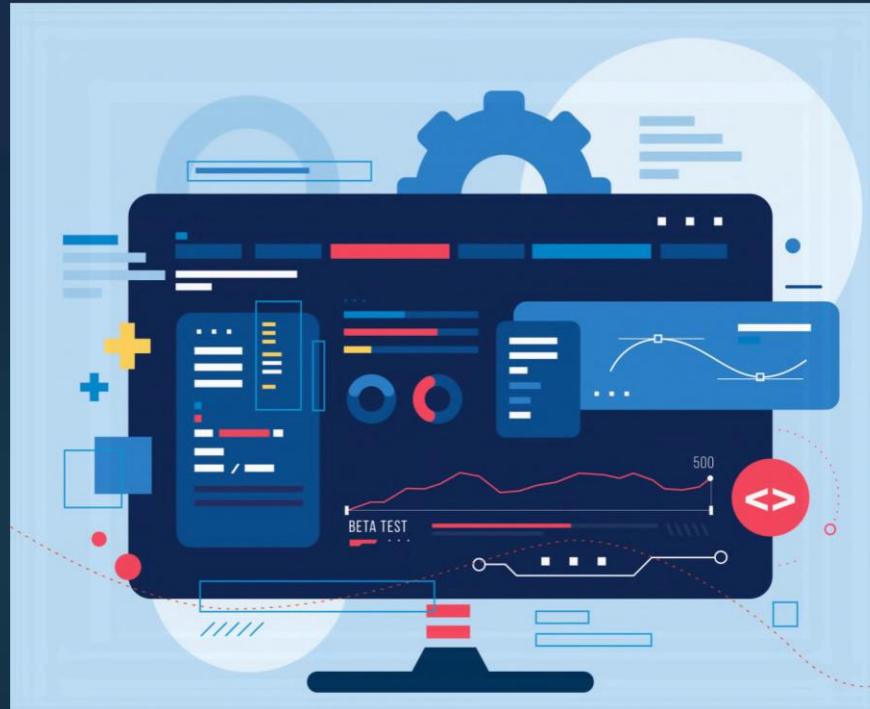
Create a new release

Packages



The screenshot shows the GitHub Pages settings interface for the repository `omprashantjain/omprashantjain.github.io`. The main content area displays the following information:

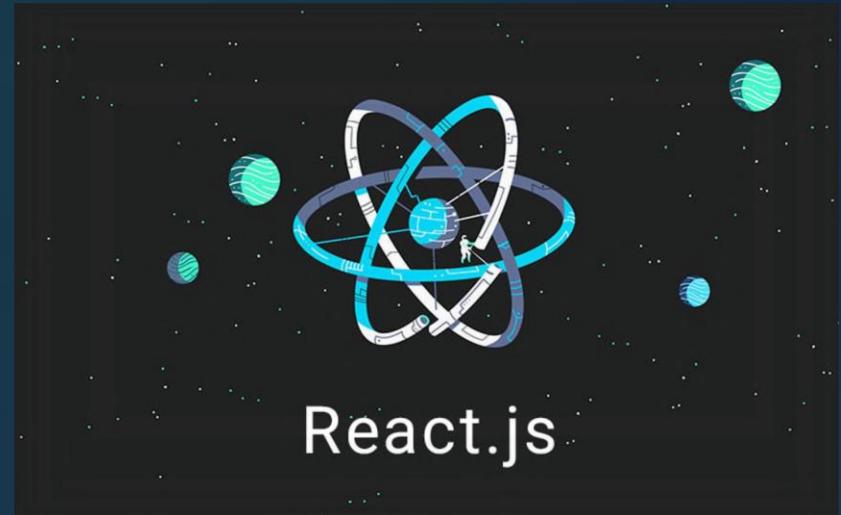
- General**: A summary section indicating that the site is live at <https://www.prashantjain.me>, last deployed by `github-pages[bot]` last year. It includes a "Visit site" button and a three-dot menu.
- Access**: Options for `Collaborators` and `Moderation options`.
- Code and automation**: Options for `Branches`, `Tags`, `Rules`, `Actions`, `Webhooks`, `Environments`, `Codespaces`, and `Pages` (which is selected).
- Build and deployment**:
 - Source**: Set to "Deploy from a branch".
 - Branch**: Set to "main".
 - A note states: "Your GitHub Pages site is currently being built from the `main` branch. Learn more about configuring the publishing source for your site."
 - Buttons for "Save" and a dropdown menu.
- Security**: Options for `Code security and analysis`, `Deploy keys`, and `Secrets and variables`.
- Integrations**: Options for `GitHub Apps` and `Email notifications`.
- Custom domain**: A section for managing custom domains, showing "www.prashantjain.me" with a "Save" button and a "Remove" button. A note indicates: "Custom domains allow you to serve your site from a domain other than `omprashantjain.github.io`. Learn more about configuring custom domains." Below this, a note says: "Your site was last deployed to the `github-pages` environment by the `pages` build and deployment workflow."



1.2. Frameworks

3.1 ReactJS

1. **Definition:** A tool for making websites interactive.
2. **Components:** Reusable pieces for building a webpage.
3. **Virtual DOM:** Makes websites faster by updating only what's needed.
4. **JSX:** A special way to write code that looks like HTML.
5. **State:** Keeps track of changes on the webpage.



- 6. **Props:** Shares information between different parts of a webpage.

3.2 AngularJS

- 1. **Definition:** A framework for building web applications, developed by Google.
- 2. **Two-Way Data Binding:** Updates both the view and the model simultaneously.
- 3. **Directives:** Custom HTML tags for added functionality.
- 4. **Dependency Injection:** Automatically manages how parts of the app work together.
- 5. **Controllers:** Manages the data for a specific part of the webpage.



3.3 VueJS

1. **Definition:** A JavaScript framework for creating web interfaces.
2. **Components:** Small, reusable parts for building a website.
3. **Reactivity:** Automatically updates the webpage when data changes.
4. **Directives:** Special tokens in HTML for added functionality.
5. **Vuex:** Helps manage shared data across the site.



1. Github

1. What is Version Control
2. What is Git and GitHub
3. Account Creation
4. Creating a Repo
5. Creating a Codespace
6. Creating a Github page
7. Publishing our project

2. FrameWorks

1. React
2. Angular
3. Vue



Level Bonus Project

Github Pages & CodeSpace

1. Create a Github account if you don't have one.
2. Create a repo for the project you have done till now.
3. Create a Codespace and make changes.



4. Publish it using Github pages.

Major Project

Idea

1. Create your Resume or Portfolio website.
2. Create a repo in Github.
3. Publish it on Github pages.
4. Add the link to your resume.



