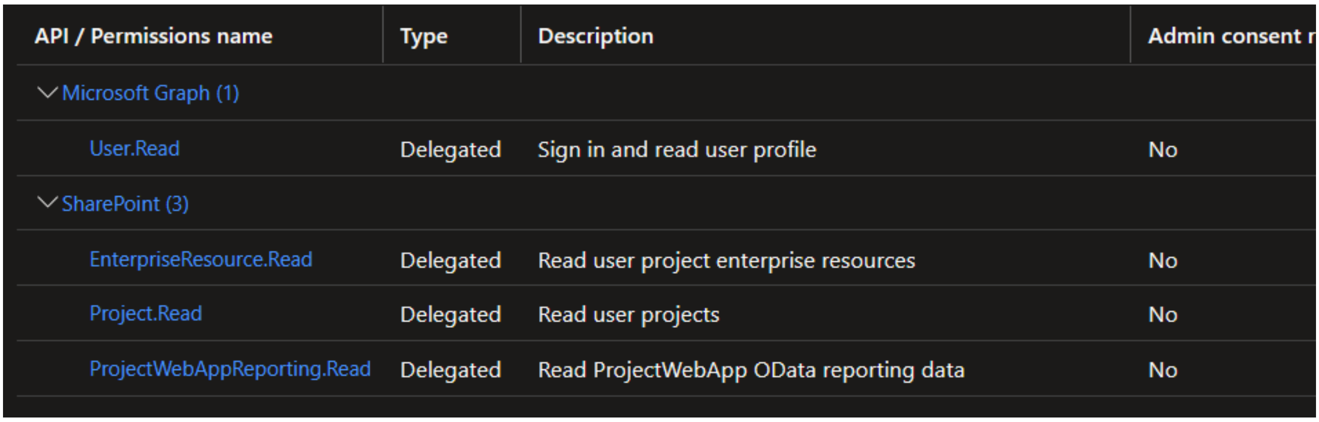
Deployment Guide

App Registration (Authentication)

Create an App registration in AAD with the following permissions.

This will allow our app to sign in on behalf of our user and pull data from `projectonline`.



API / Permissions name	Type	Description	Admin consent r
Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	No
SharePoint (3)			
EnterpriseResource.Read	Delegated	Read user project enterprise resources	No
Project.Read	Delegated	Read user projects	No
ProjectWebAppReporting.Read	Delegated	Read ProjectWebApp OData reporting data	No

> **Note**
>
> This user account needs to have read access to all projects in the Project Online Site Collection

To support with this guide the `clientId` for this app is `cd85557e-65a9-4854-b879-2671dfaee51a`

The `tenantId` is `75e67881-b174-484b-9d30-c581c7ebc177`

You should also create a secret for the app registration - You will need this later

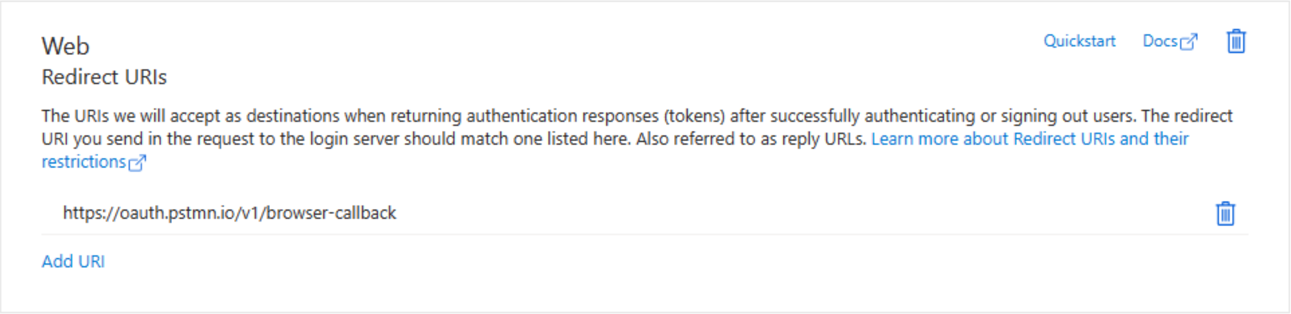
```
eDs8Q~k7XHsc..
```

The app requires a redirect URI - This URI will be where the code is returned that is required to authorize the on behalf of (delegated) application connection

Testing the authentication

For testing purposes the below URI can be used - This will enable the auth code to be returned to a browser

```
https://oauth.pstmn.io/v1/browser-callback
```



Web

Redirect URIs

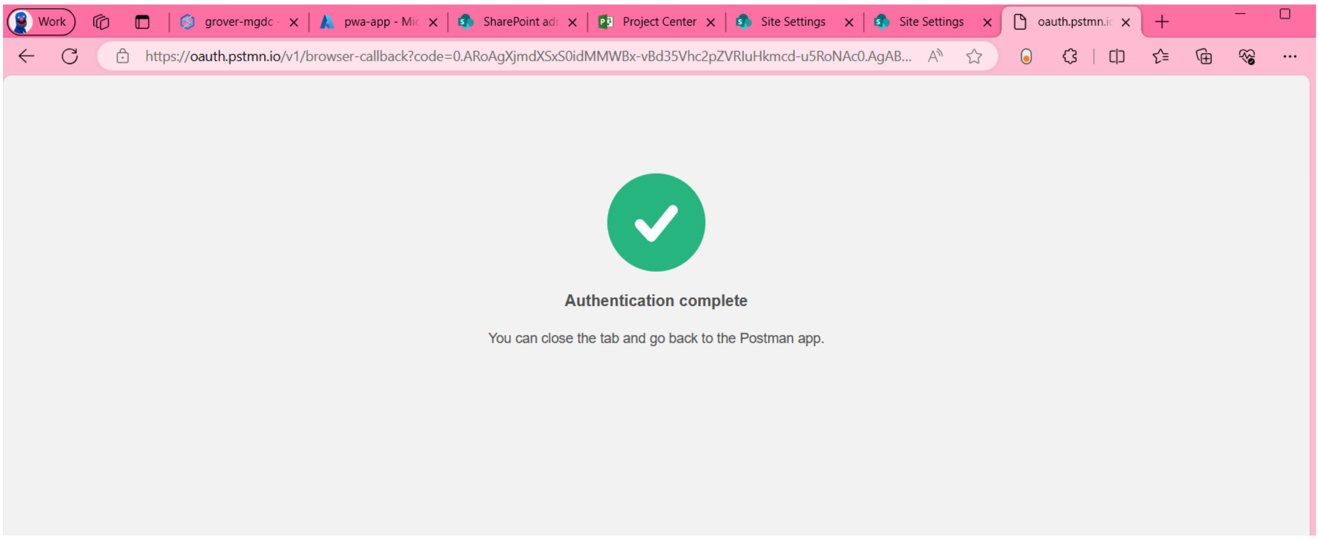
The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating or signing out users. The redirect URI you send in the request to the login server should match one listed here. Also referred to as reply URLs. Learn more about Redirect URIs and their restrictions

https://oauth.pstmn.io/v1/browser-callback

Execute the following HTTP request in a browser (if authenticated you see no prompt). This browser session needs to be authenticated by the user that has access to the Project Online Site Collection.

```
https://login.microsoftonline.com/75e67881-b174-484b-9d30-
c581c7ebc177/oauth2/v2.0/authorize
  ?client_id=cd85557e-65a9-4854-b879-2671dfaee51a
  &response_type=code
  &redirect_uri=https://oauth.pstmn.io/v1/browser-callback
  &response_mode=query&scope=profile openid email
https://graph.microsoft.com/EnterpriseResource.Read
https://graph.microsoft.com/Project.Read
https://graph.microsoft.com/ProjectWebAppReporting.Read
https://graph.microsoft.com/User.Read offline_access&state=12345
```

A authentication test would be indicated as below



https://oauth.pstmn.io/v1/browser-callback?code=0.ARoAgXjmdXSxS0idMMWBx-vBd35Vhc2pZVRIuHkmcd-u5RoNAc0.AgAB...

Authentication complete

You can close the tab and go back to the Postman app.

