

This session covers how to build dynamic webapps and services in Apps Script, using skills that you already have. During the session we'll show how to create rich interactive apps using regular HTML and JavaScript, while maintaining deep Google integration via Apps Script. We'll also cover how to use scripts to serve text content like JSON and XML



About Corey Goldfeder

[View full profile](#)

Corey is a software engineer on the Google Apps Script team, based in New York. He has previously worked on Similar Shape search for 3DWarehouse, and as a robotics researcher before joining Google.



Use what you know

Building web apps and services with Google Apps Script

Corey Goldfeder
Software Engineer, Google

You already know so many technologies

If only you could use them!

In this session you will learn about things you already know

- Things like HTML, CSS, JavaScript...
- JSON, RSS, XML...

And a few things you don't know yet

- Like how to use this stuff in Apps Script
- Or why there is a picture of a napping dog

I like dogs.





A brief history of Google Apps Script

or, the ghosts of I/O past

Where we came from

What can scripts do?

Google Apps Scripts can extend Google Apps

- Embed scripts in Spreadsheets and Sites
- Manipulate Calendar, Gmail and more
- Trigger scripts in response to events

But it's also a great platform for building standalone apps and services

- A script can be deployed at its own URL
- And be published directly to the Chrome Web Store
- Or just authorized for you, or for your domain



Building web apps, 2010

At Google I/O 2010 we launched UiApp

- Cross-browser widgets
- Easy to write and easy to understand
- Don't need to maintain separate client and server code

```
function doGet() {  
  var app = UiApp.createApplication();  
  app.add(app.createLabel("Hello World!"));  
  return app;  
}
```

GOOGLE APPS SCRIPT

Building web apps, 2011

At Google I/O 2011 we launched the GUI Builder

- WYSIWYG for UiApp.
- Build interfaces without writing code
- Compatible with UiApp



“Sometimes I just want to write HTML.”

