

Learn • Code • Create

Let's
Build
Websites!
😊

HTML

for Kids

A Fun and Easy Guide to Learn **HTML**
and Build Your Own Websites



Easy Lessons
for Beginners



Real Examples
& Practice



Fun Projects
to Build



Quizzes &
Challenges



Step-by-Step
Learning

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>My First Page</title>
5   </head>
6   <body>
7     <h1>Hello, Kids!</h1>
8     <p>Welcome to HTML World!</p>
9   </body>
10 </html>
```

You Dream
It, You
Code It!



HTML

</>

10
Modules

50
Lessons

150+
Topics

Fun
Projects

Perfect for
8-15 Years



www.codescompiler.com



Module 1 Lesson 1

What is the Internet?

1

Understanding the Internet

The Internet is a huge network that connects millions of computers and devices all around the world. It helps us share information, learn new things, talk to people, play games and much more!



2

Websites and Web Pages

A website is like a big folder on the Internet. It contains many web pages. Web pages are the pages you see in a browser. They can have text, pictures, videos and links to other pages.

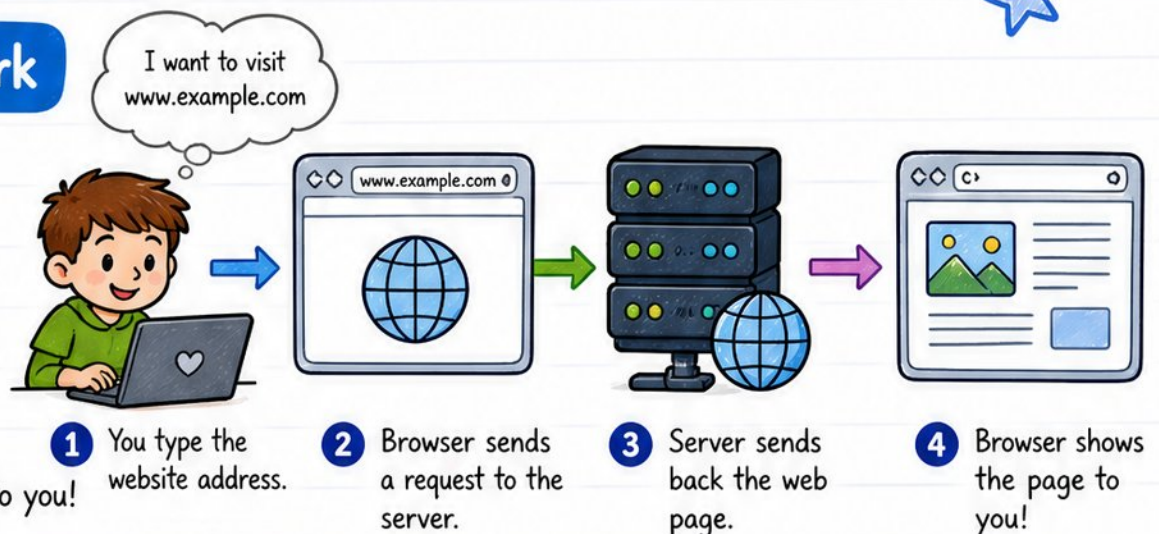


Many Pages,
One Website!

3

How Browsers Work

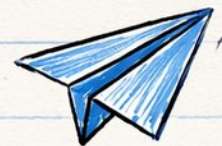
A web browser is a special program that helps you visit websites. You type a website address (URL). The browser sends a request to the server. The server sends back the web page, and the browser shows it to you!





MODULE 1

LESSON 2



Introduction to HTML



1 What is HTML?

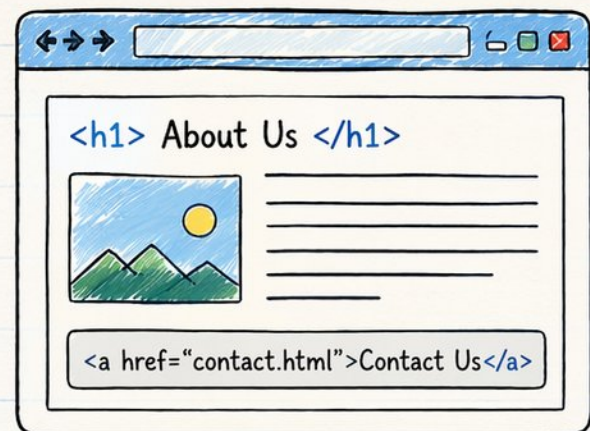
HTML stands for HyperText Markup Language. It is the standard language used to create and design web pages on the internet.



HTML is the building block of every website!

2 Why We Use HTML?

- ✓ It helps us create the structure of web pages.
- ✓ It tells the browser what content to show like headings, paragraphs, images and links.
- ✓ It is the first step to becoming a web developer!



3 Real-Life Examples of HTML

Websites



Every website you visit, like Google, YouTube, or Wikipedia, is built using HTML.

Online Forms



Forms on websites (such as sign-up or login forms) are made with HTML.

News Pages



News articles, images, videos and links on news websites are structured using HTML.

Shopping Sites

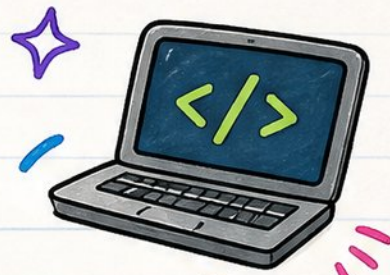


Product pages, prices, images and reviews on shopping websites are created with HTML.





Setting Up for Coding



Module **1**

Lesson **3**

1

Choosing a Text Editor

A text editor is a program where you write your code.

Some great choices are:



VS Code



Notepad++



Sublime Text

```
index.html x
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Hello!</title>
5    </head>
6    <body>
7      <h1>Hello, World!</h1>
8    </body>
9  </html>
```

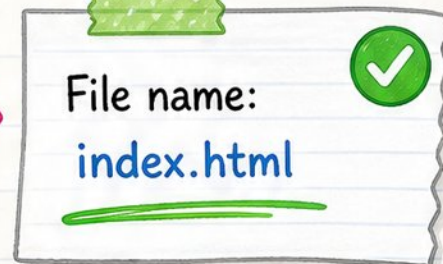


2

Saving HTML Files

Write your code and save it with a **.html** extension.

Example: **index.html**



Always save your files in a folder you can find easily!



3

Opening Files in a Browser

Open your HTML file in a web browser to see your webpage! Right-click the file and select "Open with" → your browser.



You did it!
You're ready to code! ♥



www.codescompiler.com



Module 1

Your First HTML Page

Lesson 4



Let's build your very first web page! ★



1

Creating an HTML File

Open any text editor (like Notepad, VS Code, etc.). Create a new file and save it with a `.html` extension.



New File



Save As



2

Writing "Hello World"

Write the following HTML code in your file. This code displays "Hello World" on the web page.



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>My First Page</title>
5 </head>
6 <body>
   <h1>Hello World!</h1>
</body>
</html>
```

3

Viewing the Output

Open your HTML file in any web browser (like Chrome, Edge, or Firefox). You will see your page with the message "Hello World!"



Tip:

Save your file and refresh the browser if you make changes.



You just created your first web page.

Great job! ★





Understanding HTML Structure



Module 1

Lesson 5



1 HTML Document Layout

An HTML document has a basic layout made of important parts.

- `<html>` The root of the document
- `<head>` Contains information about the page
- `<body>` Contains the content we see on the page

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Page</title>
  </head>
  <body>
    <h1>Hello, World!</h1>
    <p>Welcome to my website.</p>
  </body>
</html>
```

Tells the browser this is an HTML5 document.

Head holds meta data and the title.

Body holds all the visible content.

Closes the HTML document.

2 Opening and Closing Tags

HTML elements are written using tags.

- Opening tag: `<tag>`
- Closing tag: `</tag>`

They tell the browser where the element starts and ends.

Opening Tag

Content

Closing Tag

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

Starts the element

The text or content inside

Ends the element



3 Nesting Elements

Elements can be placed inside other elements. This is called nesting.

★ Always close the inner element before closing the outer element.

```
<div>
  <h2>'My Website</h2>
  <p>Learn <strong>HTML</strong>
    the fun way!</p>
</div>
```

`<div>` is the parent element (outer).

`<h2>` and `<p>` are children of `<div>`.

`<strong>` is inside `<p>`, so it is nested.



Keep Practicing and Keep Building!

www.codescompiler.com



Module:
Module 2

Lesson:
Lesson 1

HTML

Document Skeleton



An HTML document has a basic structure called the skeleton. It has 3 main parts.



1 <html> Tag

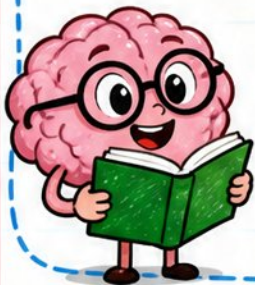


The <html> tag is the root of an HTML page. It wraps around the entire document.

```
<html>
....
....
</html>
```

It tells the browser that this is an HTML document.

2 <head> Tag



The <head> tag contains information about the document, like the title, links to CSS, and more.

```
<html>
  <head>
    ....
  </head>
  <body>
    ....
  </body>
</html>
```

It is not displayed on the webpage. It stays behind the scenes.

3 <body> Tag



The <body> tag contains all the content that is visible on the webpage, like text, images, links, and more.