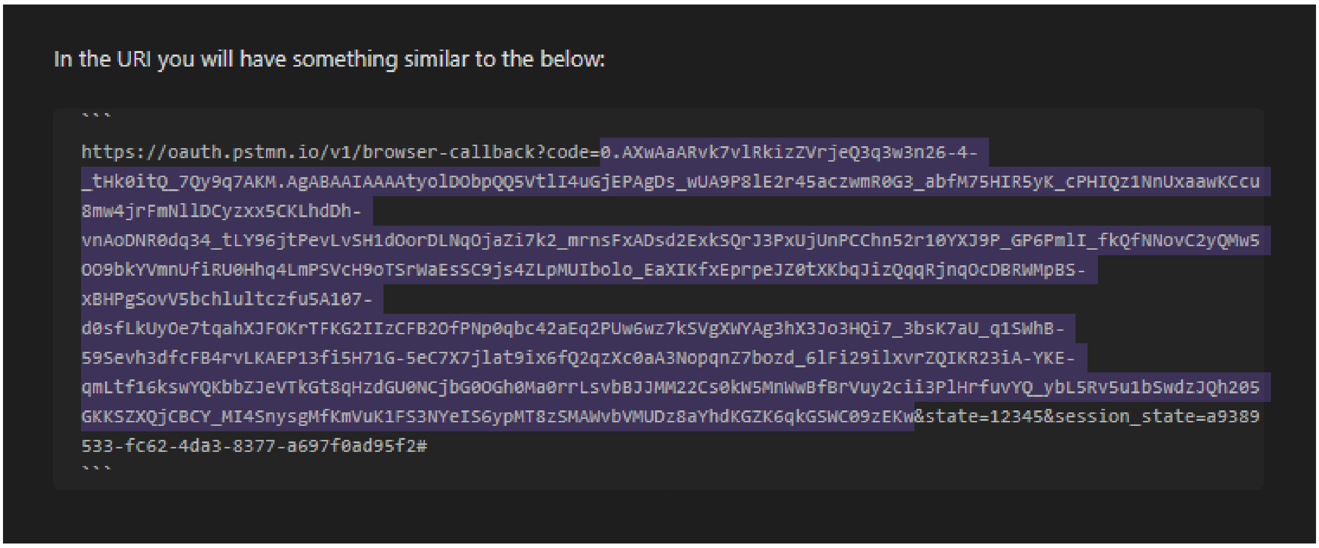
In the URL you will have something similar to the below:

```
https://oauth.pstmn.io/v1/browser-callback?code=0.AXwAaARvk7vlRkizZVrjeQ3q3w3n26-
4-
_tHk0itQ_7Qy9q7AKM.AgABAAIAAAAtyolDObpQQ5VtlI4uGjEPAgDs_wUA9P8lE2r45aczwmR0G3_abf
M75HIR5yK_cPHIQz1NnUxaawKCcu8mw4jrFmNllDCyzxx5CKLhdDh-
```

```
vnAoDNR0dq34_tLY96jtPevLvSH1dOorDLNqOjaZi7k2_mrnsFxADsd2ExkSQrJ3PxUjUnPCChn52r10Y
XJ9P_GP6PmlI_fkQfNNovC2yQMw5OO9bkYVmnUfiRU0Hhq4LmPSVcH9oTSrWaEsSC9js4ZLpMUIbolo_E
aXIKfxEprpeJZ0tXKbqJizQqqRjnqOcDBRWMpBS-xBHPgSovV5bchlultczfu5A107-
d0sfLkUyOe7tqahXJFOKrTFKG2IIzCFB2OfPNp0qbc42aEq2PUw6wz7kSVgXWYAg3hX3Jo3HQi7_3bsK7
aU_q1SWhB-59Sevh3dfcFB4rvLKAEP13fi5H71G-
5eC7X7jlat9ix6fQ2qzXc0aA3NopqnZ7bozd_6lFi29ilxvrZQIKR23iA-YKE-
qmLtf16kswYQKbbZJeVTkGt8qHzdGU0NCjbG0OGh0Ma0rrLsvbBJJMM22Cs0kW5MnWwBfBrVuy2cii3Pl
HrfuvYQ_ybL5Rv5u1bSwdzJQh205GKKSZXQjCBCY_MI4SnysgMfKmVuK1FS3NYeIS6ypMT8zSMAWvbVMU
Dz8aYhdKGZK6qkGSWC09zEKw&state=12345&session_state=a9389533-fc62-4da3-8377-
a697f0ad95f2#
```

Note

The important part of this is the `code` parameter.



In the URI you will have something similar to the below:

```
https://oauth.pstmn.io/v1/browser-callback?code=0.AXwAaARvk7vlRkizZVrjeQ3q3w3n26-4-
_tHk0itQ_7Qy9q7AKM.AgABAAIAAAAtyolDObpQQ5VtlI4uGjEPAgDs_wUA9P8lE2r45aczwmR0G3_abfM75HIR5yK_cPHIQz1NnUxaawKCcu
8mw4jrFmNllDCyzxx5CKLhdDh-
vnAoDNR0dq34_tLY96jtPevLvSH1dOorDLNqOjaZi7k2_mrnsFxADsd2ExkSQrJ3PxUjUnPCChn52r10YXJ9P_GP6PmlI_fkQfNNovC2yQMw5
OO9bkYVmnUfiRU0Hhq4LmPSVcH9oTSrWaEsSC9js4ZLpMUIbolo_EaXIKfxEprpeJZ0tXKbqJizQqqRjnqOcDBRWMpBS-
xBHPgSovV5bchlultczfu5A107-
d0sfLkUyOe7tqahXJFOKrTFKG2IIzCFB2OfPNp0qbc42aEq2PUw6wz7kSVgXWYAg3hX3Jo3HQi7_3bsK7aU_q1SWhB-
59Sevh3dfcFB4rvLKAEP13fi5H71G-5eC7X7jlat9ix6fQ2qzXc0aA3NopqnZ7bozd_6lFi29ilxvrZQIKR23iA-YKE-
qmLtf16kswYQKbbZJeVTkGt8qHzdGU0NCjbG0OGh0Ma0rrLsvbBJJMM22Cs0kW5MnWwBfBrVuy2cii3PlHrfuvYQ_ybL5Rv5u1bSwdzJQh205
GKKSZXQjCBCY_MI4SnysgMfKmVuK1FS3NYeIS6ypMT8zSMAWvbVMUDz8aYhdKGZK6qkGSWC09zEKw&state=12345&session_state=a9389
533-fc62-4da3-8377-a697f0ad95f2#
```

Now that we have the code we can get a delegated app authentication session to MSGraph

Make a HTTP to AAD to get an access token.

The request should be sent to the following URL

`https://login.microsoftonline.com/75e67881-b174-484b-9d30-c581c7ebc177/oauth2/v2.0/token`

The GUID in the URL is the tenant ID

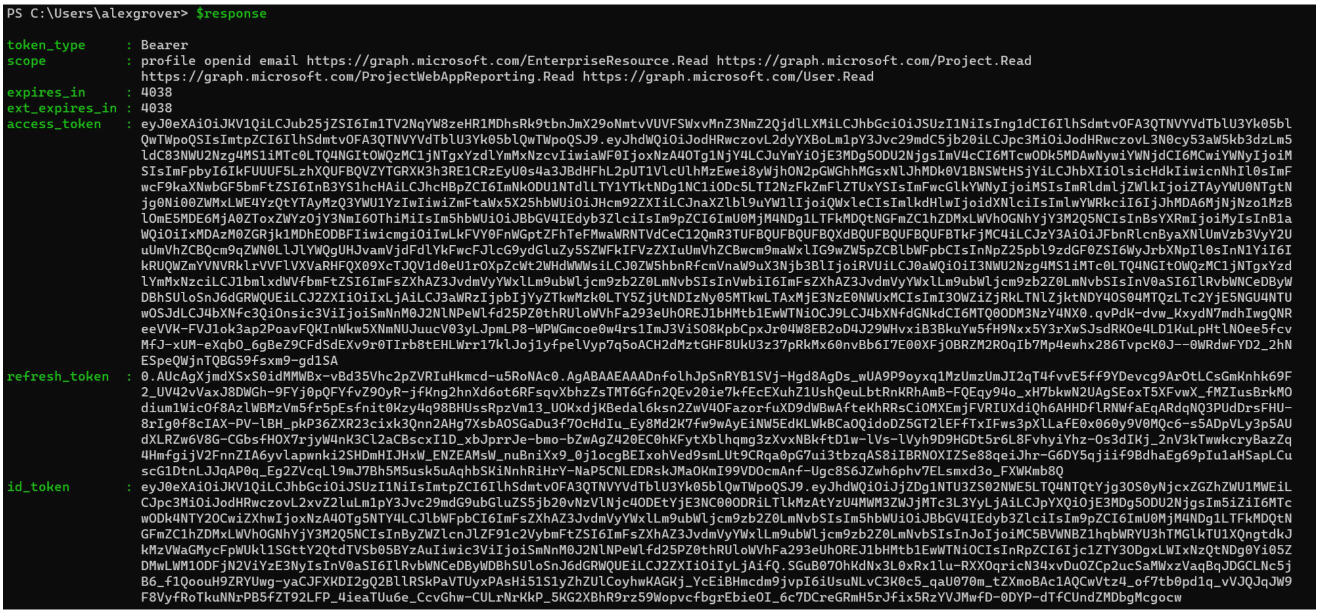
Sample parameters

Key	Value
client_id	cd85557e-65a9-4854-b879-2671dfaee51a
grant_type	authorization_code