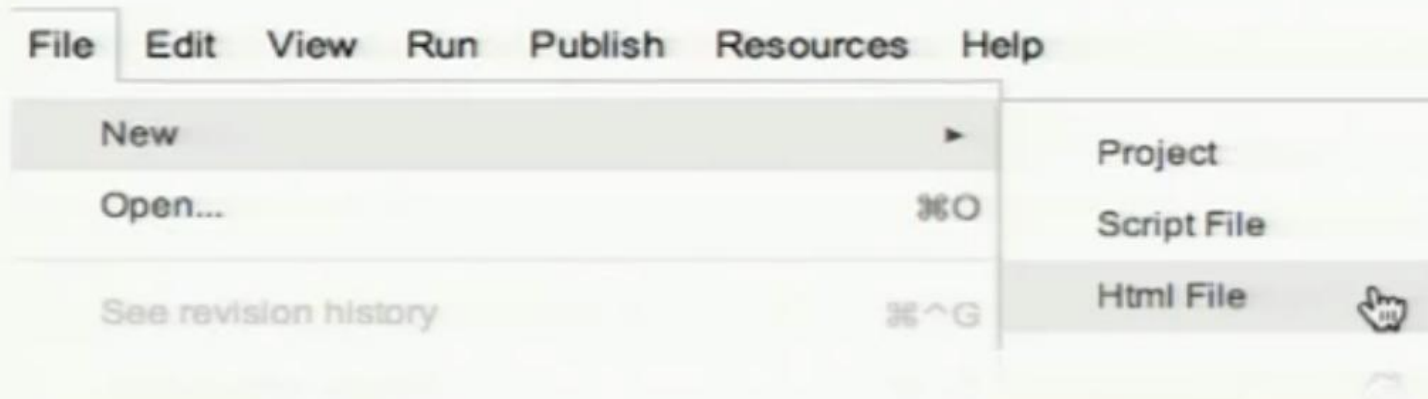
- You guys

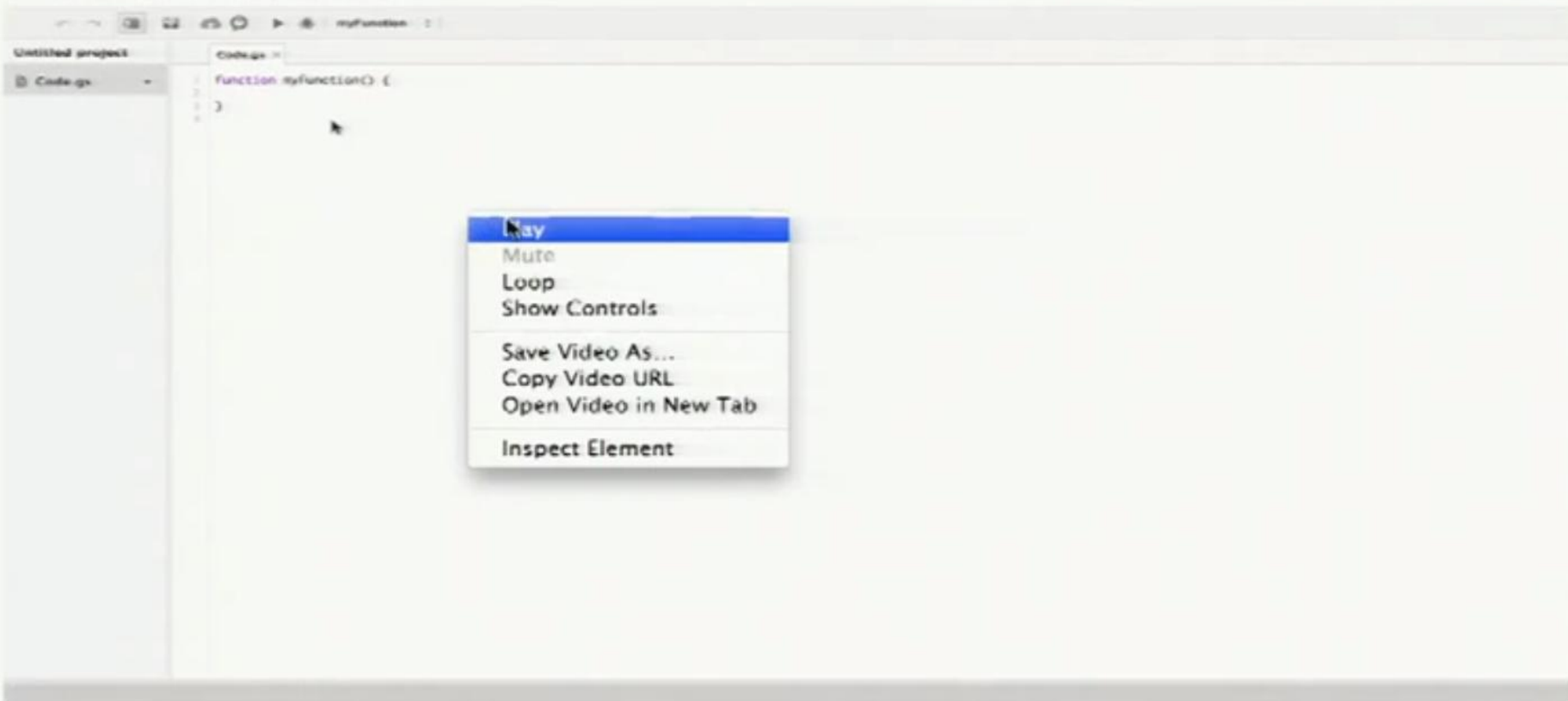


Building web apps, 2012

Introducing HtmlService

- Write HTML and JavaScript
 - Using jQuery and jQuery UI
 - With a built in API to run functions in Google Apps Script





Rename Project

Enter new project name

OK

Cancel



Untitled project

File Edit View Run Publish Resources Help

Saving

Untitled project

Code.gs

Code.gs x

```
1 function myFunction() {  
2  
3 }  
4
```

My Html App

File Edit View Run Publish Resources Help

New

Open...

⌘O

See revision history

⌘^G

Rename...

Make a copy...



Save

⌘S

Save all

⌘⇧S

Manage Versions... EXPERIMENTAL!

Project properties...

Build a user interface... EXPERIMENTAL!

Project

Script File

Html File

From script template...