```
<html>
  <head>
    ....
  </head>
  <body>
    ....
  </body>
</html>
```

This is the part users see and interact with!



www.codescompiler.com



Page Information

Module 2

Lesson 2

1

<title> Tag

The **<title>** tag is used in HTML to give a title to your web page.

It helps browsers and users know what your page is about.

Example Code

```
<!DOCTYPE html>
<html>
  <head>
    <title>My First Page</title>
  </head>
  <body>
    <h1>Welcome!</h1>
  </body>
</html>
```



2

Browser Tab Names

The text inside the **<title>** tag appears on the browser tab.

It helps users easily identify your page among many tabs.



3

Organizing Page Details

Good page information helps keep your website neat, clear, and easy to use.

It improves user experience and helps with search engines too!



Clear Title

Use a title that describes your page.



Better Search Results

Good titles help search engines understand your page.



Happy Users

Users can find and remember your page easily.



Module 2

Lesson 3

Comments in HTML

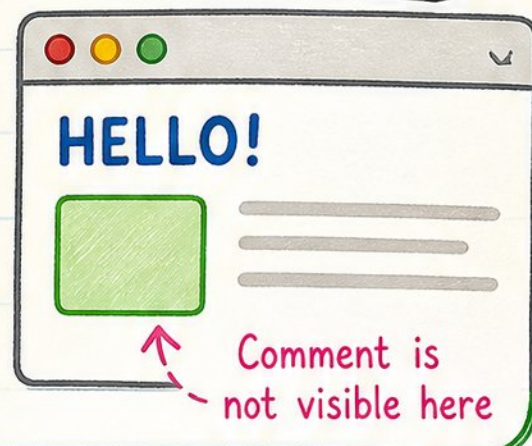


1

What are Comments?



Comments are notes you add in your HTML code to explain what it does. They are not shown on the web page.



2

Writing Comments

In HTML, comments are written between these symbols:

```
<!-- your comment here -->
```

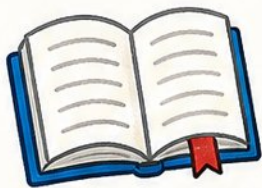
Example:

```
<!-- This is a comment -->  
<h1>Welcome to My Website!</h1>
```



3

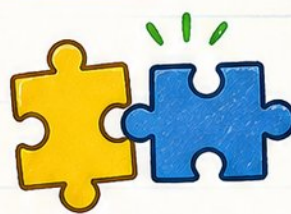
Why Comments are Useful



They explain what the code does.



They help you remember your code later.



They help others understand your code.



You can comment out code to test things easily.

www.codescompiler.com

Module 2

Lesson 4



Understanding Elements



Let's learn the building blocks of the web!

1 Tags vs Elements

Tags are the starting and ending parts of an element.



Example: Paragraph



An element = start tag + content + end tag.

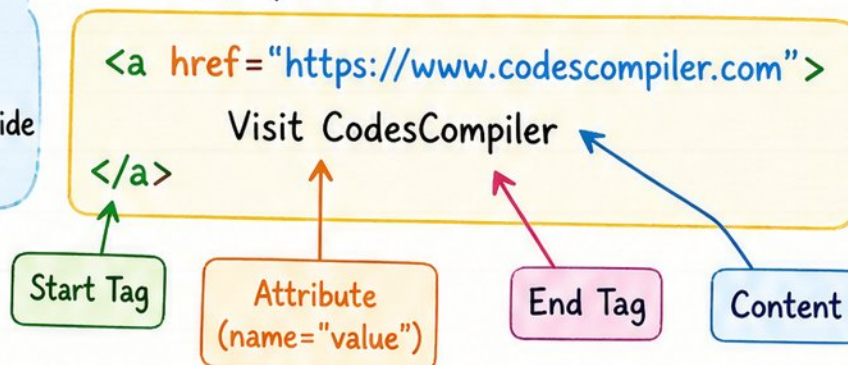
Element Example
`<p>Hello, World!</p>`

2 Attributes Introduction

Attributes give extra information about an element. They are written inside the start tag.



Example: Link with an attribute



Key Points

- Attributes are written in the start tag.
- Format: `name="value"`

3 Common HTML Elements

Here are some commonly used HTML elements:

`<h1>` to `<h6>`

H

Define headings.
`<h1>` is the most important.

```
<h1>Hello</h1>
```

`<p>`



Defines a paragraph of text.

```
<p>Hello!</p>
```

`<a>`



Creates a link to another page or website.

```
<a href="url">Link</a>
```

`<img>`



Displays an image.

```

```

`<ul>`



Creates an unordered (bullet) list.

```
<ul>...</ul>
```

`<div>`



A container used to group other elements.

```
<div>...</div>
```



www.codescompiler.com

`</>`





PRACTICE PROJECT



MODULE: Module 2



LESSON: Lesson 5

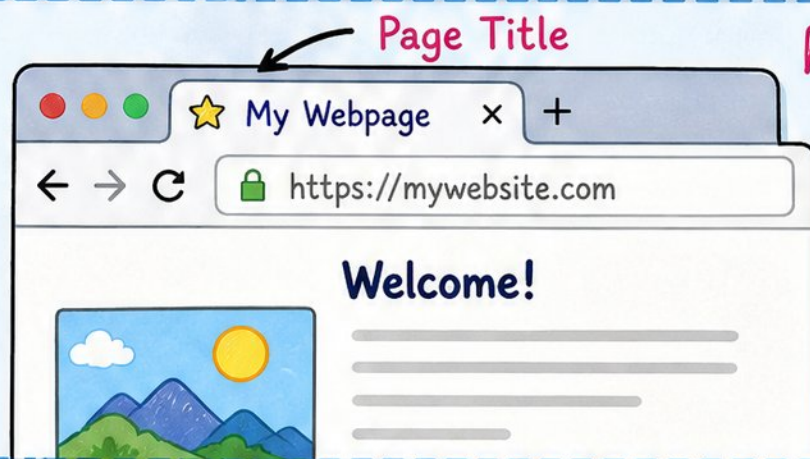
1 CREATE A PERSONAL WEBPAGE

Build your own webpage about yourself! Share your hobbies, favorite things, and more with the world. 



2 ADD A PAGE TITLE

Every webpage needs a title! It appears in the browser tab and tells visitors what your page is about.



3 USE BASIC STRUCTURE CORRECTLY

A webpage has a basic structure. Using it correctly helps your webpage look neat and work properly.

```
<!DOCTYPE html>
<html>
  <head>
    <title>My Webpage</title>
  </head>
  <body>
    <h1>Welcome!</h1>
    <p>This is my webpage.</p>
  </body>
</html>
```

<html>
The root of the page

<head>
Contains info about the page

<title>
The page title

<body>
The content of the page



www.codescompiler.com



Module:
Module 3

Lesson:
Lesson 1

Headings

Headings are used to title and organize content on a web page. HTML has 6 levels of headings:



THE 6 LEVELS OF HEADINGS

`<h1>`

This is Heading 1

Largest and most important heading

`<h2>`

This is Heading 2

Second level heading

`<h3>`

This is Heading 3

Third level heading

`<h4>`

This is Heading 4

Fourth level heading

`<h5>`

This is Heading 5

Fifth level heading

`<h6>`

This is Heading 6

Smallest and least important heading



IMPORTANCE OF HEADINGS



- ✓ Make content easy to read.
- ✓ Show the structure of a page.
- ✓ Help search engines understand your content.
- ✓ Improve accessibility for everyone.



ORGANIZING CONTENT

Use headings to break your content into clear sections.



`<h1>` Main Title

`<h2>` Section

`<h3>` Subsection

... and so on!



PRO TIP!

Use only one `<h1>` per page. Then use `<h2>` to `<h6>` in order for better structure.



Module 3

Lesson 2

Paragraphs

1

<p> Tag

The `<p>` tag in HTML is used to create a paragraph. Browsers automatically add some space before and after a paragraph.



Think of `<p>` as starting a new paragraph!

```
<p>This is a paragraph.  
It has some text inside it.  
</p>
```

Simple and powerful!

2

Writing Text Content

Anything between the opening `<p>` and closing `</p>` tags is the **text content**. You can write words, sentences, or even emojis!

```
<p>  
I love learning HTML!  
It is fun and easy.  
Let's build amazing websites! 😊  
</p>
```



3

Formatting Paragraphs

You can **format** paragraphs using different HTML tags to make your text look better and more interesting!



What It Does	Example
Make text bold	<code><p>Bold Text</p></code>
Make text italic	<code><p><i>Italic Text</i></p></code>
Underline text	<code><p><u>Underlined Text</u></p></code>
Highlight text	<code><p><mark>Highlighted Text</mark></p></code>
Change text color	<code><p style="color: blue;">Blue Text</p></code>
Change text size	<code><p style="font-size: 20px;">Big Text</p></code>



Paragraphs help organize your content and make your web pages **easy to read**!

Practice more,
Code better,
Build the future!





Text Formatting

Module: 3



Lesson: 3

1

Bold Text

Bold text is **thick** and **dark**.

It is used to make **important words** stand out.



Example:

This is **important!**



2

Italic Text

Italic text is *slanted to the right*.

It is used for *emphasis*, titles, or special words.



Example:

This is special!



3

Underlined Text

Underlined text has a line under the words.

It is used to show links or important information.



Example:

Visit our website here.



www.codescompiler.com



Line Breaks and Horizontal Lines



Lesson 4



1

 Tag

The `<br>` tag is used to break a line and move the text to the next line.



EXAMPLE

```
Hello Friends!<br>
Welcome to<br>
Codes Compiler!
```

OUTPUT

```
Hello Friends!
Welcome to
Codes Compiler!
```

2

<hr> Tag

The `<hr>` tag creates a horizontal line to separate content.



EXAMPLE

```
Welcome to Codes Compiler!
<hr>
Happy Coding!
```

OUTPUT

```
Welcome to Codes Compiler!
_____
Happy Coding!
```

3

Creating Sections

Use `<hr>` to divide your content into sections. It makes your page neat and easy to read!

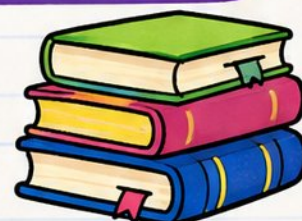


EXAMPLE

```
<h2>About Us</h2>
We love teaching kids
to code.
<hr>
<h2>Our Mission</h2>
Make coding fun and
easy for everyone!
<hr>
<h2>Join Us</h2>
Let's build the future
together.
```

OUTPUT

```
About Us
We love teaching kids
to code.
_____
Our Mission
Make coding fun and
easy for everyone!
_____
Join Us
Let's build the future
together.
```



Mini Project

Module: Module 3

Lesson: Lesson 5

In this mini project, you will create your own "About Me" page using headings, paragraphs and text formatting.

1



Create "About Me" Page

Design a web page about yourself. Include your name, a photo or avatar, your hobbies, favorite things and anything you want to share!

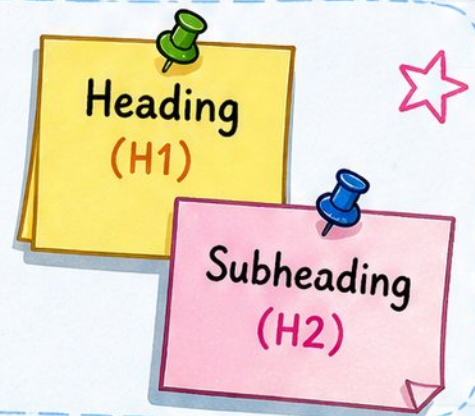


2



Add Headings and Paragraphs

Use headings (like H1, H2, H3) to organize your page. Write short paragraphs to tell more about yourself.



3

★ HELLO!

This is my **About Me** page.
I love **coding** and **drawing**.
I enjoy learning new things
every day!

Use Text Formatting

Make your page look awesome using text formatting:

- ✓ **Bold** for important words
- ✓ *Italic* for style
- ✓ **Colors** to highlight
- ✓ Links to other pages



Module:
Module 4

Lesson:
Lesson 1

Ordered Lists



Ordered Lists show items in a **specific order**!

1

Numbered Lists

Ordered lists are used when the **order** of items is important.

They are shown with numbers (1, 2, 3, ...).

My Daily Plan

1. Wake up
2. Brush my teeth
3. Eat breakfast
4. Go to school
5. Do my homework



2

 Tag

The **** tag is used to create an ordered list in HTML.

It tells the browser to **number** the list items.