Key	Value
redirect_uri	https://oauth.pstmn.io/v1/browser-callback
client_secret	eDs8Q~k7XHscra
scope	profile openid email https://graph.microsoft.com/EnterpriseResource.Read https://graph.microsoft.com/Project.Read https://graph.microsoft.com/ProjectWebAppReporting.Read https://graph.microsoft.com/User.Read offline_access
code	0.AXwAaARvk7vlRkizZVrwvd7BlMjZTXrMxUYnl11Q-mrzATMVk_BHIPT1V5-nRKPksBFIc

```
$tenantId = "75e67881-b174-484b-9d30-c581c7ebc177"
$url = "https://login.microsoftonline.com/$tenantId/oauth2/v2.0/token"

$body = @{
    client_id     = "cd85557e-65a9-4854-b879-2671dfaee51a"
    grant_type    = "authorization_code"
    redirect_uri  = "https://oauth.pstmn.io/v1/browser-callback"
    client_secret = "eDs8Q~k7XHscra"
    scope         = "profile openid email
https://graph.microsoft.com/EnterpriseResource.Read
https://graph.microsoft.com/Project.Read
https://graph.microsoft.com/ProjectWebAppReporting.Read
https://graph.microsoft.com/User.Read offline_access"
    code          = "0.ARoAgXjmdXSxS0idMMWBx-vBd35Vhc2pZVRIuHkmcd..."
}

$response = Invoke-RestMethod -Uri $url -Method Post -Body $body

$response
```

A response like the following indicates success



```
PS C:\Users\alexgrover> $response

token_type     : Bearer
scope          : profile openid email https://graph.microsoft.com/EnterpriseResource.Read https://graph.microsoft.com/Project.Read
                 https://graph.microsoft.com/ProjectWebAppReporting.Read https://graph.microsoft.com/User.Read
expires_in     : 4038
ext_expires_in : 4038
access_token   : eyJ0eXAiOiJKV1QiLCJub25jZSI6Im1TV2NqYW8zeHR1MDhsRk9tbnJmX29oNmtvVUVFSWxvMnZ3NmZ2QjdlLXMiLCJhbGciOiJSUzI1NiIsIng1dCI6IlhSdmtvOFA3QTNVYVdTblU3Yk05bl
                 QwTWpoQSIsImtpZCI6IlhSdmtvOFA3QTNVYVdTblU3Yk05blQwTWpoQSJ9.eyJhdWQiOiJodHRwczovL2dyYXBoLm1pY3Jvc29mdC5jb20iLCJpc3MiOiJodHRwczovL3N0cy53aW5kb3dzLm5
                 ldC83NWU2Nzg4MS1iMTc0LTQ4NGItOWQzMC1jNTgxYzdlYmMxNzcvIiwiaWF0IjoxNzA4OTg1NjY4LCJuYmYiOjE3MDg5ODU2NjgsImV4cCI6MTcwODk5MDAwNywiYWNjdCI6MCwiYWNyIjoiM
                 SIsImFpbyI6IkFUUUF5LzhXQUFBQVZYTGRXK3h3RE1CRzEyU0s4a3JBdHFhL2pUT1VlcUlhMzEwei8yWjhON2pGWGhhMGsxNlJhMDk0V1BNSWtHSjYiLCJhbXIiOlsicHdkIiwicnNhIl0sImF
                 wcF9kaXNwbGF5bmFtZSI6InB3YS1hcHAiLCJhcHBpZCI6ImNkODU1NTdlLTY1YTktNDg1NC1iODc5LTI2NzFkZmFlZTUxYSIsImFwcGlkYWNyIjoiMSIsImRldmljZWlkIjoiZTAyYWU0NTgtN
                 jg0Ni00ZWMxLWE4YzQtYTAyMzQ3YWU1YzIwIiwiZmFtaWx5X25hbWUiOiJHcm92ZXIiLCJnaXZlbl9uYW1lIjoiQWxleCIsImlkdHlwIjoidXNlciIsImlwYWRkciI6IjJhMDA6MjNjNzo1MzB
                 lOmE5MDE6MjA0ZToxZWYzOjY3NmI6OThiMiIsIm5hbWUiOiJBbGV4IEdyb3ZlciIsIm9pZCI6ImU0MjM4NDg1LTFkMDQtNGFmZC1hZDMxLWVhOGNhYjY3M2Q5NCIsInBsYXRmIjoiMyIsInB1a
                 WQiOiIxMDAzM0ZGRjk1MDhEODBFIiwicmgiOiIwLkFVY0FnWGptZFhTeFMwaWRNTVdCeC12QmR3TUFBQUFBQUFBQXdBQUFBQUFBQUFBTkFjMC4iLCJzY3AiOiJFbnRlcnByaXNlUmVzb3VyY2U
                 uUmVhZCBQcm9qZWN0LlJlYWQgUHJvamVjdFdlYkFwcFJlcG9ydGluZy5SZWFkIFVzZXIuUmVhZCBwcm9maWxlIG9wZW5pZCBlbWFpbCIsInNpZ25pbl9zdGF0ZSI6WyJrbXNpIl0sInN1YiI6I
                 kRUQWZmYVNVRklrVVFlVXVaRHFQX09XcTJQV1d0eU1rOXpZcWt2WHdWWWsiLCJ0ZW5hbnRfcmVnaW9uX3Njb3BlIjoiRVUiLCJ0aWQiOiI3NWU2Nzg4MS1iMTc0LTQ4NGItOWQzMC1jNTgxYzd
                 lYmMxNzciLCJ1bmlxdWVfbmFtZSI6ImFsZXhAZ3JvdmVyYWxlLm9ubWljcm9zb2Z0LmNvbSIsInVwbiI6ImFsZXhAZ3JvdmVyYWxlLm9ubWljcm9zb2Z0LmNvbSIsInV0aSI6IlRvbWNCeDByW
                 DBhSUloSnJ6dGRWQUEiLCJ2ZXIiOiIxLjAiLCJ3aWRzIjpbIjYyZTkwMzk0LTY5ZjUtNDIzNy05MTkwLTAxMjE3NzE0NWUxMCIsImI3OWZiZjRkLTNlZjktNDY4OS04MTQzLTc2YjE5NGU4NTU
                 wOSJdLCJ4bXNfc3QiOnsic3ViIjoiSmNnM0J2NlNPeWlfd25PZ0thRUloWVhFa293eUhOREJ1bHMtb1EwWTNiOCJ9LCJ4bXNfdGNkdCI6MTQ0ODM3NzY4NX0.qvPdK-dvw_KxydN7mdhIwgQNR
                 eeVVK-FVJ1ok3ap2PoavFQKInWkw5XNmNUJuucV03yLJpmLP8-WPWGmcoe0w4rs1ImJ3ViSO8KpbCpxJr04W8EB2oD4J29WHvxiB3BkuYw5fH9Nxx5Y3rXwSJsdRKOe4LD1KuLpHtlNOee5fcv
                 MfJ-xUM-eXqbO_6gBeZ9CFdSdEXv9r0TIrb8tEHLWrr17klJoj1yfpelVyp7q5oACH2dMztGHF8UkU3z37pRkMx60nvBb6I7E00XFjOBRZM2ROqIb7Mp4ewhx286TvpcK0J--0WRdwFYD2_2hN
                 ESpeQWjnTQBG59fsxm9-gd1SA
refresh_token  : 0.AUcAgXjmdXSxS0idMMWBx-vBd35Vhc2pZVRIuHkmcd-u5RoNAc0.AgABAAEAAADnfolhJpSnRYB1SVj-Hgd8AgDs_wUA9P9oyxq1MzUmzUmJI2qT4fvvE5ff9YDevcg9ArOtLCsGmKnhk69F
                 2_UV42vVaxJ8DWGh-9FYj0pQFYfvZ9OyR-jfKng2hnXd6ot6RFsqvXbhzZsTMT6Gfn2QEv20ie7kfEcEXuhZ1UshQeuLbtRnKRhAmB-FQEqy94o_xH7bkwN2UAgSEoxT5XFvwX_fMZIusBrkMO
                 dium1WicOf8AzlWBMzVm5fr5pEsfnit0Kzy4q98BHUssRpzVm13_UOKxdjKBedal6ksn2ZwV4OFazorfuXD9dWBwAfteKhRRsCiOMXEmjFVRIUXdiQh6AHHDflRNWfaEqARdqNQ3PUdDrsFHU-
                 8rIg0f8cIAX-PV-lBH_pkP36ZXR23cixk3Qnn2AHg7XsbAOSGaDu3f7OcHdIu_Ey8Md2K7fw9wAyEiNW5EdKLWkBCaOQidoDZ5GT2lEFfTxIFws3pXlLafE0x060y9V0MQc6-s5ADpVLy3p5AU
                 dXLRZw6V8G-CGbsfHOX7rjyW4nK3Cl2aCBscxI1D_xbJprrJe-bmo-bZwAgZ420EC0hKFytXblhqmg3zXvxNBkftD1w-lVs-lVyh9D9HGDt5r6L8FvhyiYhz-Os3dIKj_2nV3kTwwkcryBazZq
                 4HmfgijV2FnnZIA6yvlapwnki2SHDmHIJHxW_ENZEAMsW_nuBniXx9_0j1ocgBEIxohVed9smLUt9CRqa0pG7ui3tbzqAS8iIBRNOXIZSe88qeiJhr-G6DY5qjiif9BdhaEg69pIu1aHSapLCu
                 scG1DtnLJJqAP0q_Eg2ZVcqLl9mJ7Bh5M5usk5uAqhbSKiNnhRiHrY-NaP5CNLEDRskJMaOKmI99VDOcmAnf-Ugc8S6JZwh6phv7ELsmxd3o_FXWKmb8Q
id_token       : eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6IlhSdmtvOFA3QTNVYVdTblU3Yk05blQwTWpoQSJ9.eyJhdWQiOiJjZDg1NTU3ZS02NWE5LTQ4NTQtYjg3OS0yNjcxZGZhZWU1MWEiL
                 CJpc3MiOiJodHRwczovL2xvZ2luLm1pY3Jvc29mdG9ubGluZS5jb20vNzVlNjc4ODEtYjE3NC00ODRiLTlkMzAtYzU4MWM3ZWJjMTc3L3YyLjAiLCJpYXQiOjE3MDg5ODU2NjgsIm5iZiI6MTc
                 wODk4NTY2OCwiZXhwIjoxNzA4OTg5NTY4LCJlbWFpbCI6ImFsZXhAZ3JvdmVyYWxlLm9ubWljcm9zb2Z0LmNvbSIsIm5hbWUiOiJBbGV4IEdyb3ZlciIsIm9pZCI6ImU0MjM4NDg1LTFkMDQtN
                 GFmZC1hZDMxLWVhOGNhYjY3M2Q5NCIsInByZWZlcnJlZF91c2VybmFtZSI6ImFsZXhAZ3JvdmVyYWxlLm9ubWljcm9zb2Z0LmNvbSIsInJoIjoiMC5BVWNBZ1hqbWRYU3hTMGlkTU1XQngtdkJ
                 kMzVWaGMycFpWUkl1SGttY2QtdTVSb05BYzAuIiwic3ViIjoiSmNnM0J2NlNPeWlfd25PZ0thRUloWVhFa293eUhOREJ1bHMtb1EwWTNiOCIsInRpZCI6Ijc1ZTY3ODgxLWIxNzQtNDg0Yi05Z
                 DMwLWM1ODFjN2ViYzE3NyIsInV0aSI6IlRvbWNCeDByWDBhSUloSnJ6dGRWQUEiLCJ2ZXIiOiIyLjAifQ.SGuB07OhKdNx3L0xRx1lu-RXXOqricN34xvDuOZCp2ucSaMWxzVaqBqJDGCLNc5j
                 B6_f1QoouH9ZRYUwg-yaCJFXKDI2gQ2BllRSkPaVTUyxPAsHi51S1yZhZUlCoyhwKAGKj_YcEiBHmcdm9jvpI6iUsuNLvC3K0c5_qaU070m_tZXmoBAc1AQCwVtz4_of7tb0pd1q_vVJQJqJW9
                 F8VyfRoTkuNNrPB5fZT92LFP_4ieaTUu6e_CcvGhw-CULrNrKkP_5KG2XBhR9rz59WopvcfbgrEbieOI_6c7DCreGRmH5rJfix5RzYVJMwfD-0DYP-dTfCUndZMDbgMcgocw
```