Document



Presentation



Spreadsheet



Form



Drawing

From template...



My Html App

File Edit View Run Publish Resources Help



Select function ↕

My Html App

Code.gs

mypage.html

Code.gs

* mypage.html

```
1 <html>
2   <b>Hello World!</b>|
3 </html>
4
```

My Html App

File Edit View Run Publish Resources Help

undo redo save sync play debug myFunction

My Html App

Code.gs

mypage.html







* Code.gs x mypage.html x

```
1 function myFunction() {  
2   return |  
3 }  
4
```



ml App

File View Run Publish Resources Help

 myFunction

* Code.gs x

mypage.html x

nl

1 function myFunction() {
2 return HtmlService.
3 }
4

createHtmlOutput() : HtmlOutput

createHtmlOutput(BlobSource blob) : HtmlOutput

createHtmlOutput(String html) : HtmlOutput

createHtmlOutputFromFile(String filename) : HtmlOutput

createTemplate(BlobSource blob) : HtmlTemplate

createTemplate(String html) : HtmlTemplate

createTemplateFromFile(String filename) : HtmlTemplate

nl App

t View Run Publish Resources Help



* Code.gs x

mypage.html x

```
1 function doGet() {  
2   return HtmlService.createHtmlOutputFromFile("mypage");  
3 }  
4
```

p

Run Publish Resources Help

Publish to Gallery...

Deploy as web app...

Register in Chrome Web Store... EXPERIMENTAL!

```
1 function doGet() {  
2   return HtmlService.createHtmlOutputFromFile("my");  
3 }  
4
```

Deploy as web app

Please read this carefully. It's not the usual yada yada.

Project version:

1 ↕

Execute the app as:

me (appscriptio@gmail.com) ↕

You need to authorize the script before distributing the URL.

Who has access to the app:

- ✓ Only myself
- Anyone
- Anyone, even anonymous

Deploy

Cancel

[Help](#)

Deploy as web app

This project is now deployed as a web app.

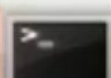
Current web app URL:

<https://script.google.com/macros/s/AKfycbwDVD6Pqmc5Xh>

Test web app for your [latest code](#).

OK

Hello World!



What can you do with it?

Let's build something fun with Gmail.

Windows (4) - appuruplex@gmail.com

https://mail.google.com/mail/#!/inbox

+You Search Images Maps Play YouTube News Gmail Drive Calendar More

Google

appuruplex@gmail.com

Gmail - 1-16 of 16

Compose

Inbox (14)

Starred

Important

Sent Mail

Drafts

More

Chat

Search people...

Apps Script

Get status here

Call phone

Corey Goldfeder

Saurabh Gupta

Valerie Blocher

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Andrew Collag	Batteries - Sometimes it's a good to take extra batteries for your phone.	12:42 pm
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Corey, me (2)	seriously, I need his address - What number on Drury Lane?	12:38 pm
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Corey Goldfeder	I did not know you could do that with JavaScript - Well, I guess I did know that you could do it with JavaScript. But now in Apple Script.	12:38 pm
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Arif Kulkarni	Google ID - Dear Prof. Rishit, Can you automate everything for me? Thanks, Arif Zapata.com	11:09 am
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Saurabh Gupta	FireDrill - all hands on deck - We need to submit a TPS report for the Chief Reciprocal Integration Officer. He needs to be briefed.	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Saurabh, me (2)	it's me or JavaScript - it's JavaScript right?	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Nikhil Singhal	TPS reports - Please submit your TPS reports for July by tomorrow, in triplicate. - Bob	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Drew Collag (sounds - (2)	funny - The coldest winter I ever spent was a summer in San Francisco.	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Corey Goldfeder	Do you know the muffin man? - I believe that he lives on Drury Lane, but I was hoping you had a more exact address.	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Kevin Winter	Slinky on a treadmill - http://www.youtube.com/watch?v=718bZ_gLuoQ	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Saurabh Gupta	my home town - this was where we use to go for school picnic every year :-)	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Je Javir Nigam	Webapps are hard, let's go scripting - All I want is a bit of html, a vint of javascript and databases for far as the eye can see. Can yr	Jun 26
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Valerie, me (2)	Very bad punst - Hey Corey, Which of these puns do you think is the worst?? Val - To write with a broken pencil is	Jun 25
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gmail Team	Customize Gmail with colors and themes - To spice up your inbox with colors and themes, check out the Themes tab under Settings.	Jun 25
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gmail Team	Import your contacts and old email - You can import your contacts and mail from Yahoo!, Hotmail, AOL, and many other web mail or i	Jun 25
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Gmail Team	Get Gmail on your mobile phone - Access Gmail on your mobile phone The days of reading your computer to get to your inbox are for	Jun 25

