

Updating Script Editor Web Part Customizations



Danny Jessee

SENIOR SHAREPOINT DEVELOPER

@dannyjessee



Script Editor Web Parts

Embed HTML, CSS, and JavaScript directly on a page

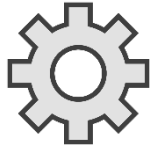
Scripts run as part of the page that contains them

Can interact with other data on the same page

Limitations compared to traditional web parts



Web Part Property Pane



Configure settings specific to a web part instance



Define custom web part properties



Client-side web parts can expose custom configurable properties



Rich controls for capturing property values



Updates to custom property values can be reflected immediately



_spPageContextInfo and pageContext

Title and URL of
current site

Avoids the need to
hard-code URLs

_spPageContextInfo
not available on
modern pages

pageContext object
is fully supported

pageContext is
available on classic
and modern pages



`_spPageContextInfo` to `pageContext`

Editors like Visual Studio Code can provide helpful IntelliSense

`_spPageContextInfo`

```
_spPageContextInfo.userId
```

```
_spPageContextInfo.webAbsoluteUrl
```

`pageContext`

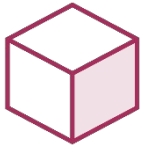
```
this.context.pageContext.user  
  .displayName
```

```
this.context.pageContext.web  
  .absoluteUrl
```

JavaScript Client Object Model



Client-side alternative to server-side SharePoint APIs



Familiar objects and syntax



Not guaranteed to work on modern pages



SharePoint PnPjs as an alternative structured API



“SharePoint Framework Development with PnP” course



SharePoint REST APIs



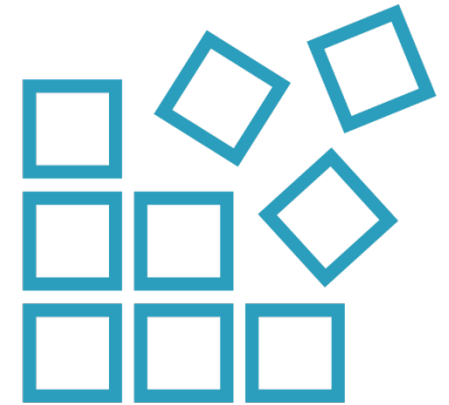
Preferred method
of interacting with
SharePoint



Well-supported
and
well-understood

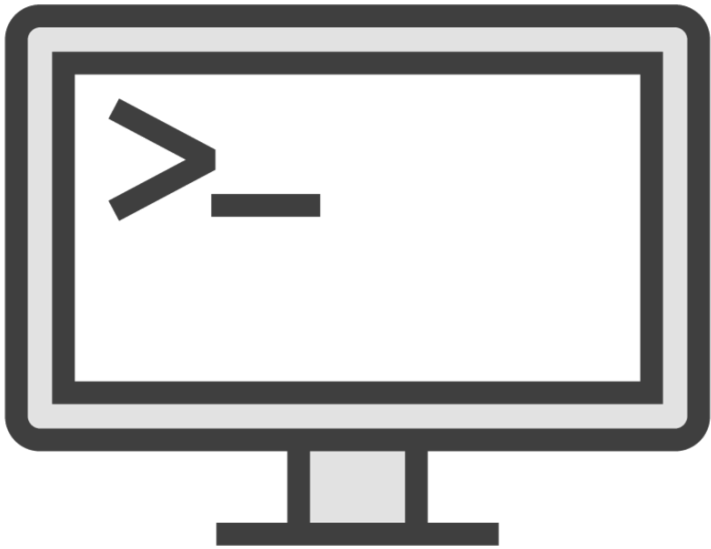


Microsoft continues
to invest in
REST APIs



REST API
requests may
now be batched

Making a REST API Call



[https://your-site-url.sharepoint.com/_api/web/lists/GetByTitle\('ListName'\)/items](https://your-site-url.sharepoint.com/_api/web/lists/GetByTitle('ListName')/items)

GET request for reading data

Response data may be Atom XML or JSON

POST request for writing data

Additional metadata for list item field values



SPHttpClient

`$('#__REQUEST
DIGEST').val()`

`this.context.
spHttpClient`

`get()` and `post()`
methods

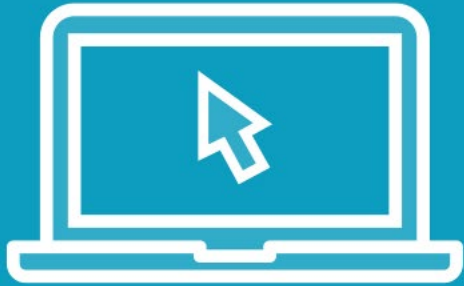
Automatically
adds a valid
request digest

`this.context.
httpClient`

Method calls
return Promises



Demo



Client-side web part

Custom property pane value

Using the pageContext object

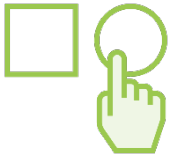
SPHttpClient for REST API calls



DOM Manipulation



Can programmatically make changes to the page structure



Target elements on a page by ID (`#suiteBar`, `#s4-bodyContainer`)



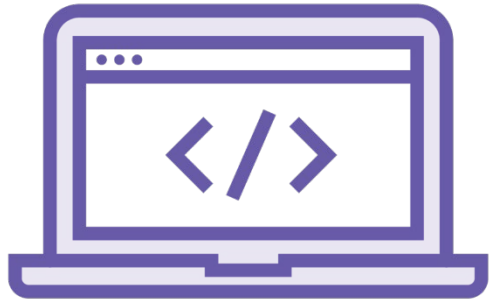
Customizations that manipulate the DOM are very fragile



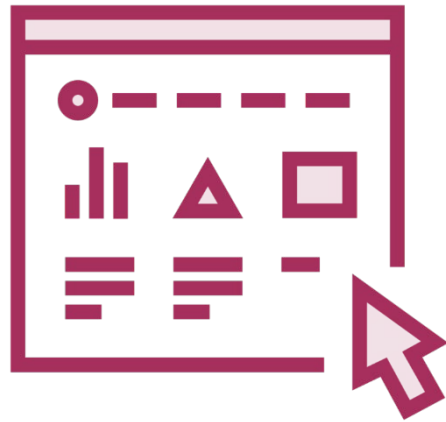
Application Customizer Extensions



Application Customizer Extensions



Customize the
modern SharePoint
user interface



Page
placeholders for
specific areas



Top and Bottom
placeholders



“Creating SharePoint
Framework
Extensions” course



Demo



Application Customizer extension

Provide default values for page placeholder contents



Review



Migrating legacy Script Editor
web part customizations

Client-side web parts

pageContext object

SPHttpClient for REST API calls

Shortcomings of DOM manipulation

Application Customizer extensions



Up Next:

Updating Other User Interface Customizations