The next step is to swap this Graph access token for an SPO token. We can do this using the refresh token that was returned in the above request.

The request should be sent to the same URL as before

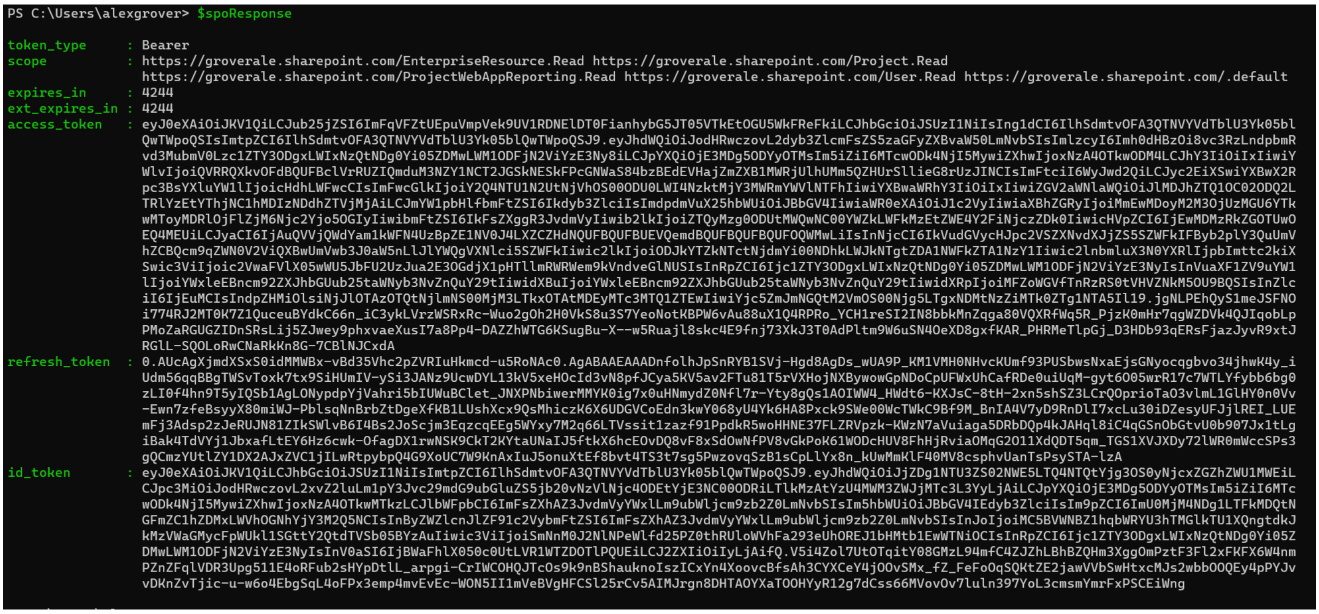
`https://login.microsoftonline.com/75e67881-b174-484b-9d30-c581c7ebc177/oauth2/v2.0/token