uLhtml x

```
ipt type="text/javascript" src="htt
ipt type="text/javascript" src="htt
k rel='stylesheet' href='http://aj
var tags = getGmailTags(); ?>
<ipt>
r tags = <?! JSON.stringify(tags)
r pretty = true;
document).ready(function() {
  window.openMe = function(x) {
    var contents = '<input style="c
    tags.words[x].ids.map(func
    $('div#pop-up').html(contents);
    $('div#pop-up').show();
    $('#accordion').accordion({acti
    google.script.run.withSuccess
  });
  $('#title').button();
  $('#close').button().click(func
  return false;
});
$('#styles').button().click(func
  pretty = !pretty;
  $('#root').attr('class', pretty
  $('#container').attr('class', p
  $('#styles').button('option',
});

action getTags() {
  setTimeout(function() {
    google.script.run.withSuccessHandl
    for (var m in newtags.threads) {
      tags.threads[m] = tags.threads[m] || newtags.threads[m];
```

Deploy as web app

Current web app URL: [Disable web app](#)

https://script.google.com/macros/s/AKfycby7c3x1H8eC

Test web app for your [latest code](#)

Please read this carefully. It's not the usual yada yada.

Project version:
1

Execute the app as:
User accessing the web app

Users will be asked to authorize the script.

Who has access to the app:
Anyone

Update

Cancel

Help

address bad **batteries** broken calendar computer
contacts discovered **down** drury earthquakes email fish free
fully given **gmail** goes hard **javascript** jun **know** lifts los mail
mobile need number out pencil phone professor puns schools **seen** shaky **show**
side smog sometimes stole **take** themes theory thief think tps twelve very write

Try it yourself!

Try Gmail Tags here: <http://goo.gl/xAaOR>

Send something to appsscriptio@gmail.com

Use What You Know!

Use the technologies you already know

- Use standard HTML, CSS and JavaScript
- Optionally use jQuery and jQuery UI
- Many other jQuery plugins just work

With all if the Apps Script goodness baked in

- Authentication with Google accounts
- Access to all Google Apps Script APIs
- Painless deployment
- Embedding in Google Apps

“But how do I call the server.”

- Perceptive audience member





Introducing the google.script API

How do I call the server?

Introducing google.script

- Make calls to the Apps Script server from your client code.
- No need to worry about AJAX and XHRs - we do it for you

```
function doSomething() {  
  Logger.log("I did something!");  
}
```

GOOGLE APPS SCRIPT

```
<script>  
  google.script.run.doSomething();  
</script>
```

HTML/JAVASCRIPT

Calling back to the server

You can pass parameters to your function (regular JavaScript objects)

```
function getSubject(threadId) {  
  // I can use that thread id here  
}
```

GOOGLE APPS SCRIPT

```
<script>  
  google.script.run.getSubject("a gmail thread id");  
</script>
```

HTML/JAVASCRIPT

Calling back to the server

You can return a value from your function

```
function getSubject(threadId) {  
  return GmailApp.getThreadById(threadId).getFirstMessageSubject();  
}
```

GOOGLE APPS SCRIPT

```
<script>  
  google.script.run.withSuccessHandler(myHandler).getSubject("a gmail thread id");  
</script>
```

HTML/JAVASCRIPT

Calling back to the server

You can return a value from your function

```
function getSubject(threadId) {  
  return GmailApp.getThreadById(threadId).getFirstMessageSubject();  
}
```