```
<ol>
  <li>Wake up</li>
  <li>Brush my teeth</li>
  <li>Eat breakfast</li>
</ol>
```

**** opens the list and **** closes the list.

3

List Items

Each item in the ordered list is written using the **** tag (list item).

The browser numbers them **automatically**.

```
<ol>
  <li>Apples</li>
  <li>Bananas</li>
  <li>Cherries</li>
</ol>
```

1. Apples
2. Bananas
3. Cherries



Remember:

Ordered lists help keep information clear and easy to follow!



Module 4

Lesson 2

Unordered Lists

Let's learn about unordered lists in **HTML**!



1

Bullet Lists

Unordered lists are used to group items that do not have a specific order.

Each item is marked with a **bullet** (•).



Example:

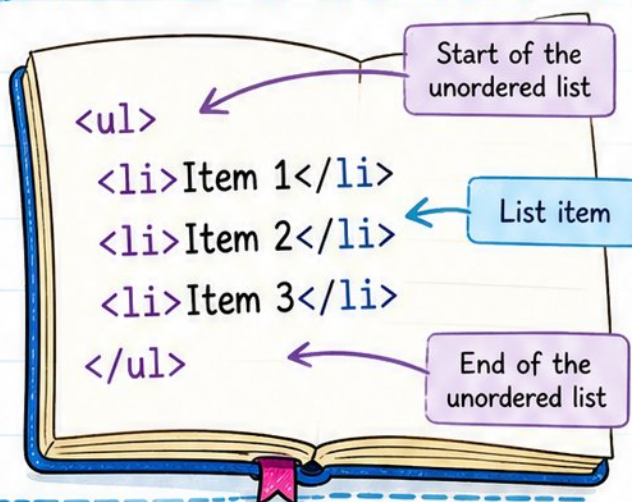
```
<ul>
  <li>Apple</li>
  <li>Banana</li>
  <li>Mango</li>
  <li>Orange</li>
</ul>
```

2

 Tag

The **** tag is used to create an unordered list.

Each item in the list is written using the **** tag.



Think of **** as a box that holds your list items!



3

Nesting Lists

You can place a list inside another list item. This is called **nesting**.

Great for creating sub-lists!



Example:

```
<ul>
  <li>Programming</li>
  <ul>
    <li>HTML</li>
    <li>CSS</li>
    <li>JavaScript</li>
  </ul>
  <li>Designing</li>
  <ul>
    <li>Photoshop</li>
    <li>Illustrator</li>
  </ul>
</ul>
```



Module 4

Lesson 3

Description Lists



Description lists are used to show pairs of terms and their definitions—perfect for explaining things!

1 <dl> Tag

The `<dl>` tag is the container for a description list. It wraps all the terms and definitions.

```
<dl>
  <dt> Term</dt>
  <dd> Definition</dd>
  <dt> Term</dt>
  <dd> Definition</dd>
</dl>
```

Think of `<dl>` as the big box that holds everything!



2 Terms and Definitions

Inside `<dl>`, we use:

`<dt>` for the term (the word or item).

`<dd>` for the definition (the explanation).

`<dt>` → Term

Example: HTML

`<dd>` → Definition

Example: The standard markup language for creating web pages.

Term comes first, then its definition!



3 Real-Life Examples

Description lists are great for FAQs, glossaries, recipes, and more! 😊

Example: Favorite Fruits

```
<dl>
  <dt> Apple</dt>
  <dd> A sweet, crunchy fruit that keeps the doctor away.</dd>
  <dt> Banana</dt>
  <dd> A soft, tasty fruit that gives you energy.</dd>
</dl>
```



Sweet & crunchy!



Tasty & energizing!

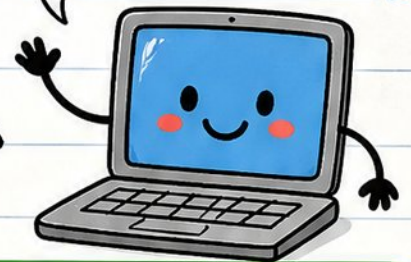
Module 4

Combining Lists

Lesson 4



Let's learn how
to combine lists
in fun ways!



1 Mixed Lists

A mixed list can have different types of items—like numbers, words, and even symbols!

It's like a collection of different things in one list.

Example:

[1, "Apple", 3.14, 😊, True]

Number

Text

Decimal

Symbol

Boolean



2 Creating Menus

Lists help us create menus by grouping choices together.

Perfect for apps, games, or even your favorite restaurant!



≡ Pizza Menu ≡

🍕	Margherita	\$8.50
🍕	Veggie Supreme	\$9.75
🍕	Pepperoni	\$9.25
🥤	Soft Drink	\$2.00
🍪	Chocolate Cookie	\$1.50

Organized
choices make it
easy to pick!



3 Organizing Information

Lists help us keep information neat and easy to understand.

You can combine and sort lists to make everything clear and useful!



Classroom Supplies

- Pencils
- Erasers
- Notebooks
- Glue Stick
- Scissors



Sorted by Type

Writing: Pencils, Notebooks
Editing: Erasers
Craft: Glue Stick, Scissors





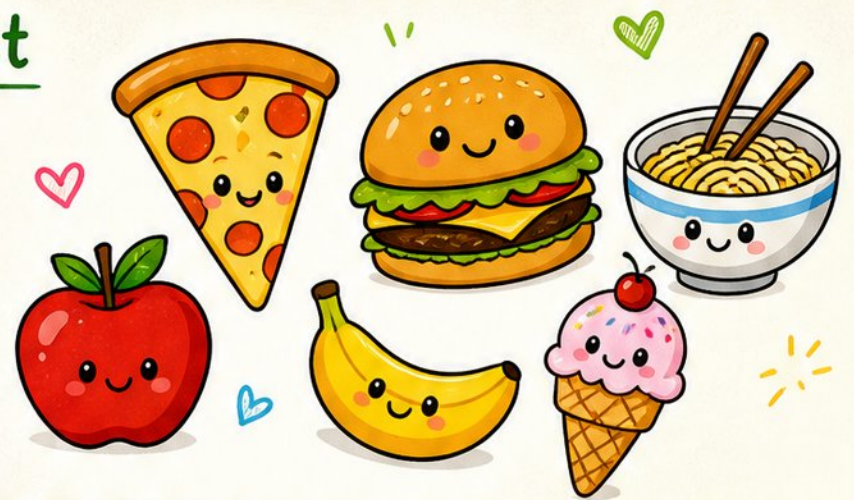
Mini Project

Module:
Module 4

Lesson:
Lesson 5

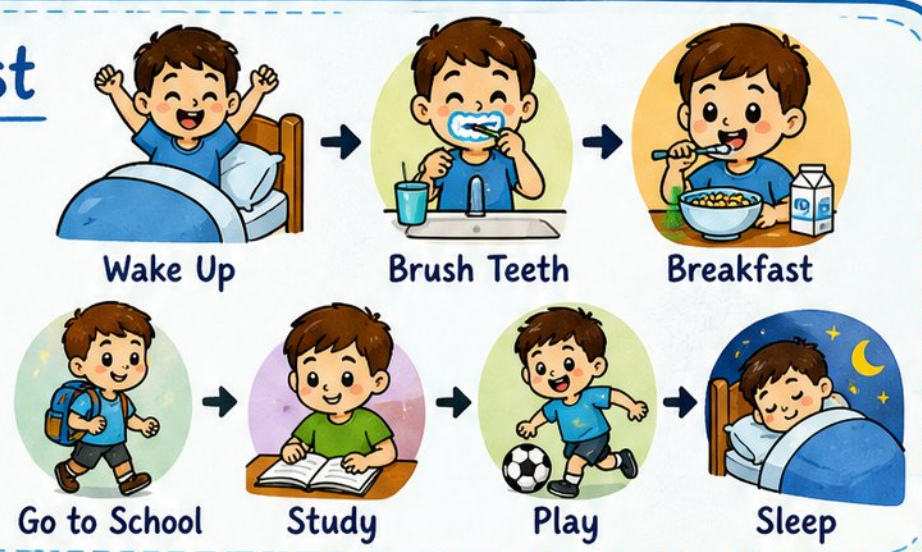
1 Favorite Foods List

1. _____
2. _____
3. _____
4. _____
5. _____



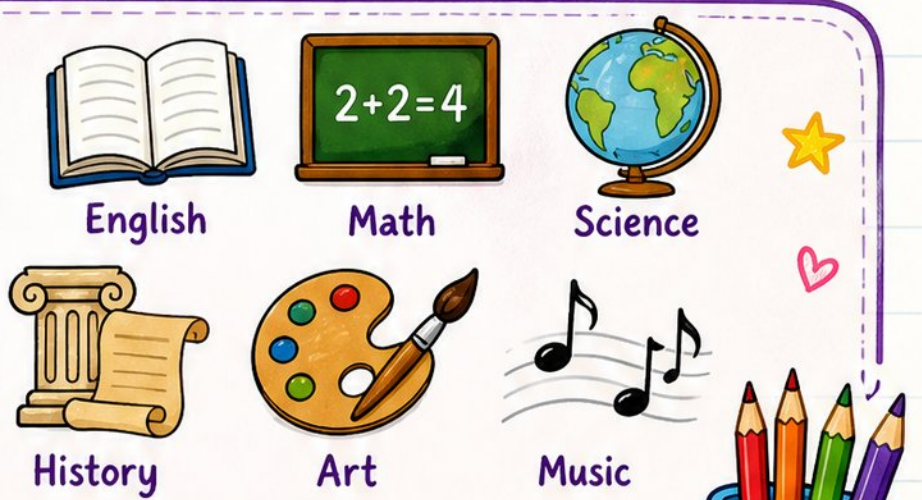
2 Daily Routine List

1. _____
2. _____
3. _____
4. _____
5. _____



3 School Subjects List

1. _____
2. _____
3. _____
4. _____
5. _____



Module 5

Lesson 1

Images in HTML

Let's add pictures to our web pages!



1

 Tag

The `<img>` tag is used to embed an image in an HTML page.