The GUID in the URL is the tenant ID

Parameters are slightly different

Key	Value
client_id	cd85557e-65a9-4854-b879-2671dfaee51a
grant_type	refresh_token
refresh_token	0.AXwAaARvk7vlRkizZVrjeQ3q3w3n26-4-.....
client_secret	eDs8Q~k7XHscra
scope	https://m365x82565687.sharepoint.com/.default

```
$tenantId = "75e67881-b174-484b-9d30-c581c7ebc177"
$url = "https://login.microsoftonline.com/$tenantId/oauth2/v2.0/token"

$body = @{
    client_id      = "cd85557e-65a9-4854-b879-2671dfaee51a"
    grant_type     = "refresh_token"
    refresh_token  = $response.refresh_token
    client_secret  = "eDs8Q~k7XHscraR"
    scope          = "https://groverale.sharepoint.com/.default"
}

$spoResponse = Invoke-RestMethod -Uri $url -Method Post -Body $body

$spoResponse
```



```
PS C:\Users\alexgrover> $spoResponse

token_type     : Bearer
scope          : https://groverale.sharepoint.com/EnterpriseResource.Read https://groverale.sharepoint.com/Project.Read
                 https://groverale.sharepoint.com/ProjectWebAppReporting.Read https://groverale.sharepoint.com/User.Read https://groverale.sharepoint.com/.default
expires_in     : 4244
ext_expires_in : 4244
access_token   : eyJ0eXAiOiJKV1QiLCJub25jZSI6ImFqVFZtUEpuVmpVek9UV1RDNElDT0FianhybG5JT05VTkEtOGU5WkFReFkiLCJhbGciOiJSUzI1NiIsIng1dCI6IlhSdmtvOFA3QTNVYVdTblU3Yk05bl
                 QwTWpoQSIsImtpZCI6IlhSdmtvOFA3QTNVYVdTblU3Yk05blQwTWpoQSJ9.eyJhdWQiOiJodHRwczovL2dyb3ZlcmFsZS5zaGFyZXBvaW50LmNvbSIsImlzcyI6Imh0dHBzOi8vc3RzLndpbmR
                 vd3MubmV0Lzc1ZTY3ODgxLWIxNzQtNDg0Yi05ZDMwLWM1ODFjN2ViYzE3Ny8iLCJpYXQiOjE3MDg5ODYyOTMsIm5iZiI6MTcwODk4NjI5MywiZXhwIjoxNzA4OTkwODM4LCJhY3IiOiIxIiwiY
                 WlvIjoiQVRRQXkvOFdBQUFBclVrRUZIQmduM3NZY1NCT2JGSkNESkFPcGNWaS84bzBEdEVHajZmZXB1MWRjUlhUMm5QZHUrSllieG8rUzJINCIsImFtciI6WyJwd2QiLCJyc2EiXSwiYXBwX2R
                 pc3BsYXluYW1lIjoicHdhLWFwcCIsImFwcGlkIjoiY2Q4NTU1N2UtNjVhOS00ODU0LWI4NzktMjY3MWRmYWVlNTFhIiwiYXBwaWRhY3IiOiIxIiwiZGV2aWNlaWQiOiJlMDJhZTQ1OC02ODQ2L
                 TRlYzEtYThjNC1hMDIzNDdhZTVjMjAiLCJmYW1pbHlfbmFtZSI6Ikdyb3ZlciIsImdpdmVuX25hbWUiOiJBbGV4IiwiaWR0eXAiOiJ1c2VyIiwiaXBhZGRyIjoiMmEwMDoyM2M3OjUzMGU6YTk
                 wMToyMDRlOjFlZjM6Njc2Yjo5OGIyIiwibmFtZSI6IkFsZXggR3JvdmVyIiwib2lkIjoiZTQyMzg0ODUtMWQwNC00YWZkLWFkMzEtZWE4Y2FiNjczZDk0IiwicHVpZCI6IjEwMDMzRkZGOTUwO
                 EQ4MEUiLCJyaCI6IjAuQVVjQWdYam1kWFN4UzBpZE1NV0J4LXZCZHdNQUFBQUFBUEVQemdBQUFBQUFBQUFOQWMwLiIsInNjcCI6IkVudGVycHJpc2VSZXNvdXJjZS5SZWFkIFByb2plY3QuUmV
                 hZCBQcm9qZWN0V2ViQXBwUmVwb3J0aW5nLlJlYWQgVXNlci5SZWFkIiwic2lkIjoiODJkYTZkNTctNjdmYi00NDhkLWJkNTgtZDA1NWFkZTA1NzY1Iiwic2lnbmluX3N0YXRlIjpbImttc2kiX
                 Swic3ViIjoic2VwaFVlX05wWU5JbFU2UzJua2E3OGdjX1pHTllmRWRWem9kVndveGlNUSIsInRpZCI6Ijc1ZTY3ODgxLWIxNzQtNDg0Yi05ZDMwLWM1ODFjN2ViYzE3NyIsInVuaXF1ZV9uYW1
                 lIjoiYWxleEBncm92ZXJhbGUub25taWNyb3NvZnQuY29tIiwidXBuIjoiYWxleEBncm92ZXJhbGUub25taWNyb3NvZnQuY29tIiwidXRpIjoiMFZoWGVfTnRzRS0tVHVZNkM5OU9BQSIsInZlc
                 iI6IjEuMCIsIndpZHMiOlsiNjJlOTAzOTQtNjlmNS00MjM3LTkxOTAtMDEyMTc3MTQ1ZTEwIiwiYjc5ZmJmNGQtM2VmOS00Njg5LTgxNDMtNzZiMTk0ZTg1NTA5Il19.jgNLPEhQyS1meJSFNO
                 i774RJ2MT0K7Z1QuceuBYdkC66n_iC3ykLVrzWSRxRc-Wuo2gOh2H0VkS8u3S7YeoNotKBPW6vAu88uX1Q4RPRo_YCH1reSI2IN8bbkMnZqga80VQXRfWq5R_PjzK0mHr7qgWZDVk4QJIqobLp
                 PMoZaRGUGZIDnSRsLij5ZJwey9phxvaeXusI7a8Pp4-DAZZhWTG6KSugBu-X--w5Ruajl8skc4E9fnj73XkJ3T0AdPltm9W6uSN4OeXD8gxfKAR_PHRMeTlpGj_D3HDb93qERsFjazJyvR9xtJ
                 RGlL-SQOLoRwCNaRkKn8G-7CBlNJCxdA
refresh_token  : 0.AUcAgXjmdXSxS0idMMWBx-vBd35Vhc2pZVRIuHkmcd-u5RoNAc0.AgABAAEAAADnfolhJpSnRYB1SVj-Hgd8AgDs_wUA9P_KM1VMH0NHvcKUmf93PUSbwsNxaEjsGNyocqgbvo34jhwK4y_i
                 Udm56qqBBgTWSvToxk7tx9SiHUmIV-ySi3JANz9UcwDYL13kV5xeHOcId3vN8pfJCya5KV5av2FTu81T5rVXHojNXBywowGpNDoCpUFWxUhCafRDe0uiUqM-gyt6O05wrR17c7WTLYfybb6bg0
                 zLI0f4hn9T5yIQSb1AgLONypdpYjVahri5bIUWuBClet_JNXPNbiwerMMYK0ig7x0uHNmydZ0Nfl7r-Yty8gQs1AOIWW4_HWdt6-KXJsC-8tH-2xn5shSZ3LCrQOprioTaO3vlmL1GlHY0n0Vv
                 -Ewn7zfeBsyyX80miWJ-PblsqNnBrbZtDgeXfKB1LUshXcx9QsMhiczK6X6UDGVCoEdn3kwY068yU4Yk6HA8Pxck9SWe00WcTWkC9Bf9M_BnIA4V7yD9RnDlI7xcLu30iDZesyUFJjlREI_LUE
                 mFj3Adsp2zJeRUJN81ZIkSWlvB6I4Bs2JoScjm3EqzcqEEg5WYxy7M2q66LTVssit1zazf91PpdkR5woHHNE37FLZRVpzk-KWzN7aVuiaga5DRbDQp4kJAHql8iC4qGSnObGtvU0b907Jx1tLg
                 iBak4TdVYj1JbxafLtEY6Hz6cwk-OfagDX1rwNSK9CkT2KYtaUNaIJ5ftkX6hcEOvDQ8vF8xSdOwNfPV8vGkPoK61WODcHUV8FhHjRviaOMqG2O11XdQDT5qm_TGS1XVJXDy72lWR0mWccSPs3
                 gQCmzYUtlZY1DX2AJxZVC1jILwRtpybpQ4G9XoUC7W9KnAxIuJ5onuXtEf8bvt4TS3t7sg5PwzovqSzB1sCpLlYx8n_kUwMmKlF40MV8csphvUanTsPsySTA-lzA
id_token       : eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiIsImtpZCI6IlhSdmtvOFA3QTNVYVdTblU3Yk05blQwTWpoQSJ9.eyJhdWQiOiJjZDg1NTU3ZS02NWE5LTQ4NTQtYjg3OS0yNjcxZGZhZWU1MWEiL
                 CJpc3MiOiJodHRwczovL2xvZ2luLm1pY3Jvc29mdG9ubGluZS5jb20vNzVlNjc4ODEtYjE3NC00ODRiLTlkMzAtYzU4MWM3ZWJjMTc3L3YyLjAiLCJpYXQiOjE3MDg5ODYyOTMsIm5iZiI6MTc
                 wODk4NjI5MywiZXhwIjoxNzA4OTkwMTkzLCJlbWFpbCI6ImFsZXhAZ3JvdmVyYWxlLm9ubWljcm9zb2Z0LmNvbSIsIm5hbWUiOiJBbGV4IEdyb3ZlciIsIm9pZCI6ImU0MjM4NDg1LTFkMDQtN
                 GFmZC1hZDMxLWVhOGNhYjY3M2Q5NCIsInByZWZlcnJlZF91c2VybmFtZSI6ImFsZXhAZ3JvdmVyYWxlLm9ubWljcm9zb2Z0LmNvbSIsInJoIjoiMC5BVWNBZ1hqbWRYU3hTMGlkTU1XQngtdkJ
                 kMzVWaGMycFpWUkl1SGttY2QtdTVSb05BYzAuIiwic3ViIjoiSmNnM0J2NlNPeWlfd25PZ0thRUloWVhFa293eUhOREJ1bHMtb1EwWTNiOCIsInRpZCI6Ijc1ZTY3ODgxLWIxNzQtNDg0Yi05Z
                 DMwLWM1ODFjN2ViYzE3NyIsInV0aSI6IjBWaFhlX050c0UtLVR1WTZDOTlPQUEiLCJ2ZXIiOiIyLjAifQ.V5i4Zol7UtOTqitY08GMzL94mfC4ZJZhLBhBZQHm3XggOmPztF3Fl2xFKFX6W4nm
                 PZnZFqlVDR3Upg511E4oRFub2sHYpDtlL_arpgi-CrIWCOHQJTcOs9k9nBShauknoIszICxYn4XoovcBfsAh3CYXCeY4jOOvSMx_fZ_FeFoOqSQKtZE2jawVVbSwHtxcMJs2wbbOOQEy4pPYJv
                 vDKnZvTjic-u-w6o4EbgSqL4oFPx3emp4mvEvEc-WON5II1mVeBVgHFCSl25rCv5AIMJrgn8DHTAOYXaTOOHYyR12g7dCss66MVovOv7luln397YoL3cmsmYmrFxPSCEiWng
```