GOOGLE APPS SCRIPT

```
<script>  
  google.script.run.withSuccessHandler(myHandler).getSubject("a gmail thread id");  
  
  function myHandler(subject) {  
    alert("The subject was: " + subject);  
  }  
</script>
```

HTML/JAVASCRIPT

Calling back to the server

You can handle errors from the server

```
function getSubject(threadId) {  
  x.y(); //Oops! This will throw a ReferenceError  
  ...  
}
```

GOOGLE APPS SCRIPT

```
<script>  
  google.script.run.withFailureHandler(onFailure).getSubject(...);  
  
  function onFailure(error) {  
    alert("Error! " + error.message);  
  }  
</script>
```

HTML/JAVASCRIPT

“I want to dynamically create HTML!”

- Mr. Power User, Esq.





Introducing Google Apps Script Templates

Generating dynamic pages

I want an HTML page with a table taken from a spreadsheet

- I could generate the HTML in code
- But that's complex and error prone
- Introducing **Google Apps Script Templates**

Generating dynamic pages

Write your HTML as before

```
<html>  
  My Spreadsheet as a table  
  <table>  
    <!-- Table content should go here... -->  
  </table>  
</html>
```

HTML/JAVASCRIPT

Generating dynamic pages

But use special 'scriptlet' tags to add in JavaScript

HTML/JAVASCRIPT

```
<html>
  My Spreadsheet as a table
  <table>
    <? for (var i = 0; i < data.length; ++i) { ?>
      <tr>
        <? for (var j = 0; j < data[i].length; ++j) { ?>
          <td> <?= data[i][j] ?> </td>
        <? } ?>
      </tr>
    <? } ?>
  </table>
</html>
```


Generating dynamic pages

Just three tags to remember

- `<? ... ?>` a 'scriptlet' adds code to a template
- `<?= ... ?>` a 'printing scriptlet' adds code and prints a value
- `<?!= ... ?>` a 'force printing scriptlet' prints a value without escaping

Code.gs x

```
1 function myFunction() {  
2   var t = HtmlService.createTemplateFromFile("mypage");  
3   t.data = SpreadsheetApp.getActiveSpreadsheet().getDataRange().getValues();  
4   SpreadsheetApp.getActiveSpreadsheet().show(t.evaluate().setWidth(800).setHei  
5 }  
6  
7
```

```
1 <html>
2   <style>.header{background:lightblue}
3     td{border-top:solid black 1px} .corner{border:0}</style>
4   <div style="text-align:center">
5     <table cellpadding=0; cellspacing=10>
6       <? for (var i = 0; i < data.length; ++i) { ?>
7         <tr>
8           <? for (var j = 0; j < data[i].length; ++j) { ?>
9             <td class="<?= i == 0 ^ j == 0 ? 'header'
10              : i == 0 & j == 0 ? 'corner' : '' ?>">
11               <?= data[i][j] ?> </td>
12             <? } ?>
13           </tr>
14         <? } ?>|
15       </table>
16     </div>
17 </html>
```


Spreadsheet To Table

File Edit View Run Publish Resources Help

Execution transcript...

Log...

32x18px

Compact controls

~O P

Full screen

Spreadsheet To Table

Code.gs

mypage.html

```
1 createTemplateFromFile("mypage");  
2 App.getActiveSpreadsheet().getDataRange().getValues();  
3  
4 Logger.log(t.getCode());  
5 SpreadsheetApp.getActiveSpreadsheet().show(t.evaluate().setWidth(800).setHeight(170).setTitle("As a table!"));  
6  
7  
8
```

Spreadsheet To Table

File Edit View Run Publish Resources Help

myFunction 0.1

Spreadsheet To Table

15 Code.gs

⑤ mypage.html

Code.gs mypage.html

```
function myFunction() {
  var t = HtlService.createTemplateFromFile("myPage");
  t.data = SpreadsheetApp.getActiveSpreadsheet().getSheetRange().getValues();
  Logger.log(t.getCode());
  SpreadsheetApp.getActiveSpreadsheet().show(t.evaluate().setWidth(800).setHeight(170).setTitle("As a table!"));
}
```

