Global Office 365
Developer Bootcamp

Putting it Together (Part 1)

Bob German

Microsoft

Many thanks to our sponsors!

BA INSIGHT

K2®

Rencore

Powell 365

Microsoft

Quotes Web Part

Demonstration
Code walk-through

Similar Plunk → https://link.bobg.tv/ReactQuotes

SPFx Example → https://link.bobg.tv/SPFxQuotes

Calling REST Services from SPFx

- HttpClient
  - For fetching non-SharePoint resources
- GraphHttpClient
  - For talking to Office Graph
  - Manages extremely annoying things like getting bearer tokens
- SPHttpClient
  - For talking to SharePoint
  - Ensures proper headers, request digest, etc.
  - But… there is another way…
- PnP-JS-Core
  - Fluent API for calling SharePoint resources

Fetching Data with XMLHttpRequest

```
var xhttp = new XMLHttpRequest();
xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
       document.getElementById("demo").innerHTML =
           xhttp.responseText;
    }
};
xhttp.open("GET", "filename", true);
xhttp.send();
```

Fetching Data with jQuery

```
$.ajax({
  url: "filename",
  success: function (result) {
    document.getElementById("demo").innerHTML =
        result;
  }
  error: function (err) { alert (“Error!”); }
});
```

Fetching Data with jQuery and Promises

```
$.ajax({
  url: "filename"
})
.done(function (result) {
    document.getElementById("demo").innerHTML =
        result;
})
.fail(function (result) {
  alert (“Error!”);
});
```

Promises, Promises

Asynchronous

Synchronous

Promise

Beyond Promises

- **`async`** and **`await`** keywords make using Promises look like synchronous calls
- Still uses ES6 promises under the hood (or TypeScript will emulate for ES3)

Mini-Demo: from Promises to async/await

https://link.bobg.tv/Promises1

https://link.bobg.tv/Promises2

Fetch Demo

HttpClient
SpHttpClient
GraphHttpClient

fetch()

https://link.bobg.tv/SPFxFetch

PnP JS Core

```
import pnp from "sp-pnp-js";

// GET /_api/web
pnp.sp.web.get().then(r => {
    console.log(r);
});

// GET /_api/web/lists/getByTitle('Tasks')/items(1)
pnp.sp.web.lists.getByTitle("Tasks").items.getById(1).get().then(r => {

    console.log(r);
});
```

PnP JS Core - OData

```
pnp.sp.web.lists.getByTitle("Tasks").items.filter("Title eq
'Value'").top(2).orderBy("Modified").get().then(r => {

    console.log(r);
});

// Getting the second "page" of results from the top query
pnp.sp.web.lists.getByTitle("Tasks").items.filter("Title eq
'Value'").skip(2).top(2).orderBy("Modified").get().then(r => {
    console.log(r);
});
```

PnP JS Core - Caching

```
pnp.setup({
    defaultCachingStore: "session", // or "local"
    defaultCachingTimeoutSeconds: 30,
    globalCacheDisable: false // or true to disable caching
                              // for debugging/testing
});

pnp.sp.web.lists.getByTitle("Tasks").items.top(5)
   .orderBy("Modified").usingCaching().get().then(r => {
        console.log(r)
});
```

Always last before get()

PnP JS Core - Batching

```
let batch = pnp.sp.createBatch();

pnp.sp.web.lists.inBatch(batch).get().then(r => {
    console.log(r)
});

pnp.sp.web.lists.getByTitle("Tasks").items.inBatch(batch).get().then
(r => {
    console.log(r)
});

batch.execute().then(() => console.log("All done!"));
```

PnP JS Core

Demonstration
Code walk-through

https://link.bobg.tv/SPFxQuotes