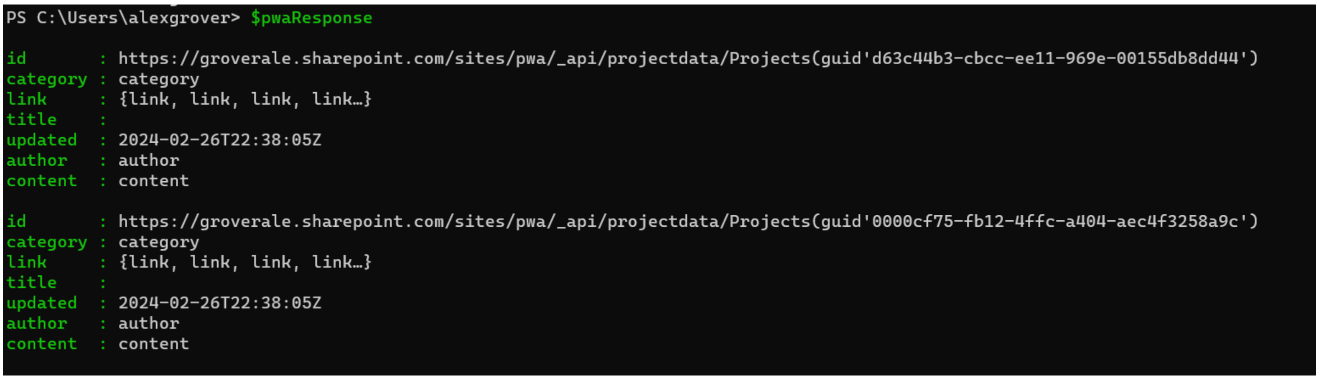
It's now possible to call the Project Online rest APIs using the access token returned in the `spoResponse`.

To test this we can make another rest call, but this time to the ProjectOnline APIs

`https://groverale.sharepoint.com/sites/pwa/_api/projectdata/Projects`

```
$projectOnlineAPI =
"https://groverale.sharepoint.com/sites/pwa/_api/projectdata/Projects"  # Replace
this with your PWA endpoint

$headers = @{
    "Authorization" = "Bearer $($spoResponse.access_token)"
}

$pwaResponse = Invoke-RestMethod -Uri $projectOnlineAPI -Method Get -Headers
$headers

$pwaResponse
```

A list of projects indicates success



```
PS C:\Users\alexgrover> $pwaResponse

id       : https://groverale.sharepoint.com/sites/pwa/_api/projectdata/Projects(guid'd63c44b3-cbcc-ee11-969e-00155db8dd44')
category : category
link     : {link, link, link, link…}
title    :
updated  : 2024-02-26T22:38:05Z
author   : author
content  : content

id       : https://groverale.sharepoint.com/sites/pwa/_api/projectdata/Projects(guid'0000cf75-fb12-4ffc-a404-aec4f3258a9c')
category : category
link     : {link, link, link, link…}
title    :
updated  : 2024-02-26T22:38:05Z
author   : author
content  : content
```

Token Expiration

Refresh token is active for 90 days - So as long as the app does something every 90 days the token should be refreshed and we don't need to do anything

This is confirmed here, has long as the app is in daily use the refresh token will last forever

Refresh tokens in the Microsoft identity platform - Microsoft identity platform | Microsoft Learn

Token Revocation

Refresh token can be revoked for a number of reasons. Our refresh token has been issued to a confidential clients so the following table explains the instances when the refresh token will be revoked

Change	Confidential client token
Password expires	Stays alive
Password changed by user	Stays alive
User does SSPR	Stays alive
Admin resets password	Stays alive
User revokes their refresh tokens	Revoked
Admin revokes all refresh tokens for a user	Revoked
Single sign-out	Stays alive

Refresh tokens in the Microsoft identity platform - Microsoft identity platform | Microsoft Learn

Deploy Azure Components

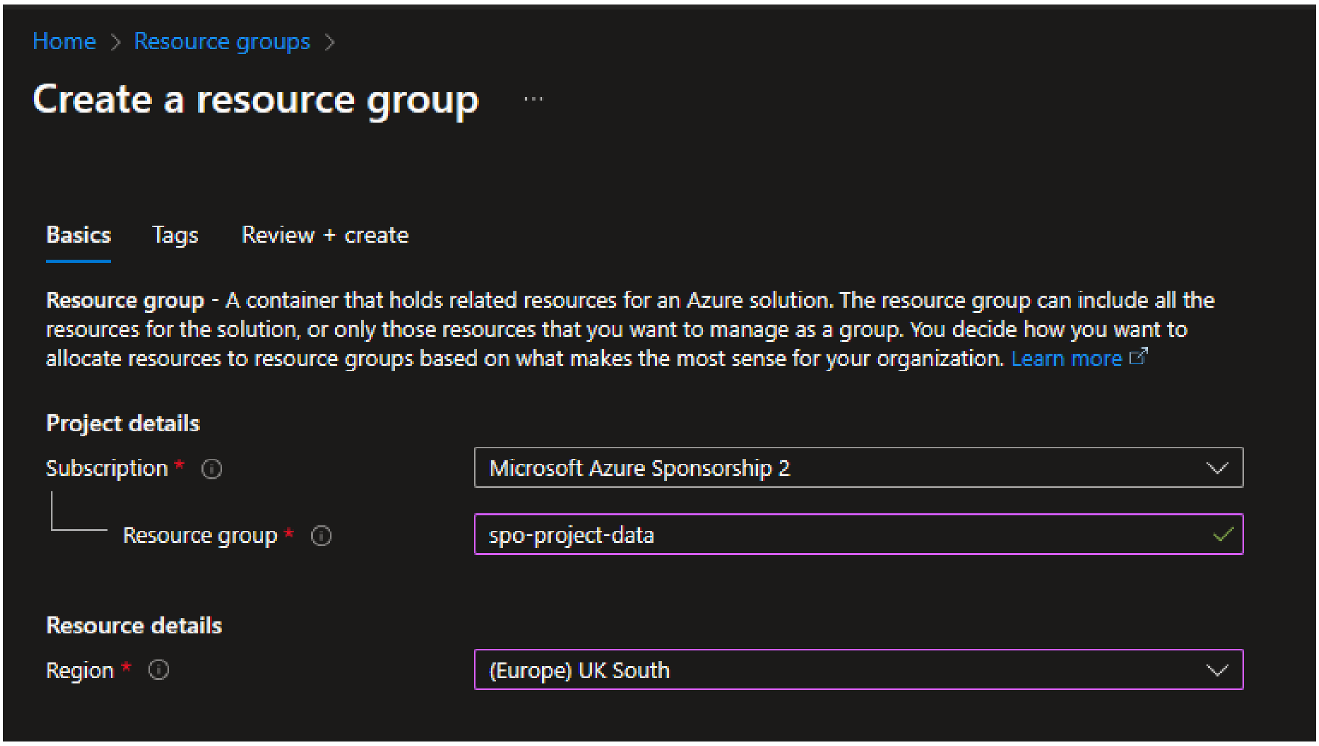
An Azure function has been developed as the vehicle to automate the process above and also send the ProjectOnline data to a SQL database.