Logging output

```

    out (var i = 0; i < data.length; ++i) {
        output_ += "<tr>\n";
        output_ += "    "; for (var j = 0; j < data[i].length; ++j) {
            output_ += "<td class='\" + \"'>"; output_$ = i == 0 ? "\n" : "";
                j == 0 && j == 0 ? "corner" : ""; output_ += "\" + \"\n";
            output_ += "    "; output_$ = data[i][j]; output_ += "    ";
            output_ += "    "; }
        output_ += "</tr>\n";
        output_ += "    "; }
        output_ += "</table>\n";
        output_ += "</div>\n";
        output_ += "</div>\n";
/* End of user code */
return output.$out.append("");
}

```

27000

```

7
8
9 (function() { var output = HtmlService.initTemplate(); output._ = '<html>\n';
10   output._ = '  <style>.header{background:lightblue}\n' +
11     '    td{border-top:solid black 1px} .corner{border:0}</style>\n' +
12     '  <div style=\"text-align:center\">\n' +
13     '    <table cellpadding=0; cellspacing=0>\n' +
14     '      '; for (var i = 0; i < data.length; ++i) { ;
15   output._ = '        <tr>\n';
16   output._ = '          '; for (var j = 0; j < data[i].length; ++j) { ;
17   output._ = '            <td class=\"\"; output._$ = i == 0 ^ j == 0 ? 'header'
18     : i == 0 && j == 0 ? 'corner' : ''; output._ = ' \">\n';
19   output._ = '          '; output._$ = data[i][j] ; output._ = ' </td>\n';
20   output._ = '        '; } ;
21   output._ = '      </tr>\n';
22   output._ = '    '; } ;
23   output._ = '  </table>\n';
24   output._ = ' </div>\n' +
25     '</html>';
26   /* End of user code */
27   return output.$out.append('');
28 })();

```


Spreadsheet To Table

File Edit View Run Publish Resources Help

Running function r

myFunction

Spreadsheet To Table

Code.gs

mypage.html

Code.gs

mypage.html

```
1 function myFunction() {  
2   var t = HtmlService.createTemplateFromFile("mypage");  
3   t.data = SpreadsheetApp.getActiveSpreadsheet().getDataRange().getValues();  
4   Logger.log(t.getCodeWithComments());  
5   SpreadsheetApp.getActiveSpreadsheet().show(t.evaluate().setWidth(800));  
6 }  
7
```

```

2 <style>.header{background:lightblue}
3   td{border-top:solid black 1px} .corn
4 <div style="text-align:center">
5 <table cellpadding=0; cellspacing=10>
6 | <? x.y(); for (var i = 0; i < data.len
   <tr>
7     <? for (var j = 0; j < data[i].len
8       <td class="<?= i == 0 ^ j == 0 ?
9         : i == 0 && j == 0 ? 'corner'
10        <?= data[i][j] ?> </td>
11      <? } ?>
12    </tr>
13  <? } ?>
14 </table>
15 </div>
16 </html>

```

ReferenceError: "x" is not defined. (line 6) [Dismiss](#)

```
nypage");  
().getDataRange().getValues();  
evaluate().setWidth(800).setHeight(170).setTitle("As a
```

Generating dynamic pages

So what was that about 'escaping' and 'force printing'?

- `<?= ... ?>` Will make sure you don't accidentally inject JavaScript or HTML
- `<?!= ... ?>` Will print exactly what you specified.

```
<? var stringWithScript = '<script>doVeryBadThing()</script>'; ?>
```

HTML/JAVASCRIPT

This one is safe: `<?= stringWithScript ?>`

This one is dangerous: `<?!= stringWithScript ?>`

This one is safe: `<script>doVeryBadThing()</script>`

OUTPUT

This one is dangerous:

HtmlService

To recap:

- Use **HtmlService** to serve HTML and JavaScript
- Call back to the server with **google.script**
- Generate dynamic content with **Google Apps Script Templates**

HtmlService limitations

HtmlService is built with Google Caja, for your protection.

- Emulates EcmaScript5 strict mode
- Scripts can't access cookies or other browser resources
- Scripts can't dynamically install new scripts

HtmlService limitations

Scripts can't dynamically install new scripts.