```

```



Remember:

`<img>` is an empty tag. It doesn't have a closing tag.

2

Image Sources

The `src` attribute tells the browser where the image is located.

Local Image



You can use:

Image is in the same folder as your HTML file.

Example:

```

```

Online Image



Image is on another website (Internet).

Example:

```

```

3

Alternate Text

The `alt` attribute provides text that describes the image.

- ✓ It shows if the image can't be loaded.
- ✓ It helps people using screen readers.
- ✓ It's good for **SEO** (search engines).



`alt="Mountain with trees"`

Screen reader: "Mountain with trees"

Module:
Module 5

Lesson:
Lesson 2

Image Attributes



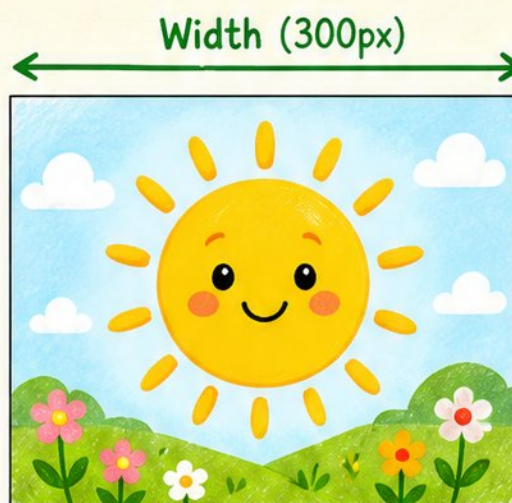
1

Width and Height

These attributes set the size of an image on a webpage.

```

```



Bigger numbers make the image bigger!

2

Image Alignment

The **align** attribute controls how an image sits with the text.

align="left"



Image on the left

align="center"



Image in the center

align="right"

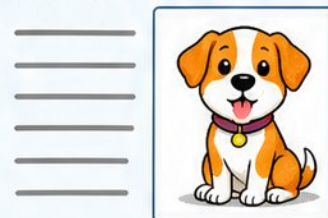


Image on the right

3

Responsive Basics

Make images look good on all devices!

```

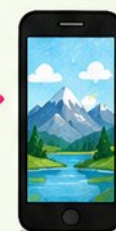
```



Laptop



Tablet



Phone

The image adapts to fit any screen!



www.codescompiler.com



Audio Elements

Module 5

Lesson 3

1 <audio> Tag

The `<audio>` tag is used to add sound or music to a webpage. It tells the browser where the audio file is.

```
<audio src="song.mp3">
Your browser does not
support the audio element.
</audio>
```



2 Playing Sounds

When the webpage loads, the audio can start playing. You can play a short clip, a song, or even background music!



Audio formats that work:

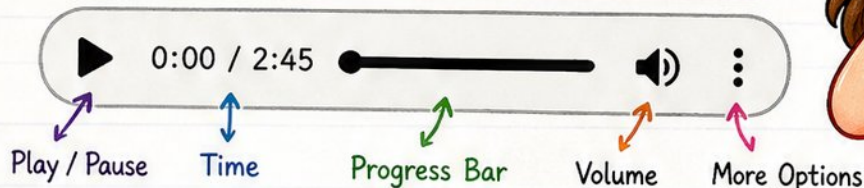
- ☒ MP3
- ☒ OGG
- ☒ WAV

3 Audio Controls

You can add controls so users can play, pause, adjust volume, and more. This makes it easy for everyone to listen!

Add controls like this:

```
<audio src="song.mp3" controls>
Your browser does not
support the audio element.
</audio>
```



Sound makes websites fun!

Let your webpages be heard!

Module 5



Video Elements

Lesson 4



1 <video> Tag

The `<video>` tag is used to embed a video in an HTML page.



Basic Syntax:

```
<video width="640" height="360"
controls>
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the
  video tag.
</video>
```



2 Adding Videos

Use the `<source>` tag inside the `<video>` tag to add video files.



You can add multiple formats for better compatibility!

Example:

```
<video width="640" height="360" controls>
  <source src="video.mp4" type="video/mp4">
  <source src="video.webm" type="video/webm">
  <source src="video.ogv" type="video/ogg">
  Your browser does not support the video tag.
</video>
```



3 Video Controls

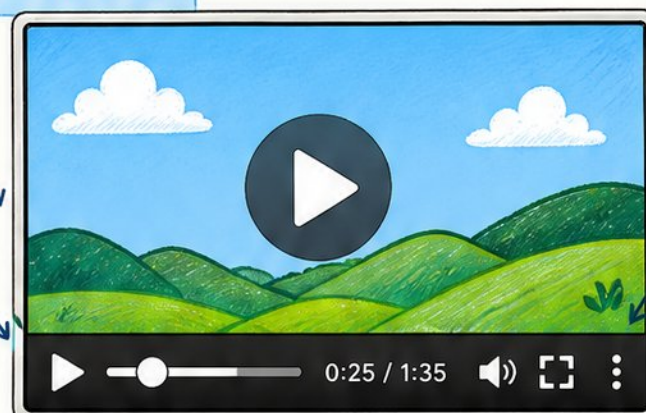
The `controls` attribute adds play, pause, volume, and more buttons to the video.



With Controls:

Play / Pause

Progress Bar



Volume

Fullscreen

More Options





Mini Project



Module: Module 5

Lesson: Lesson 5

1

Create a Photo Gallery

Add your favorite photos and make your own beautiful gallery!



2

Add Audio

Add music or sound to make your project more exciting!



3

Add Video

Add videos to show your ideas in action and tell your story!



www.codescompiler.com



Hyperlinks

Module:
Module 6



Lesson:
Lesson 1

Hyperlinks help us connect one page to another.
They make the web interactive and fun to explore!

1

<a> Tag

The <a> tag is used to create a hyperlink in HTML.
It tells the browser where to go when we click.

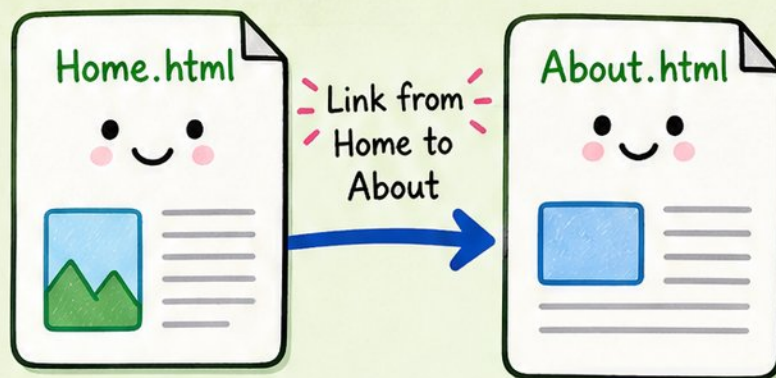


This is
an <a>
tag!

2

Linking Pages

We can use hyperlinks to link one page of our website to another page.
It helps users navigate easily within our site.



3

External Links

We can link to other websites on the internet using external links.
These links open in a new tab or window.



This link
takes you
to another
website!

www.codescompiler.com

Internal Navigation

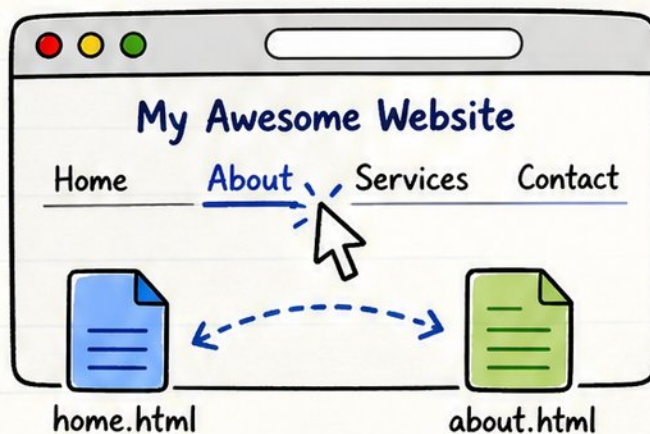
Module:
Module 6

Moving around your website
using links and paths!

Lesson:
Lesson 2

1 Linking Within a Website

Internal links help users move from one page to another on the same website. They make your website easy to explore!

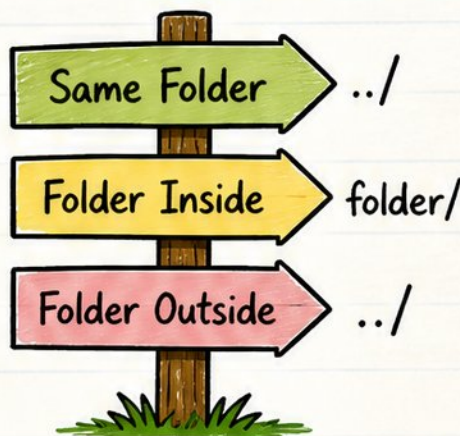


Example:

From home.html, clicking "About" uses a link to about.html.

2 Relative Paths

Relative paths tell the browser how to reach a file from the current page's location. They do not use the full website address!

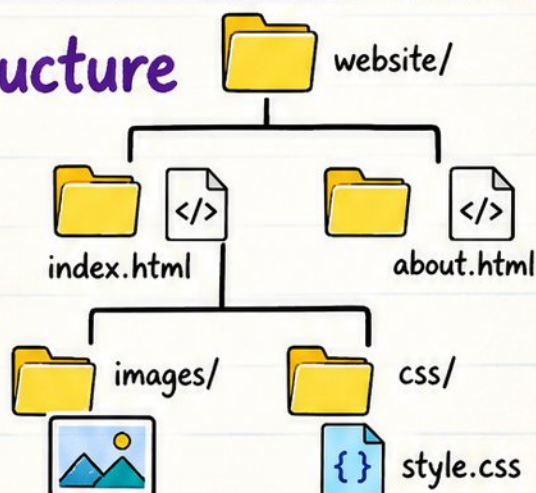


Examples:

- `../contact.html` (same folder)
- `images/logo.png` (inside images folder)
- `../about.html` (one folder up)

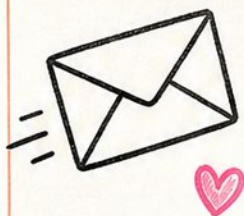
3 Folder Structure

A good folder structure keeps your website organized and makes links easier to manage and understand!



Why it's important:

- ✓ Easy to find files
- ✓ Cleaner links
- ✓ Better for teamwork
- ✓ Great for website growth!



Email Links



Module: Module 6

Lesson: Lesson 3



1

Mailto Links

A mailto link opens the user's email program and starts a new email to a specific address.

Example:

```
<a href="mailto:hello@codescompiler.com">  
Email Us</a>
```



Opens your email!

2

Contact Buttons

Contact buttons make it easy for people to reach you. When clicked, they open an email using a mailto link.

Example:

```
<a class="btn" href="mailto:hello@codescompiler.com">  
Contact Us</a>
```



3

Useful Link Types

Besides email links, websites use many types of links to help users.



Website Links
Go to another web page.



Download Links
Download files like PDFs.



Map Links
Open locations in maps.



Phone Links
Call a phone number.



Links make the web interactive and helpful!



www.codescompiler.com





Navigation Menus



Module: Module 6

Lesson: Lesson 4



1

Menu Creation

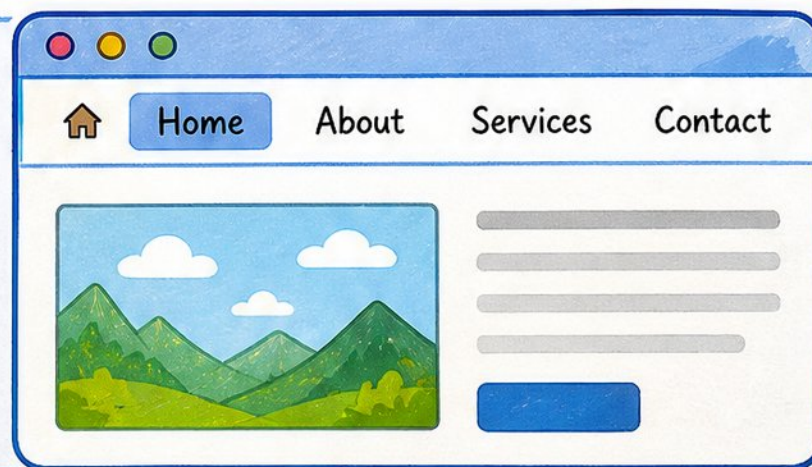
Menu creation is the first step! We add links to important pages like **Home**, **About**, **Services**, and **Contact** so users can find what they need.



2

Navigation Bars

Navigation bars show the menu on top (or side) of a website. They help users move from one page to another quickly.

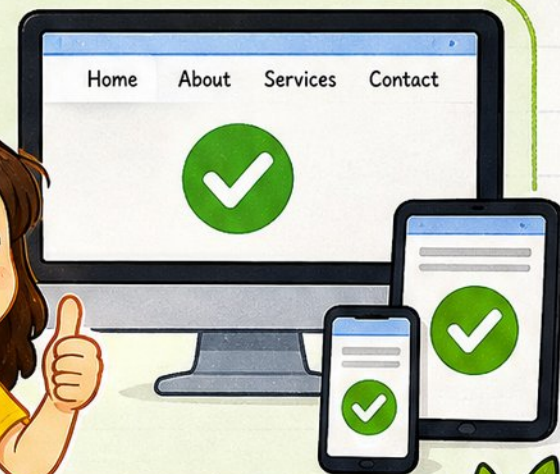


3

User-Friendly Navigation

A good menu is **simple**, **clear**, and **easy** to use. It should be consistent on all pages and work well on all devices!

SIMPLE
CLEAR
EASY



www.codescompiler.com





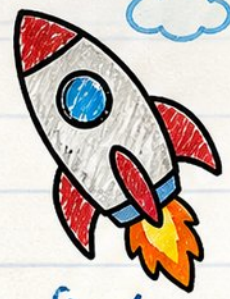
Module 6



Lesson 5

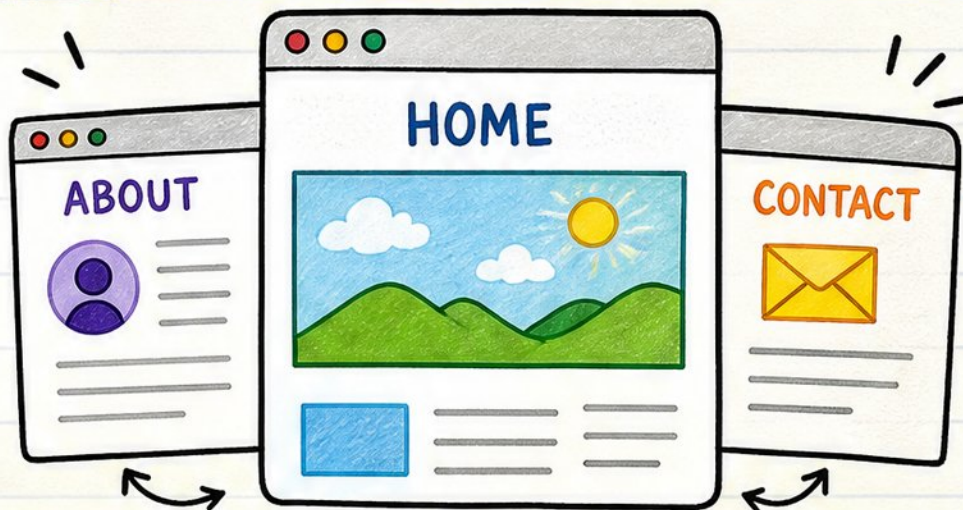


Mini Project



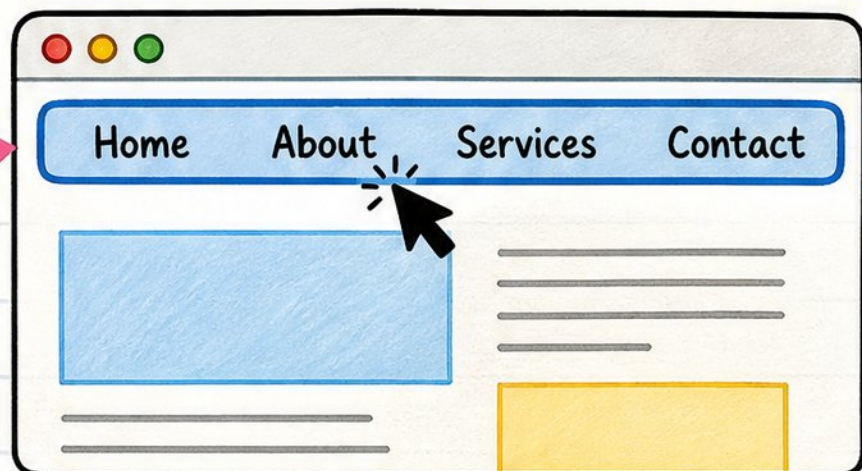
1 Multi-Page Website

A website can have multiple pages to share different information.



2 Navigation Menu

A navigation menu helps users move between pages easily. It is usually placed at the top of the website.

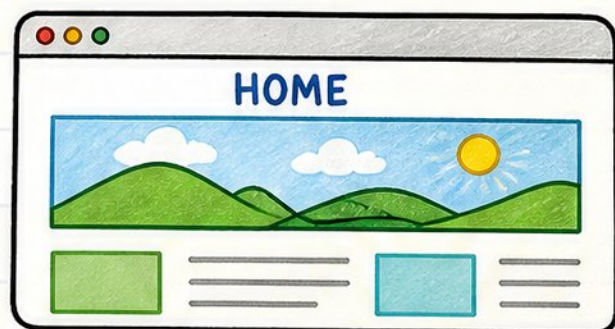


3 Home and About Pages



Home Page

The home page is the first page visitors see. It gives a quick overview of your website.



About Page

The about page tells visitors about you or your website.



www.codescompiler.com



Module 7

Introduction to Tables

Lesson 1

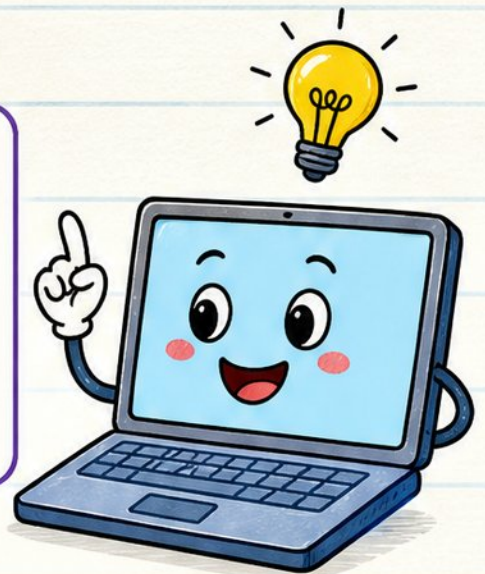
1

<table> Tag

The `<table>` tag is used to create a table in HTML. Everything inside `<table>...</table>` is part of the table.

Example

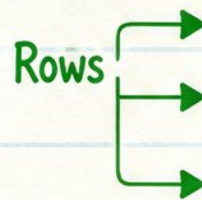
```
<table>
  <tr>
    <td>Data</td>
  </tr>
</table>
```



2

Rows and Columns

Rows run **horizontally** across the table.
Columns run **vertically** down the table.



Name	Age	City
Ava	10	Delhi
Ben	11	Mumbai
Cara	12	Kolkata

Columns

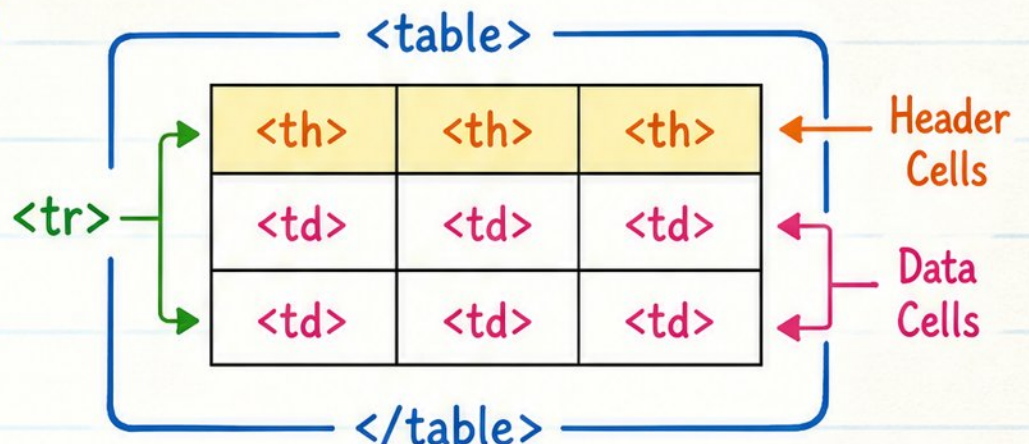


3

Table Structure

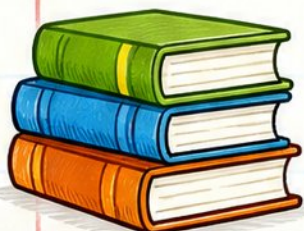
A table is built using these main tags:

- ★ `<table>` : Defines the table
- ★ `<tr>` : Defines a row
- ★ `<th>` : Defines a header cell
- ★ `<td>` : Defines a data cell
- ★ `</table>` : Ends the table



Remember:

Tables help us organize information in rows and columns!



www.codescompiler.com



Module 7

Lesson 2

Table Components



1

<tr> Rows

The `<tr>` tag defines a row in an HTML table. It holds one or more cells.

`<tr>` →

`<tr>` →

`<tr>` →

Name	Age	City
Alice	10	New York
Bob	11	London
Charlie	12	Tokyo

2

<td> Cells

The `<td>` tag defines a cell in a table. It contains the data (content) of the table.

Name	Age	City
Alice	10	New York
Bob	11	London
Charlie	12	Tokyo

Each box is a `<td>` cell.

3

<th> Headers

The `<th>` tag defines a header cell in a table. It is used to label columns (or rows).

Name	Age	City
Alice	10	New York
Bob	11	London
Charlie	12	Tokyo

These are `<th>` header cells.



Quick Tip! ★

A table is built using:

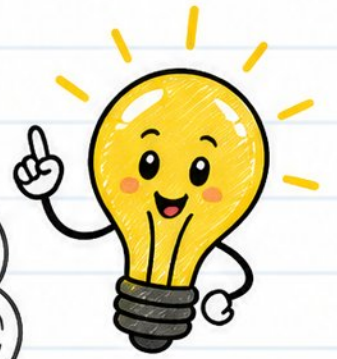
```
<table> <tr> <th>/<td> </tr> </table>
```



Module 7

Lesson 3

Table Styling Basics



We can style HTML tables to make them look **neat**, **clean**, and **easy to read**!

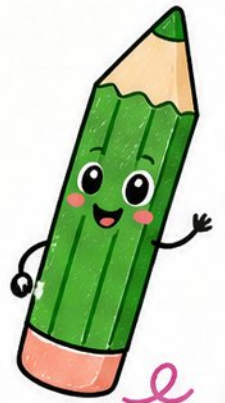
1 Borders

Borders are the lines around the table and cells. They help separate the content.

Name	Age	City
Ava	10	Delhi
Rohan	12	Mumbai
Isha	11	Pune

Example:

```
table {  
  border: 2px solid  
    #2e7d32;  
  border-collapse:  
    collapse;  
}
```



2 Cell Spacing

Cell spacing adds space between the cells. It makes the table look more **open** and **clean**.

Name	Age	City
Ava	10	Delhi
Rohan	12	Mumbai
Isha	11	Pune

Example:

```
table {  
  border-collapse:  
    separate;  
  border-spacing: 10px;  
}
```



3 Table Layout

Table layout controls how the columns are set. It helps the table fit better on the page.

Auto (default)

Columns adjust to content.

Subject	Marks
Maths	95
Science	88
English	76

Fixed

Columns stay equal width.

Subject	Marks
Maths	95
Science	88
English	76

Example:

```
table {  
  table-layout: fixed;  
  width: 100%;  
}
```



Quick Tip!

Good table styling makes data **easy** to read and your pages look awesome!

Remember:

Clean tables = Happy Readers!



Module: Module 7

Lesson: Lesson 4

Advanced Tables

Let's
take our
tables to the
next level!

1 Rowspan

Rowspan lets a cell span across multiple rows (up and down). It helps to group related information together!



Example:

Category	Item	Quantity
Fruits	Apple	10
	Banana	20
	Mango	15
Vegetables	Carrot	12
	Potato	18

"Fruits" cell spans 3 rows.
"Vegetables" cell spans 2 rows.

2 Colspan

Colspan lets a cell span across multiple columns (left and right). It's great for creating headings or titles!



Example:

Monthly Sales Report			
Month	Sales (in Units)		
	Product A	Product B	Product C
January	120	150	100
February	130	160	110
March	140	170	120

"Monthly Sales Report" cell spans 4 columns.

3 Organizing Data

Well-organized tables make data easy to read and understand. Use clear headings, group related data, and keep it neat!



Example of Organized Data:

Student	Grade	Math	Science	English	Total
Ava	5	90	85	92	267
Ben	5	88	90	85	263
Cara	5	95	92	93	280
Dan	5	80	78	82	240

★ Clear headings + neat data = easy understanding!

Mini Project

Module: **Module 7**

Lesson: **Lesson 5**

1 School Timetable



Time / Day	Mon	Tue	Wed	Thu	Fri
8:30 – 9:30	English	Maths	Science	English	Maths
9:30 – 10:30	Maths	English	Social Studies	Maths	Science
10:30 – 11:00	BREAK				
11:00 – 12:00	Science	Social Studies	English	Science	Social Studies
12:00 – 1:00	LUNCH				
1:00 – 2:00	Computer	Art	Maths	Computer	Art
2:00 – 3:00	Sports	Library	Music	Value Education	Sports

2 Marks Table



Subject	Max Marks	Marks Obtained	Percentage
English	100	88	88%
Maths	100	92	92%
Science	100	85	85%
Social Studies	100	78	78%
Computer	100	95	95%
Art	100	90	90%
Total	600	528	88%

3 Weekly Planner



Great Work! ♥

Monday

- School
- Homework
- Reading



Tuesday

- School
- Maths Practice
- Draw a Picture



Wednesday

- School
- Science Project
- Play Outside



Thursday

- School
- Computer Practice
- Music Practice



Friday

- School
- Revise Lessons
- Help at Home



Saturday

- Sports
- Hobby Time
- Watch Educational Video



Sunday

- Family Time
- Reading
- Plan for Next Week





Introduction to Forms



Module:
Module 8

Lesson:
Lesson 1



1 <form> Tag

The `<form>` tag is used to create an HTML form. It acts as a container that holds all the form elements like text boxes, buttons, checkboxes, and more.

```
<form action="submit.html"
      method="post">
  <!-- form elements -->
</form>
```



2 Why Forms Matter

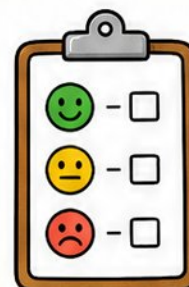
Forms help websites talk to users! They let people send information, like signing up, logging in, giving feedback, or searching for something. Without forms, many websites wouldn't work!



Sign Up



Log In



Feedback



Search

3 Form Structure

A form usually has these parts:

- 1 `<form>` - starts the form
- 2 Form elements - input, select, textarea, button, etc.
- 3 Attributes - action, method (tells where and how to send data)
- 4 `</form>` - ends the form

Example

```
<form action="submit.html" method="post">
  <label for="name">Name:</label>
  <input type="text" id="name" name="name"><br>
  <label for="email">Email:</label>
  <input type="email" id="email" name="email"><br>
  <label for="msg">Message:</label><br>
  <textarea id="msg" name="message"></textarea><br>
  <button type="submit">Submit</button>
</form>
```

1

3

2

4



Forms make
websites
interactive!



www.codescompiler.com



Module:
Module 8

Lesson:
Lesson 2

Input Fields



Input fields help us enter **information** in websites and apps!

1 Text Inputs



Text inputs let us type letters, words, and sentences.

Example:

Hello, World! |

Used for:

Names, emails, addresses, search boxes, and more.



2 Password Inputs



Password inputs hide what we type using dots or asterisks to keep our information safe.

Example:

.....

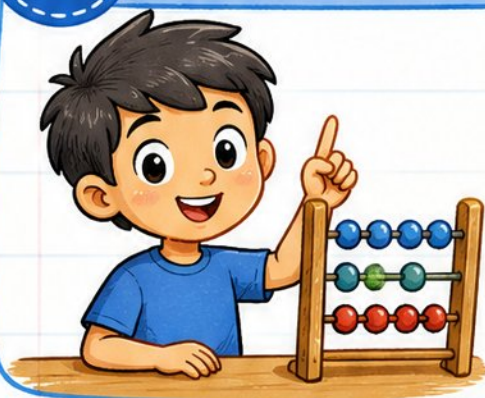


Used for:

Login pages, accounts, and private information.



3 Number Inputs



Number inputs let us enter numbers. We can usually increase or decrease the value.

Example:

25



Used for:

Age, quantity, price, marks, and more.



www.codescompiler.com



Selection Controls

Module: **Module 8**

Lesson: **Lesson 3**

Selection controls let users choose one or more options from a list. They make forms interactive and fun!

1



Radio Buttons

Use radio buttons when the user can select **ONLY ONE** option from a list.

Example:

Choose your favorite color:

- ☒ Blue
- ☐ Green
- ☐ Red



Only one circle can be selected at a time!

2



Checkboxes

Use checkboxes when the user can select **ONE, SOME, or ALL** options from a list.

Example:

Pick your hobbies:

- ☒ Reading
- ☒ Drawing
- ☐ Gaming
- ☒ Music



You can choose any number of options!

3



Dropdown Menus

Use dropdown menus when there are many options and the user can select **ONLY ONE** from a list.

Example:

Choose your grade:

Select grade ▼

Select grade

Grade 1

Grade 2

Grade 3

Grade 4

Grade 5



Great for long lists—keeps forms clean!

Remember: Choose the right control to make your form easy to use and understand!

Module 8

Lesson 4

Buttons and Submission

1 Submit Button

The Submit button sends the information from a form to the server.

It is used when the user has filled the form and wants to save or send it.

Name:
Email:
Message:

SUBMIT



2 Reset Button

The Reset button clears all the data from the form fields.

It helps the user start over and remove any mistakes.

Name:
Email:
Message:

RESET

Name:
Email:
Message:

3 User Interaction

Buttons help users interact with forms on webpages.

They make websites easy to use, helpful and interactive!



1. User fills the form.

2. Clicks Submit to send data.

3. Data is sent to the server.

4. Success message is shown!



www.codescompiler.com





Mini Project



Module: Module 8

Lesson: Lesson 5

1

Student Registration Form



Full Name:



Date of Birth:



Gender:

☐ Male

☐ Female

☐ Other



Email:



Class / Grade:

Address:

2

Contact Form



Name:



Email:



Phone Number:



Message:

3

Feedback Form



How would you rate this lesson?



Excellent



Good



Average



Poor



Very Poor

What did you like the most?

Any suggestions for improvement?

Would you recommend this lesson to others?

☐ Yes

☐ No

☐ Maybe

Semantic Elements



Semantic elements in HTML help us give meaning to the different parts of a web page. They make our code **easier to read and understand!**

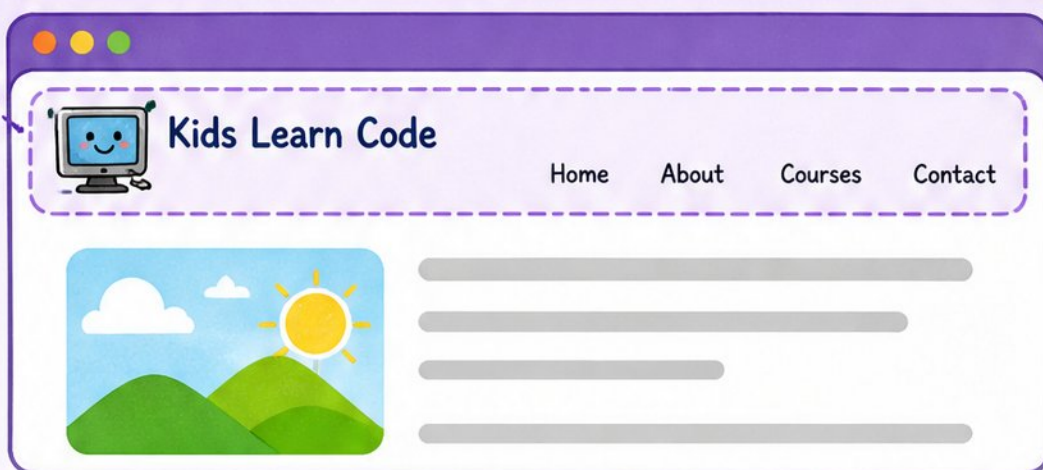
HTML

<header>

Represents the introductory content at the top of the page or section.



It usually contains a logo, website name, heading, or navigation menu.



<main>

Represents the main content of the page.



This is where the unique and important content of the page lives.

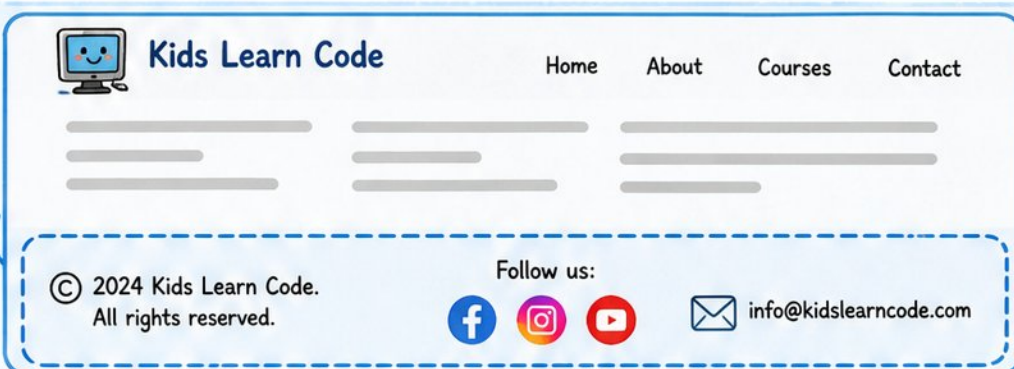


<footer>

Represents the footer content at the bottom of the page or section.



It usually contains copyright info, links, or contact details.



Remember:

Using semantic elements makes websites more accessible for everyone and helps search engines understand your content better!



Why use Semantic Elements?

- ✓ Improves readability
- ✓ Boosts SEO
- ✓ Helps accessibility
- ✓ Makes code organized

Smart code makes a better web!



Module 9

Lesson 2

Content Sections

These tags help organize the main parts of a webpage.

1

<section>

The **<section>** tag is used to group related content together. It helps divide the webpage into sections or chapters.

<section>... </section>



2

<article>

The **<article>** tag is used for independent content that can stand on its own, like a blog post, news story, or article.

<article>... </article>

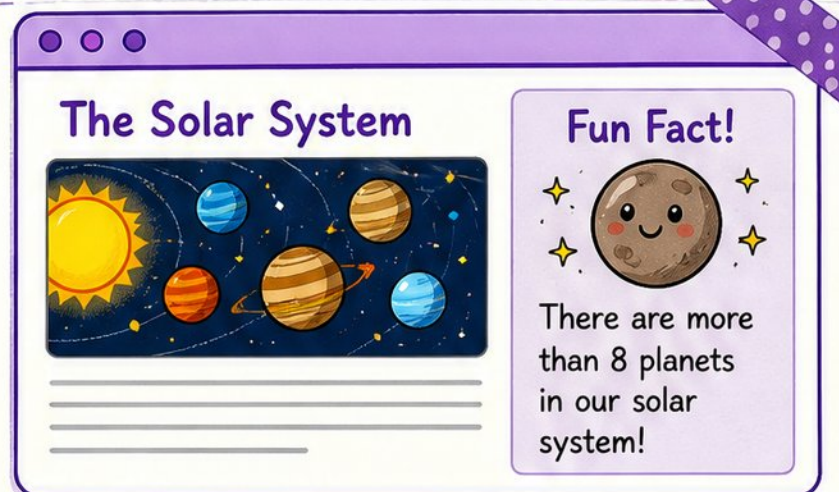


3

<aside>

The **<aside>** tag is used for content that is related to the main content, but not essential. Like side notes or extra info.

<aside>... </aside>



www.codescompiler.com

Module:

Module 9

Lesson:

Lesson 3

Accessibility Basics

Making digital things easy for everyone to use!



1 Alt Text

Alt text describes images so people who can't see them can understand what's there.



Example Alt Text:

"A happy golden puppy playing with a red ball on the grass."

2 Labels

Labels tell us what something is or what it does.

Example:

Your Name:

Email:

This label tells you to enter your name.



3 Readable Content

Text should be easy to read with clear fonts, good contrast, and the right size.

Example:

This text is hard to read.

✗ Hard to Read

This text is easy to read!

✓ Easy to Read

Remember: Small changes can make a big difference for everyone!

www.codescompiler.com

Module 9

Lesson 4

Writing Clean HTML



Clean HTML is easy to read, understand, and helps everyone work better!

1

Proper Indentation

Indent your code to show the structure. It makes it easy to read and understand.

❌ Hard to Read