We need the following

- Resource group
- Azure SQL db
- Azure Function, with managed identity
- KeyVault, access policies

Resource Group

A resource group will be uses as a container for all our resources



Create a resource group ···

Resource group - A container that holds related resources for an Azure solution. The resource group can include all the resources for the solution, or only those resources that you want to manage as a group. You decide how you want to allocate resources to resource groups based on what makes the most sense for your organization. Learn more

Project details

Subscription * : Microsoft Azure Sponsorship 2

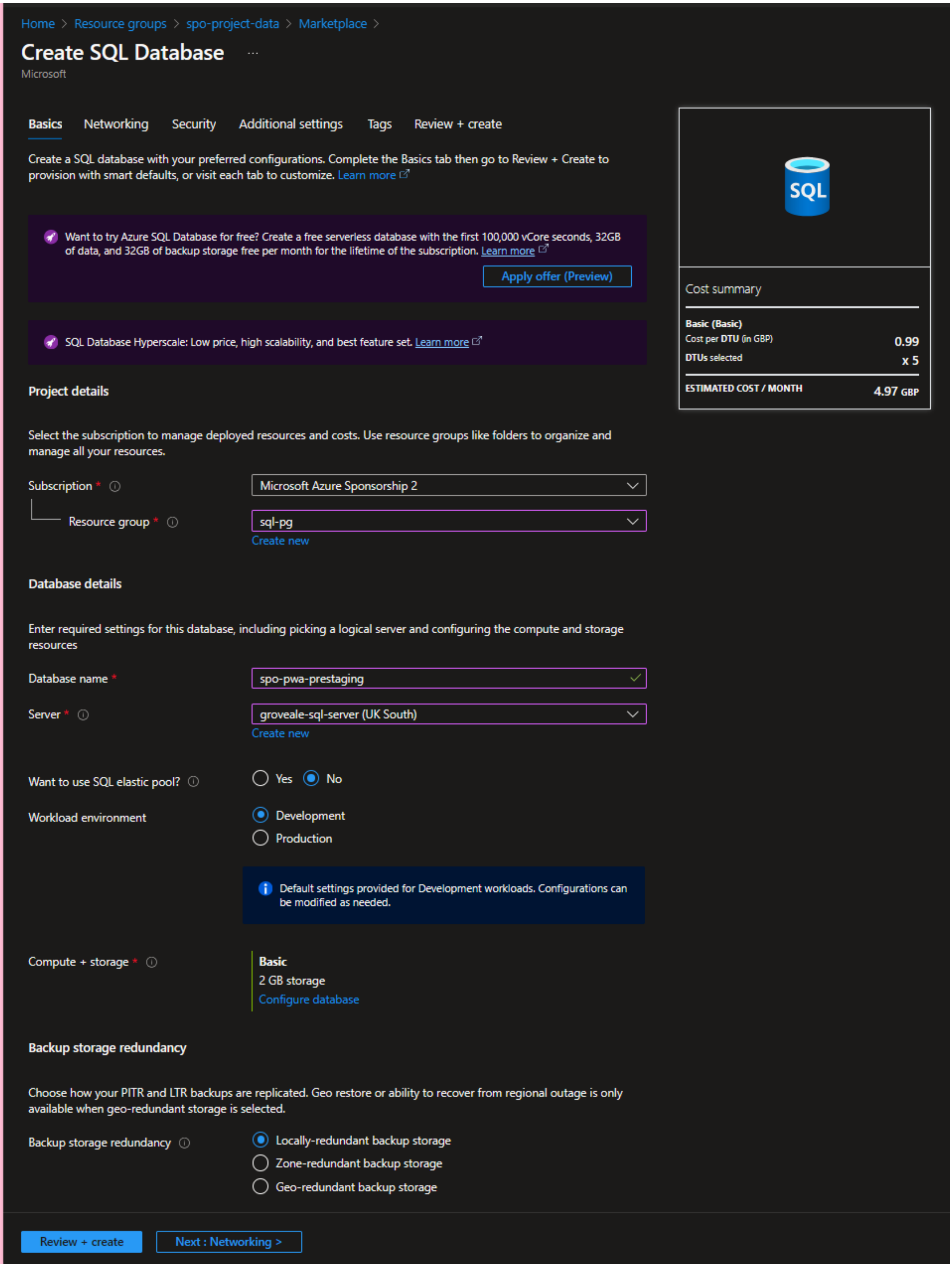
Resource group * : spo-project-data ✓

Resource details

Region * : (Europe) UK South

Azure SQL database

You may already have an existing db but for testing purpose you may want to deploy another. I have used an existing SQL server to host my db so the resource group is different to what we created above



Create SQL Database ...

Microsoft

Create a SQL database with your preferred configurations. Complete the Basics tab then go to Review + Create to provision with smart defaults, or visit each tab to customize. Learn more

Want to try Azure SQL Database for free? Create a free serverless database with the first 100,000 vCore seconds, 32GB of data, and 32GB of backup storage free per month for the lifetime of the subscription. Learn more

SQL Database Hyperscale: Low price, high scalability, and best feature set. Learn more

Cost summary

Basic (Basic)

Cost per DTU (in GBP)	0.99
DTUs selected	x 5
ESTIMATED COST / MONTH	4.97 GBP

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : Microsoft Azure Sponsorship 2

Resource group * : sql-pg

Create new

Database details

Enter required settings for this database, including picking a logical server and configuring the compute and storage resources

Database name * : spo-pwa-prestaging ✓

Server * : groveale-sql-server (UK South)

Create new

Want to use SQL elastic pool? ○ Yes ● No

Workload environment ● Development ○ Production

Default settings provided for Development workloads. Configurations can be modified as needed.

Compute + storage * : Basic
2 GB storage
Configure database

Backup storage redundancy

Choose how your PITR and LTR backups are replicated. Geo restore or ability to recover from regional outage is only available when geo-redundant storage is selected.

Backup storage redundancy
● Locally-redundant backup storage
○ Zone-redundant backup storage
○ Geo-redundant backup storage

SQL Auth

Got to Connection strings and note down the ADO.NET (SQL authentication) property.

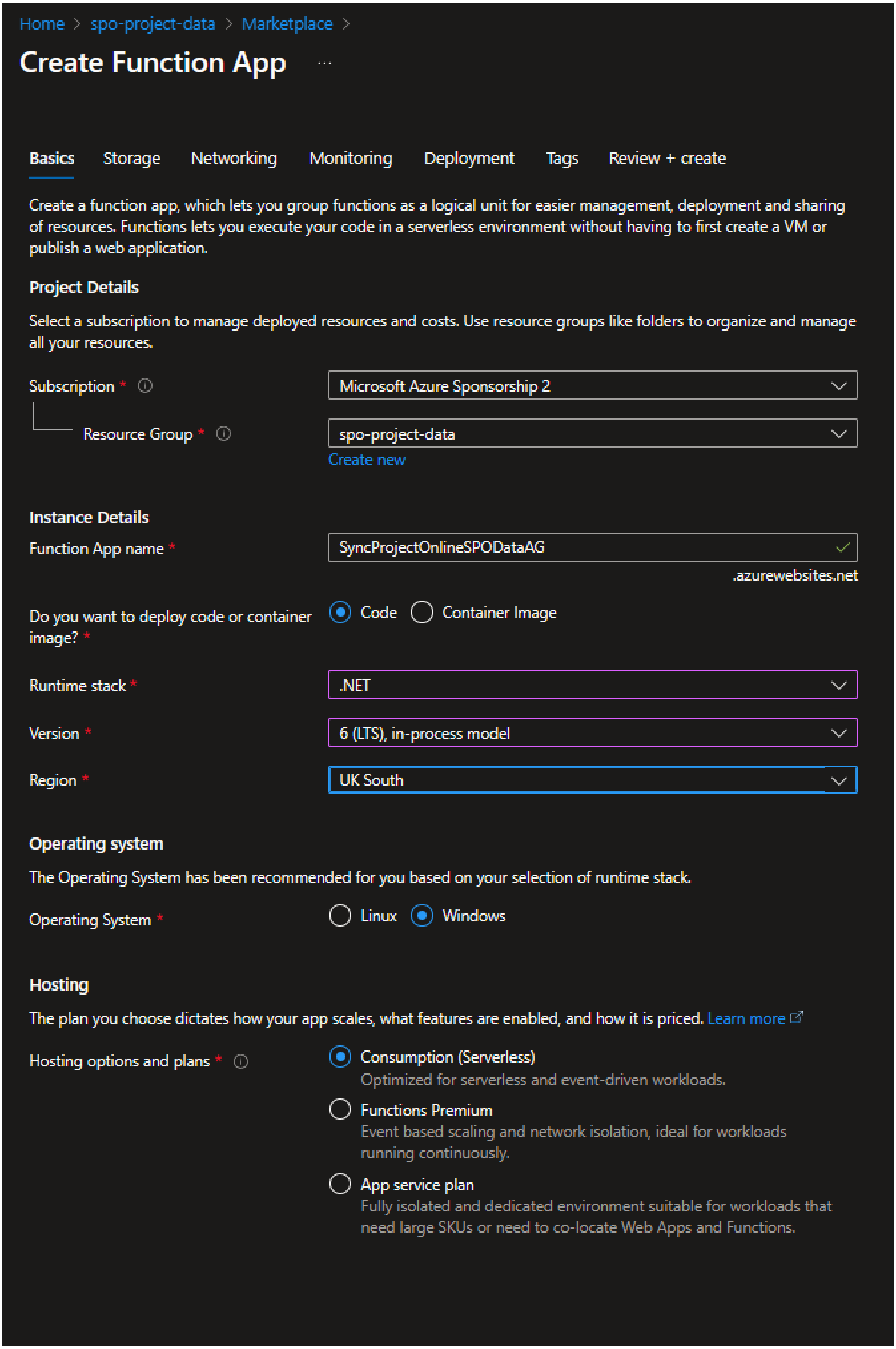
This will be needed later

Azure Function

Create an Azure function with the below configuration.