- Can't dynamically add a new `<script>` tag
- This works:

```
document.write("<input type='button' id='button1'>");  
document.getElementById("button1").onclick = function() {  
    alert(123);  
};
```

HTML/JAVASCRIPT

- This doesn't work:

```
document.write("<input type='button' onclick='alert(123)'>");
```

HTML/JAVASCRIPT

HtmlService awesomeness

HtmlService is built with Google Caja, for your convenience.

- Emulates EcmaScript5 strict mode - *everywhere!*
- Fills in missing 'modern' JavaScript functions on all browsers.
- You can use jQuery and jQuery UI, even embedded in a Spreadsheet

Google Caja is open source!

developers.google.com/caja/.



Is that really all?

I think he still has a few minutes...

Building web services in Google Apps Script

Introducing ContentService

- Return text from a script
- ...JSON, RSS, XML, etc. - perfect for writing services
- ...no Caja or UiApp sanitization applied

ContentService

This is basically everything:

```
function doGet() {  
  return ContentService.createTextOutput('Hello World!').setMimeType(ContentService.PLAIN_TEXT);  
}
```

GOOGLE APPS SCRIPT

Gmail Feed

File Edit View Run Publish Resources Help

doGet

Gmail Feed

Code.gs

rss.html

Code.gs

rss.html

```
1 function doGet() {  
2   return ContentService.createTextOutput(HtmlService.createTemplateFromFile("rss").evaluate().getContent())  
3     .setMimeType(ContentService.MimeType.RSS);  
4 }  
5
```

Code.gs x

rss.html x

```
1 <rss version="2.0">
2 <channel>
3   <title><?= "Gmail Feed" ?></title>
4   <? var threads = GmailApp.getInboxThreads();
5     var messages = GmailApp.getMessagesForThreads(threads);
6     for (var t in threads) { ?>
7     <item>
8       <title><?= threads[t].getFirstMessageSubject() ?></title>
9       <description><?= messages[t][0].getBody() ?></description>
10      <guid><?= threads[t].getId() ?></guid>
11      <pubDate><?= threads[t].getLastMessageDate().toUTCString() ?></pubDate>
12    </item>
13    <? } ?>
14  </channel>
15 </rss>
```



Search Reader

Reader

A look at what's new in your Reader

SUBSCRIBE

New in your subscriptions

Enter a search term to find feeds or paste a feed url.

<https://script.google.com/macros/s/AKfycb>

Add

e.g., googleblog.blogspot.com or [cnn](https://cnn.com)

★ Starred items

📊 Trends

Browse for stuff

► Explore

▼ Subscriptions

Google Blog (3)

all you need is web

nts in the northern hemisphere say goodbye to each other at the San Diego meeting with thousands of educators and a Society for Technology in Education (ISTE) conference. ...

Gmail Feed

You have subscribed to "Gmail Feed."

Need another Glass demo... ☆

Prof. Script,

Thanks for volunteering. We really need you to jump out of a zeppelin, ride a motorbike and abseil down the side of the building so your can write some awesome AppScript to automate the creation of a cloud tag that can be viewed by Google Glass.

Thanks
Jasvir

☆ Email ☐ Keep unread ☐ Add tags

Blimps ☆

Dear Professor,

Are my scripts available from my blimps?

Are my blimps controllable by my scripts?

Please let me know.

Which output should I use?

UiApp, ContentService, HtmlService oh my!

All of our output types complement each other

- Use UiApp to build simple cross-browser-compatible UI's
- Use HtmlService to build complex AJAX web apps
- Use ContentService to build services and RSS feeds

Bringing it all together...

Let's wrap up all of the things Google Apps Script has launched this week.