```
<html>
<head><title>My Page</title></head>
<body>
  <h1>Hello!</h1>
  <p>Welcome to my website.</p>
</body>
</html>
```

✅ Easy to Read

```
<html>
  <head>
    <title>My Page</title>
  </head>
  <body>
    <h1>Hello!</h1>
    <p>Welcome to my website.</p>
  </body>
</html>
```



2

Meaningful Names

Use names that describe the content. It helps everyone know what your code is about.

❌ Not Meaningful

```
<div id="d1">
  <h2>Our Services</h2>
  <p>We build amazing websites.</p>
</div>
<div class="c1">
  
</div>
```

✅ Meaningful

```
<section id="services">
  <h2>Our Services</h2>
  <p>We build amazing websites.</p>
</section>
<figure class="team-photo">
  
</figure>
```



3

Code Organization

Keep your code organized with sections, comments, and consistent structure. It keeps your project neat!

```
<!DOCTYPE html>
<html lang="en">
  <!-- Head Section -->
  <head>
    <meta charset="UTF-8">
    <title>My Website</title>
  </head>
  <!-- Body Section -->
  <body>
    <!-- Header -->
    <header>
      <h1>Welcome!</h1>
    </header>
    <!-- Main Content -->
    <main>
      <p>This is my website.</p>
    </main>
    <!-- Footer -->
    <footer>
      <p>© 2024 My Website</p>
    </footer>
  </body>
</html>
```

Use Sections

Divide your page into logical sections.

Add Comments

Comments explain your code.

Be Consistent

Use the same style for neat and clean code.



Mini Project



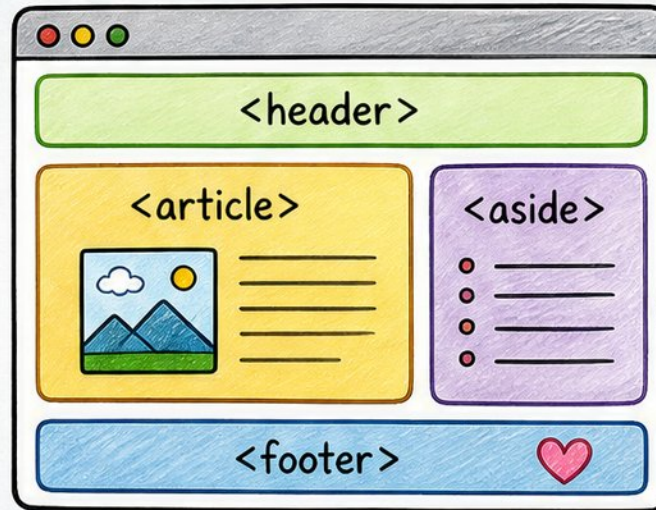
Module: Module 9



Lesson: Lesson 5

1 Semantic Blog Page

Use HTML semantic elements like `<header>`, `<main>`, `<article>`, `<aside>` and `<footer>` to build a meaningful blog page.



Meaningful Structure!



2 Accessible Layout

Make your blog accessible for everyone by using proper headings, alt text for images, good contrast and readable fonts.



- Proper Headings
- Alt Text for Images
- Good Contrast
- Readable Fonts



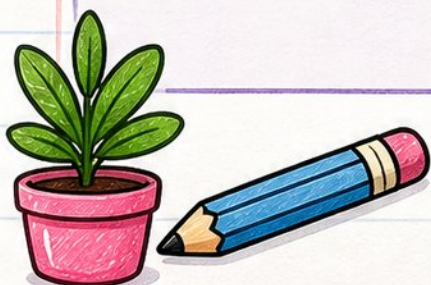
Design for Everyone!

3 Organized Code

Write clean, well-indented and commented code. It's easy to read, understand and to maintain.

