







Apps Script **HtmlService**

 Unleashing Creativity with Google Apps Script: Mastering HtmlService! 

Introduction to HtmlService	1
Question 1: Creating a Simple HTML Page	2
Question 2: Serving an HTML File	3
Question 3: Embedding JavaScript in HTML Service	4
Question 4: Using Templated HTML	5
Question 5: Serving CSS with HTML	5
Question 6: Communicating with Google Apps Script Functions	6
Question 7: Passing Data from Client to Server	7
Question 8: Incorporating Google Material Design	8
Question 9: Creating a User Interface for Spreadsheet	9
Question 10: Serving HTML with IFrame Sandbox Mode	10

Introduction to HtmlService

Dive into the fascinating world of Google Apps Script today, particularly the versatile HtmlService class. For those looking to add a creative and interactive touch to their Google Sheets, Docs, or web apps, these insights are a treasure trove!  

- Simple HTML Creation: Learn to create and serve simple HTML pages. #GoogleAppsScript #HtmlService #WebDevelopment
- Serving HTML Files: Master the art of serving HTML files from your script project. #HTML #Coding #Scripting

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

- JavaScript Integration: Embed JavaScript in your HTML pages for dynamic content. #JavaScript #WebDesign #DigitalCreation
- Dynamic Templated HTML: Use templated HTML for personalized content delivery. #TemplatedHTML #CustomWeb #TechTips
- Styling with CSS: Bring life to your pages with CSS styling. #CSS #WebStyling #CreativeCoding
- Server-Side Function Calls: Connect HTML elements to Google Script functions. #FunctionCalls #InteractiveDesign #Scripting
- Client-Server Data Transfer: Learn to pass data from client to server seamlessly. #DataTransfer #WebApps #GoogleScripts
- Material Design Implementation: Incorporate Google's Material Design for a sleek look. #MaterialDesign #UIUX #DesignInTech
- Custom UI for Google Sheets: Create unique user interfaces within Google Sheets. #CustomUI #GoogleSheets #SpreadsheetMagic
- IFrame Sandbox Mode: Serve secure HTML content in IFrame sandbox mode. #IFrame #WebSecurity #SandboxMode

Each tip includes a detailed explanation and a practical code example, making them accessible for a variety of applications. Embrace these strategies to transform your Google Sheets, Docs, or standalone web apps into interactive and visually appealing masterpieces. 🌟📊

Question 1: Creating a Simple HTML Page

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

Question: How do you create and serve a simple HTML page using Google Apps Script?

Answer:

```
function doGet() {  
  
    return HtmlService.createHtmlOutput('<h1>Hello World!</h1>');  
  
}
```

Explanation: This script uses `HtmlService.createHtmlOutput()` to create a basic HTML page with a heading. This is typically used in the `doGet()` function to serve HTML content when a web app URL is accessed.

Question 2: Serving an HTML File

Question: How can you serve an HTML file stored in the script project?

Answer:

```
function doGet() {  
  
    var html = HtmlService.createHtmlOutputFromFile('Index');  
  
    return html;  
  
}
```

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

Explanation: This function serves an HTML page from a file named 'Index.html' in the script project. `createHtmlOutputFromFile()` is used to load the HTML content.

Question 3: Embedding JavaScript in HTML Service

Question: How do you embed JavaScript in an HTML page served by Google Apps Script?

Answer:

```
function doGet() {  
  
    var html = HtmlService.createHtmlOutput('<script>alert("Hello,  
world!");</script><p>Hello, world!</p>');  
  
    return html;  
  
}
```

Explanation: This script creates an HTML output that includes JavaScript. Note that due to Content Security Policy (CSP), certain JavaScript practices (like inline JavaScript) may not work, and alternatives (like using `google.script.run`) are recommended.

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

Question 4: Using Templated HTML

Question: How can you use templated HTML to dynamically insert values into a page?

Answer:

```
function doGet() {  
  
    var template = HtmlService.createTemplate('<p>Hello, <?=  
    ?>!  
</p>');  
  
    template.userName = 'Alice';  
  
    return template.evaluate();  
  
}
```

Explanation: This code uses the templating feature of HtmlService to insert a dynamic value (userName) into the HTML. <?= ?> is used to embed the variable.

Question 5: Serving CSS with HTML

Question: How do you include CSS in an HTML page served by Google Apps Script?

Answer:

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

```
function doGet() {  
  
    var html = HtmlService.createHtmlOutput('<style>body { color: blue;  
}</style><p>This is a blue text.</p>');  
  
    return html;  
  
}
```

Explanation: This script embeds CSS directly into the HTML output, changing the color of the text to blue.

Question 6: Communicating with Google Apps Script Functions

Question: How can you call a server-side Google Apps Script function from HTML served by HtmlService?

Answer:

```
function doGet() {  
  
    var html = HtmlService.createHtmlOutput('<button  
onclick="google.script.run.myFunction()">Click me</button>');  
  
    return html;  
  
}
```

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

```
function myFunction() {  
  
  Logger.log('Button was clicked!');  
  
}
```

Explanation: This script creates an HTML output with a button. When clicked, it calls the server-side function myFunction() using google.script.run.

Question 7: Passing Data from Client to Server

Question: How do you pass data from the HTML page to a server-side function?

Answer:

```
function doGet() {  
  
  var html = HtmlService.createHtmlOutput('<input id="name"  
type="text"><button  
onclick="submitName()">Submit</button><script>function submitName() {  
var name = document.getElementById("name").value;
```

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

```
google.script.run.withSuccessHandler(function(response) {
console.log(response); }).processName(name); }</script>');

return html;

}
```

```
function processName(name) {

Logger.log('Name submitted: ' + name);

return 'Received ' + name;

}
```

Explanation: This code includes a text input and a button in the HTML. When the button is clicked, it calls a JavaScript function submitName(), which retrieves the input value and passes it to the server-side function processName.

Question 8: Incorporating Google Material Design

Question: How can you use Google's Material Design components in an HTML page?

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

Answer:

```
function doGet() {  
  
    var html = HtmlService.createHtmlOutput('<link rel="stylesheet"  
href="https://fonts.googleapis.com/icon?family=Material+Icons"><button  
class="material-icons">face</button>');  
  
    return html;  
  
}
```

Explanation: This script includes Google's Material Design stylesheet and uses a Material icon in a button.

Question 9: Creating a User Interface for Spreadsheet

Question: How do you create a custom user interface in a Google Sheet using HtmlService?

Answer:

```
function showCustomUi() {  
  
    var html = HtmlService.createHtmlOutput('<h2>Custom UI</h2>');  
  
    SpreadsheetApp.getUi().showModalDialog(html, 'My Custom UI');
```

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>

```
}
```

Explanation: This function uses `HtmlService` to create a simple HTML output and then uses `SpreadsheetApp.getUi().showModalDialog()` to display it as a modal dialog in Google Sheets.

Question 10: Serving HTML with IFrame Sandbox Mode

Question: How can you serve HTML content in IFrame sandbox mode?

Answer:

```
function doGet() {  
  
    var html = HtmlService.createHtmlOutput('<p>Sandboxed  
content</p>').setSandboxMode(HtmlService.SandboxMode.IFRAME);  
  
    return html;  
  
}
```

Explanation: This script serves HTML content using IFrame as the sandbox mode, which provides a secure and isolated environment for the content.

These questions and answers demonstrate various uses of the HtmlService class in Google Apps Script, illustrating how to create interactive and dynamic web pages and user interfaces.

Learn more about JavaScript with Examples and Source Code. Google Apps Script and Workspace Laurence Svekis Courses <https://basescripts.com/>