- Scripts that aren't tied to spreadsheets or sites, at script.google.com
- Scripts that run with the authority of the user, not of the script author
- The ScriptDb store for building scripts that need a database **Room 7, 5:15pm**
- Direct Chrome Web Store integration
- HtmlService and Templates for building dynamic web apps

Code.gs x

ui.html x

```
1 var stops = { "": 1, "a":1,"able":1,"about":1,"across":1,"after":1,"all":1,"
2
3 function doGet() {
4   return HtmlService.createTemplateFromFile("ui").evaluate();
5 }
6
7 function getThreadBody(id) {
8   var thread = GmailApp.getThreadById(id);
9   return thread.getMessage()[0].getBody() + '<br><a href=' + thread.getPer
0 };
1
2 function getGmailTags() {
3   var words = {}, threads = GmailApp.getInboxThreads(0, 50);
4   GmailApp.getMessageForThreads(threads).map(function(messages, i){
5     messages.map(function(message) { (message.getSubject() + " " + message.
6       function(word){ word = word.toLowerCase(); if (!/\W|\d/.test(word) &&
7         words[word] = words[word] || {pos:[]}; words[word].pos.push(i); } )
8   var mean = 0, sd = 0, numWords = Object.keys(words).length;
9   for (var m in words) { mean += words[m].pos.length; }
```

Code.gs x

ui.html x

```
1 var stops = { "": 1, "a":1, "able":1, "about":1, "across":1, "after":1, "all":1, "
2
3 function doGet() {
4   return HtmlService.createTemplateFromFile("ui").evaluate();
5 }
6
7 function getThreadBody(id) {
8   var thread = GmailApp.getThreadById(id);
9   return thread.getMessage()[0].getBody() + '<br><a href=' + thread.getPer
0 };
1
2 function getGmailTags() {
3   var words = {}, threads = GmailApp.getInboxThreads(0, 50);
4   GmailApp.getMessagesForThreads(threads).map(function(messages, i){
5     messages.map(function(message) { (message.getSubject() + " " + message.
6       function(word){ word = word.toLowerCase(); if (!/\W|\d/.test(word) &&
7         words[word] = words[word] || {pos:[]}; words[word].pos.push(i); } }
8   var mean = 0, sd = 0, numWords = Object.keys(words).length;
9   for (var m in words) { mean += words[m].pos.length; }
```

```
<html>
  <head>
    <script type="text/javascript" src="http://code.jquery.com/jc
    <script type="text/javascript" src="http://code.jquery.com/ui
    <link rel='stylesheet' href='http://ajax.googleapis.com/ajax/
    <? var tags = getGmailTags(); ?>
    <script>
      var tags = <?!= JSON.stringify(tags) ?>;
      var pretty = true;
      $(document).ready(function() {
        window.openMe = function(x) {
          var contents = '<input style="color:red; font-size:25px;" type="text" value="Search" />
            tags.words[x].ids.map(function(id) { return "<h3 i
          $('div#pop-up').html(contents);
          $('div#pop-up').show();
          $('#accordion').accordion({active: false, autoHeight:
            google.script.run.withSuccessHandler(ui.newContent.
            ...
        }
      }
    }
  }
</script>
</head>
</html>
```



```

function getTags() {
  setTimeout(function() {
    google.script.run.withSuccessHandler(function(newtags){
      for (var m in newtags.threads) {
        tags.threads[m] = tags.threads[m] || newtags.threads[m];
      }
      var words = newtags.words, cloud = '';
      var sorted = Object.keys(words).sort();
      for (var t in sorted) {
        cloud += "<a href='#' class='tag' + words[sorted[t]].weight";
      }
      $('#cloud').html(cloud);
      for (var w in tags.words) {
        $('#word_' + w).click(function(){ openMe(this.id.substring(7));
      }
      getTags();
    }).getGmailTags();
  }, 30000);
}

```

address apps automate available bad batteries believe

blimps computer contacts corey **down** drury email exact

fully **gmail** goes goldfeder google hard hope hoping **javascript** jun

know lives mail make minions mobile more muffin **need** number out phone
fessor puns seen show side sometimes **take** themes think tps use very write

address apps automate available bad batteries believe

blimps computer contacts corey **down** drury email exact fully

gmail goes goldfeder google hard hope hoping javascript jun **know** lives

il make minions mobile more muffin **need** number out phone professor puns seen show

sometimes **take** themes think tps use very write

“Go away or I will replace you with a very small Google Apps Script.”

- Anonymous



`<?= "Thank You!" ?>`

Questions?

Please join the Google Apps Script community on Stack Overflow.

