

## Task 4: Mastering HTML iFrames

### Objective:

Learn to effectively utilize HTML iFrames to embed external content, such as a map, a video, or another webpage, into your own webpage. This task focuses on understanding the usage, attributes, and limitations of iFrames in HTML, which are crucial for integrating third-party content seamlessly.

### Pre-requisites:

- Basic understanding of HTML elements and attributes
- Familiarity with a code editor like Visual Studio Code

### Concepts Covered:

- Basic iFrame Embedding
- Embedding a Map
- Embedding a Video

### Concepts:

#### 1. Basic iFrame Embedding:

The `<iframe>` tag is used to embed another document within the current HTML document. It allows you to display an external webpage within your own webpage.

```
<iframe src="https://example.com" width="600" height="400" frameborder="0"></iframe>
```

#### 2. Embedding a Map:

You can use an iFrame to embed a Google Map or another mapping service. Adjust the size and border to fit the layout of your page.

```
<iframe
  src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!
1d3151.8354345092976!2d144.9537353153169!3d-37.81627917975195!2m3!1f0!2f0!3f0!
3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x6ad642af0f11fd81%3A0xf0727f9d7e3b2b12!
2sFederation%20Square!5e0!3m2!1sen!2sau!4v1616990368374!5m2!1sen!2sau"
  width="600" height="450" style="border:0;" allowfullscreen=""
  loading="lazy"></iframe>
```

#### 3. Embedding a Video:

Use an iFrame to embed a YouTube video or a video from another platform. Ensure the video is responsive and fits different screen sizes.

```
<iframe
  width="560" height="315"
  src="https://www.youtube.com/embed/dQw4w9WgXcQ"
  frameborder="0"
  allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope;
picture-in-picture"
  allowfullscreen>
</iframe>
```

### Setup:

#### 1. Install Visual Studio Code (VS Code):

Download and install VS Code from [Visual Studio Code](https://code.visualstudio.com/).

#### 2. Web Browsers:

Use Google Chrome or Mozilla Firefox for viewing your webpage and utilizing their developer tools for debugging.

### Tasks:

#### 1. Basic iFrame Embedding (15 minutes):

- Embed a simple webpage within an iFrame on your HTML page.
- Understand the basic attributes of an iFrame like `src`, `width`, `height`, and `frameborder`.

#### 2. Embedding a Map (15 minutes):

- Use an iFrame to embed a Google Map or another mapping service into your webpage.
- Adjust the size and border to fit the layout of your page.



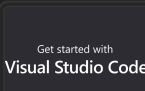

#### 3. Embedding a Video (15 minutes):

- Embed a YouTube video or a video from another platform using an iFrame.
- Explore how to ensure the video is responsive and fits different screen sizes.

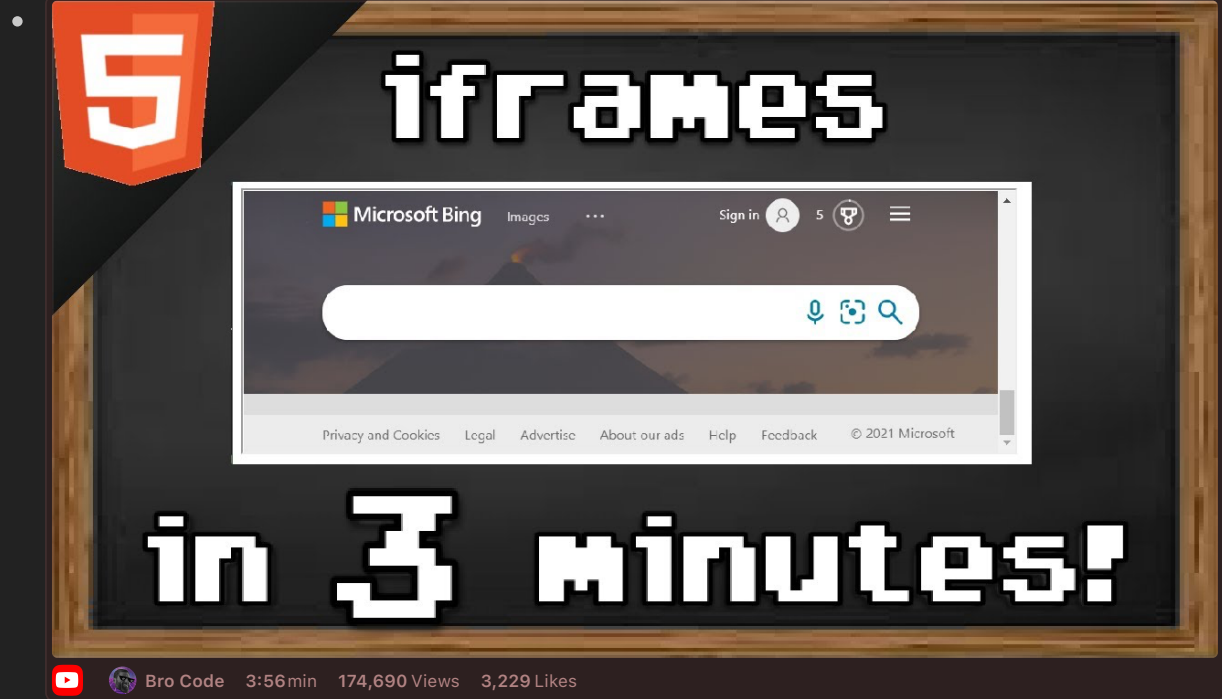
### Instructions:

1. Write the required code in `index.html`.
2. Open the file in your web browser to ensure the code displays correctly.
3. Use the browser's developer tools to debug and inspect the elements.

### Resources:

-  **<iframe>: The Inline Frame element - HTML: HyperText Markup Language**  
The <iframe> HTML element represents a nested browsing context, embedding another H...  
 <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/iframe>
-  **Documentation for Visual Studio Code**  
Find out how to set-up and get the most from Visual Studio Code. Optimized for building and d...  
 <https://code.visualstudio.com/docs>

## Videos:



## GitHub Instructions:

### 1. Open in Visual Studio Code:

After clicking on the "Open in Visual Studio Code" button from the GitHub Classroom confirmation page, VSCode will open the repository directly. If prompted, select "Open" or "Allow" to open the repository in VSCode.

### 2. Open the Terminal in VSCode:

In VSCode, open a terminal by selecting Terminal > New Terminal from the top menu.

### 3. Complete the Task:

In VSCode, write your solution in the `index.html` file.

### 4. Run and Test Your Code:

Open your `index.html` file in a web browser to ensure it works correctly. Use the following command:

```
open index.html
```

### 5. Commit Your Changes:

In the VSCode terminal, add your changes to git:

```
git add index.html
```

Commit your changes with a meaningful message:

```
git commit -m "Completed task 4"
```

## 6. Push Your Changes to Your Repository:

Push your changes to your forked repository:

```
git push origin main
```

## 7. Create a Pull Request:

Go to your repository on GitHub.

Click on the "Pull Requests" tab.

Click the "New Pull Request" button.

Ensure the base repository is the original template repository and the base branch is `main`.

Ensure the head repository is your forked repository and the compare branch is `main`.

Click "Create Pull Request".

Add a title and description for your pull request and submit it.

## Summary of Commands:

```
# Open in Visual Studio Code

# Open terminal in VSCode

# Complete the task by editing index.html

# Navigate to the directory containing index.html
cd path/to/your/index.html

# Run your code
open index.html

# Add, commit, and push your changes
git add index.html
git commit -m "Completed task 4"
git push origin main

# Create a pull request on GitHub
```