```
index.html
1 <!DOCTYPE html>
2 <html lang="en">
3   <head>
4     <meta charset="UTF-8">
5     <title>My Blog</title>
6   </head>
7   <body>
8     <!-- Blog Header -->
9     <header> </header>
10    <!-- Main Content -->
11  </body>
12 </html>
```

Clean Code
Happy Mind!



www.codescompiler.com



Planning a Website

Module:

Module 10

Lesson:

Lesson 1



1

Website Goals

Before building a website, we need to know **WHY** we're making it and **WHAT** we want to achieve.



Ask Yourself:

- ✓ Who is my website for?
- ✓ What is the main purpose?
- ✓ What do I want visitors to do?



Clear goals = Better website!

2

Content Planning

Plan the content your website will have. Good content keeps visitors interested!

Plan Your Content

- Home
- About Us
- Services / Features
- Blog / News
- Gallery
- Contact Us

Tips:

- Know your audience.
- Keep it simple and clear.
- Use helpful and interesting content.

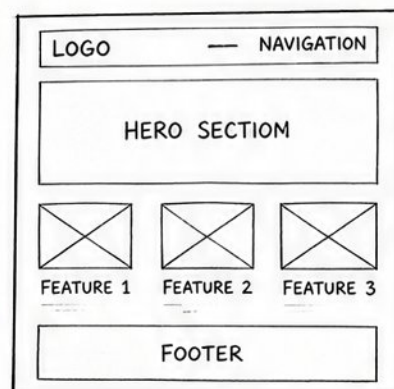


3

Wireframe Basics

A wireframe is a simple sketch of your website layout. It shows where things will go.

It's like a blueprint for your website!



Header
(Logo + Menu)

Main Section
(Big message or image)

Content Blocks
(Information sections)

Footer
(Contact, links, copyright)

Plan it first, build it next!





Personal Portfolio Website



Module: **Module 10**

Lesson: **Lesson 2**

1

About Me Section

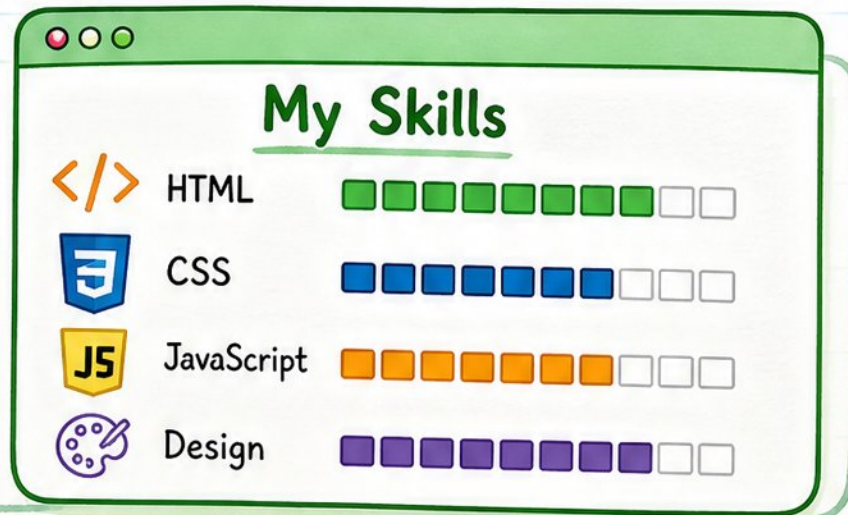
Introduce yourself! Share your name, a photo, your interests, and a few things about yourself.



2

Skills Section

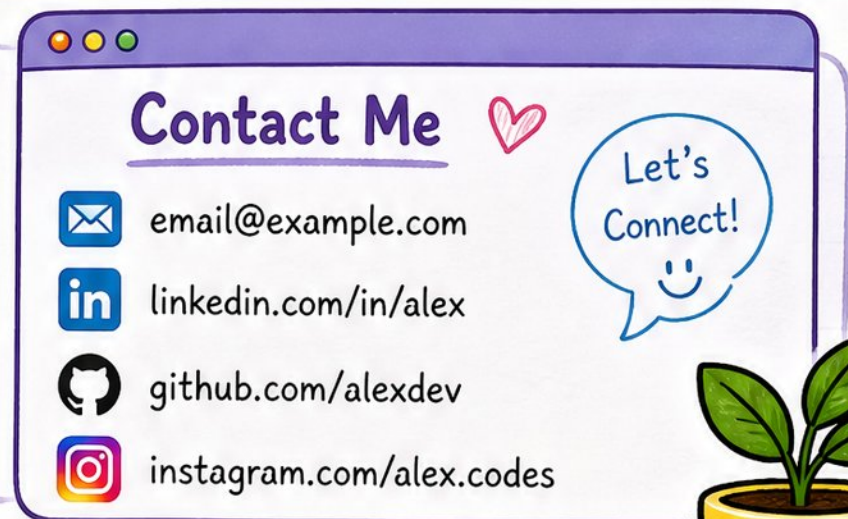
Show the skills you have learned. You can use icons or progress bars to make it look awesome!



3

Contact Information

Make it easy for people to contact you. Add your email and links to your social profiles.



www.codescompiler.com



School Website Project



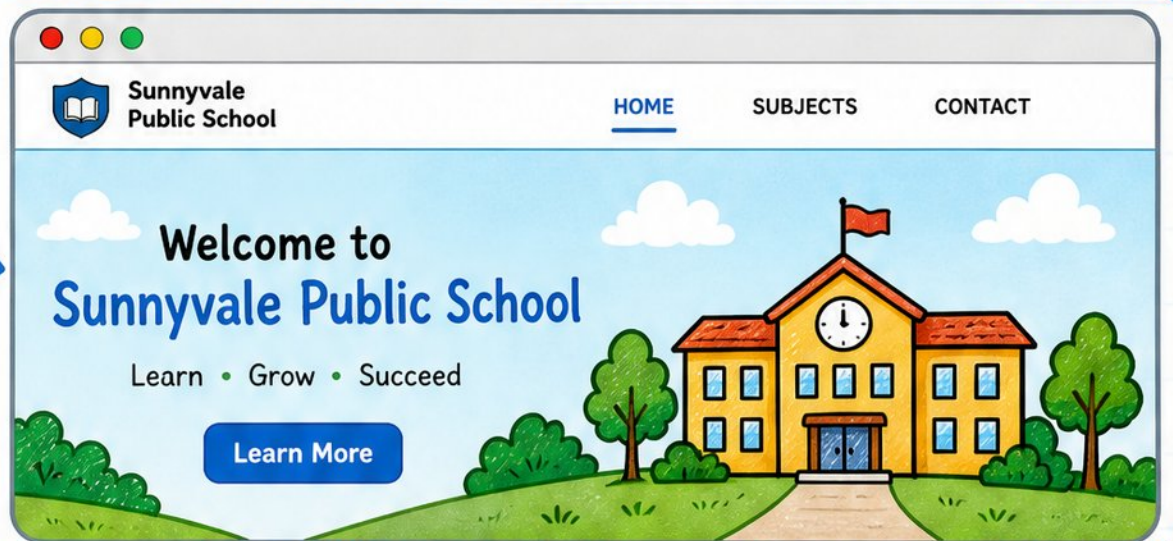
Module: Module 10

Lesson: Lesson 3



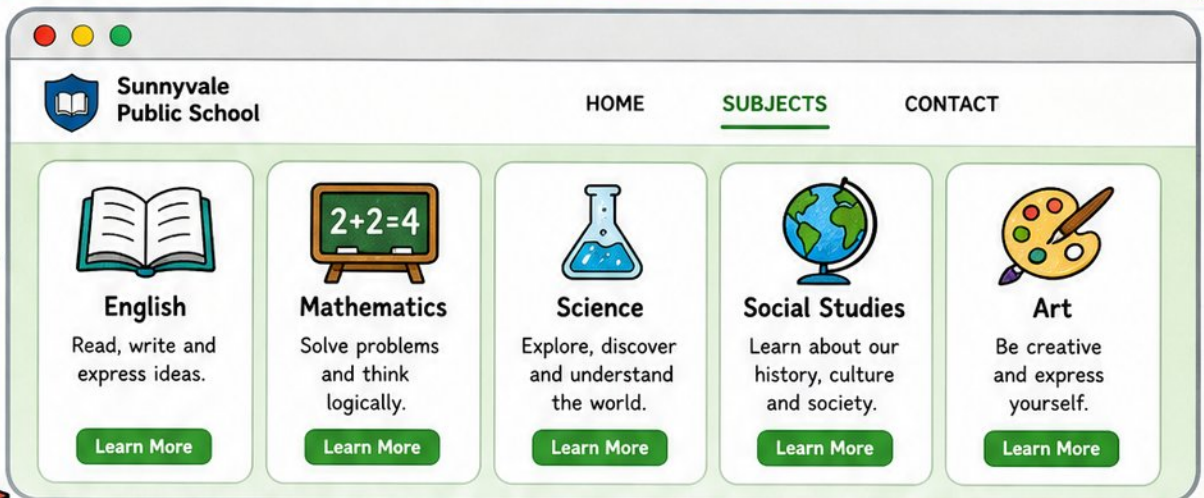
1 Home Page

The Home Page is the first page everyone sees. It gives a warm welcome and introduces our school.



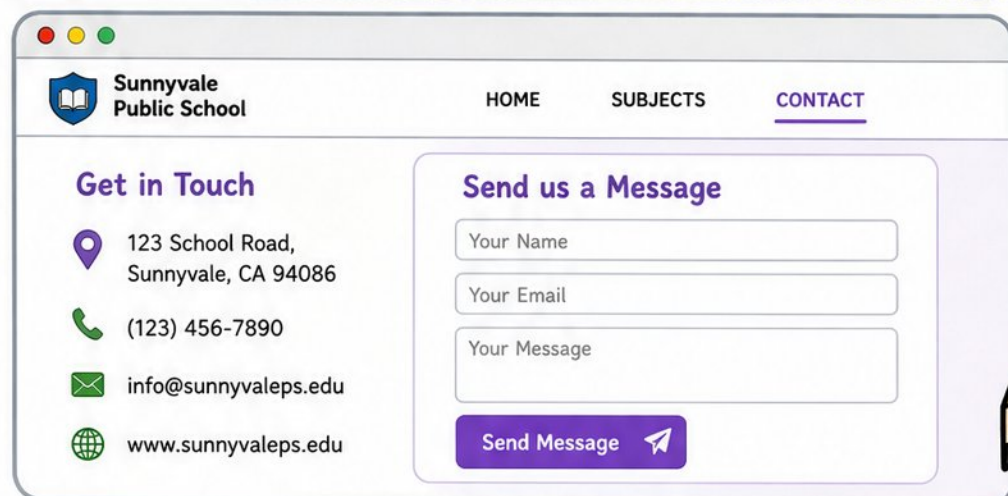
2 Subjects Page

This page shows all the subjects we study. It helps students explore and learn more about each one.



3 Contact Page

This page helps visitors get in touch with our school. We're always happy to hear from you!



www.codescompiler.com



Module:
Module 10

Lesson:
Lesson 4

HOBBY WEBSITE PROJECT

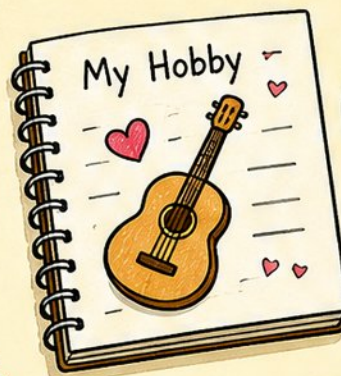


1

Favorite Hobby



Choose a hobby
you love and write
about it!



2

Images and Links

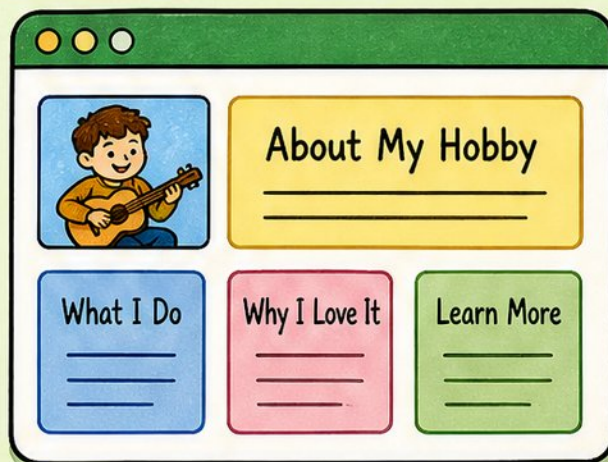
Add pictures and links
to websites that share
more about your hobby.



3

Organized Content

Keep your website
clean, neat, and easy
to understand.



- ☒ Clear Title
- ☒ Short Paragraphs
- ☒ Headings
- ☒ Good Layout

www.codescompiler.com

Final Capstone Project

Module:
Module 10

Lesson:
Lesson 5

1

Complete Multi-Page Website

Build a complete website with multiple pages like Home, About, Services, and Contact.



2

Navigation and Forms

Add easy navigation so users can move between pages. Create forms like a contact form to collect information.



Home

About

Services

Contact

Contact Us

Name

Email

Message

Send Message

3

Project Review and Publishing

Review your website for errors and make improvements. Then publish your project and share it with the world!



Your Website
is Live!

www.codescompiler.com