Notable settings.

- .NET Runtime
- Version 6 (LTS), in-process model
- Windows



Create a function app, which lets you group functions as a logical unit for easier management, deployment and sharing of resources. Functions lets you execute your code in a serverless environment without having to first create a VM or publish a web application.

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription *	Microsoft Azure Sponsorship 2
Resource Group *	spo-project-data Create new

Instance Details

Function App name *	SyncProjectOnlineSPODataAG ✓ .azurewebsites.net
Do you want to deploy code or container image? *	● Code ○ Container Image
Runtime stack *	.NET
Version *	6 (LTS), in-process model
Region *	UK South

Operating system

The Operating System has been recommended for you based on your selection of runtime stack.

Operating System * ○ Linux ● Windows

Hosting

The plan you choose dictates how your app scales, what features are enabled, and how it is priced. Learn more

Hosting options and plans *

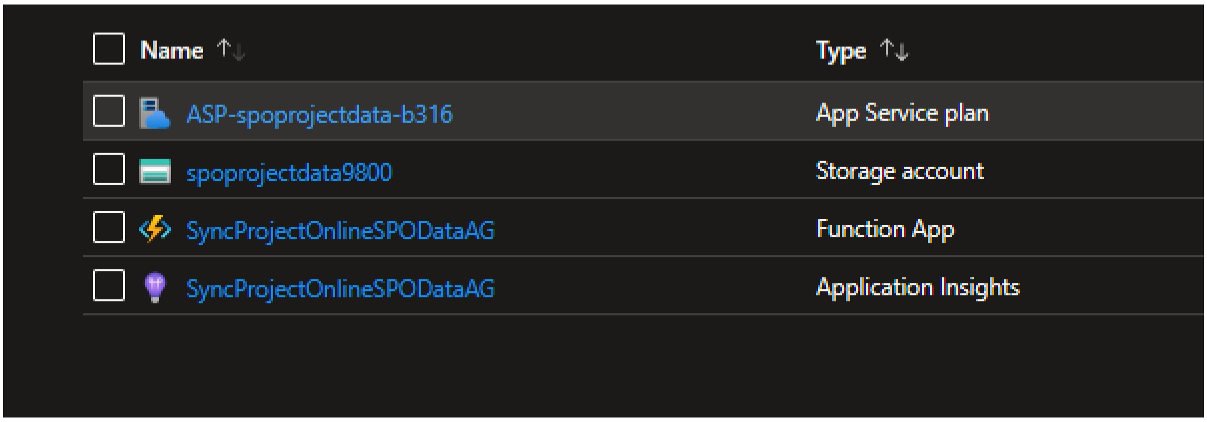
● Consumption (Serverless)
Optimized for serverless and event-driven workloads.

○ Functions Premium
Event based scaling and network isolation, ideal for workloads running continuously.

○ App service plan
Fully isolated and dedicated environment suitable for workloads that need large SKUs or need to co-locate Web Apps and Functions.

If you create with the these settings you will have a storage account, app service plan and app insights created for you as well as the Function app.

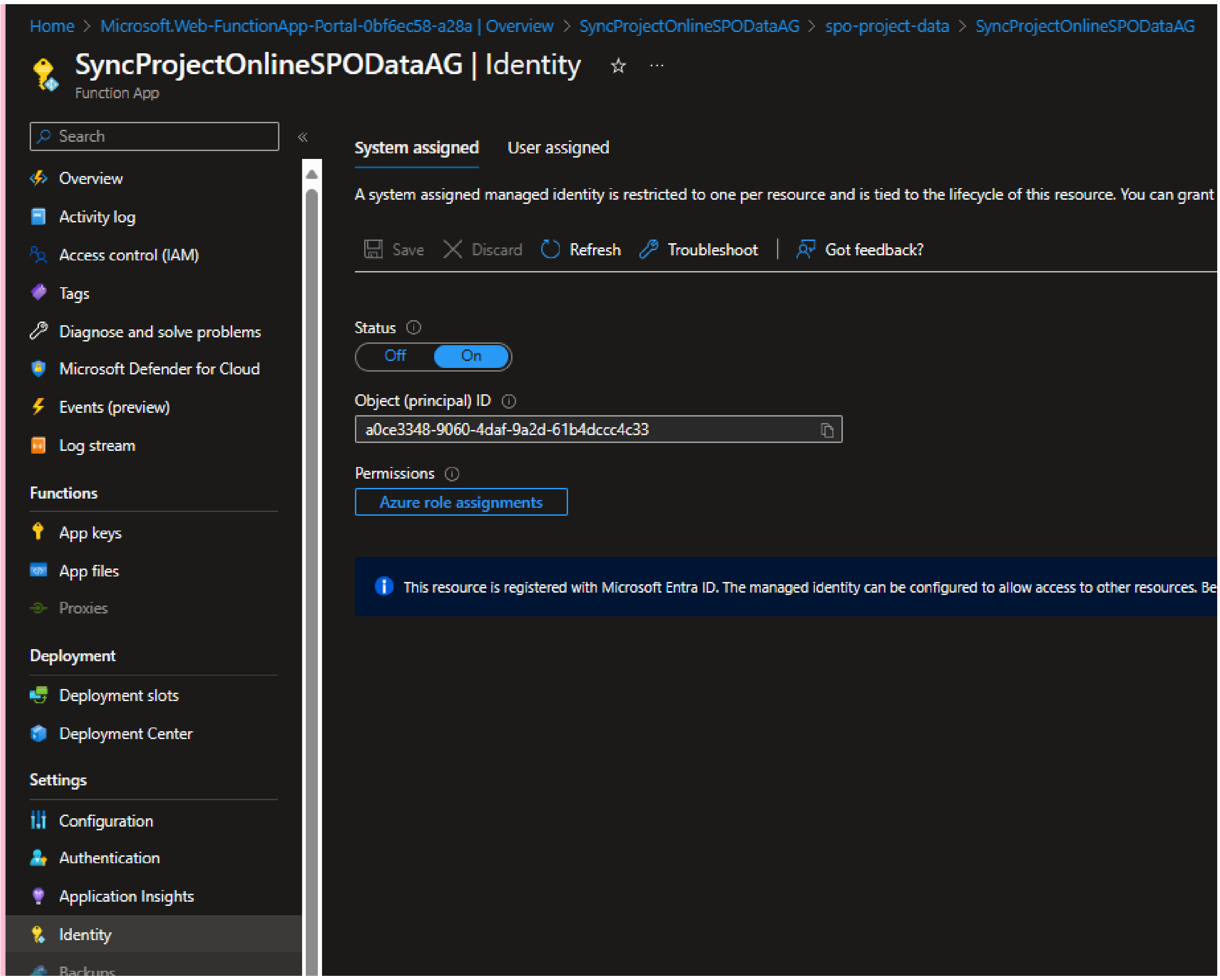
The resource group should contain similar to the below



Name	Type
ASP-spoprojectdata-b316	App Service plan
spoprojectdata9800	Storage account
SyncProjectOnlineSPODataAG	Function App
SyncProjectOnlineSPODataAG	Application Insights

Function Identity

We will give the Azure function an AAD assigned managed identity. With this identity, the function can access the KeyVault to obtain the refresh token without the need for an additional app registration



SyncProjectOnlineSPODataAG | Identity

Function App

A system assigned managed identity is restricted to one per resource and is tied to the lifecycle of this resource. You can grant

Status: On

Object (principal) ID: a0ce3348-9060-4daf-9a2d-61b4dccc4c33

Permissions

This resource is registered with Microsoft Entra ID. The managed identity can be configured to allow access to other resources. Be

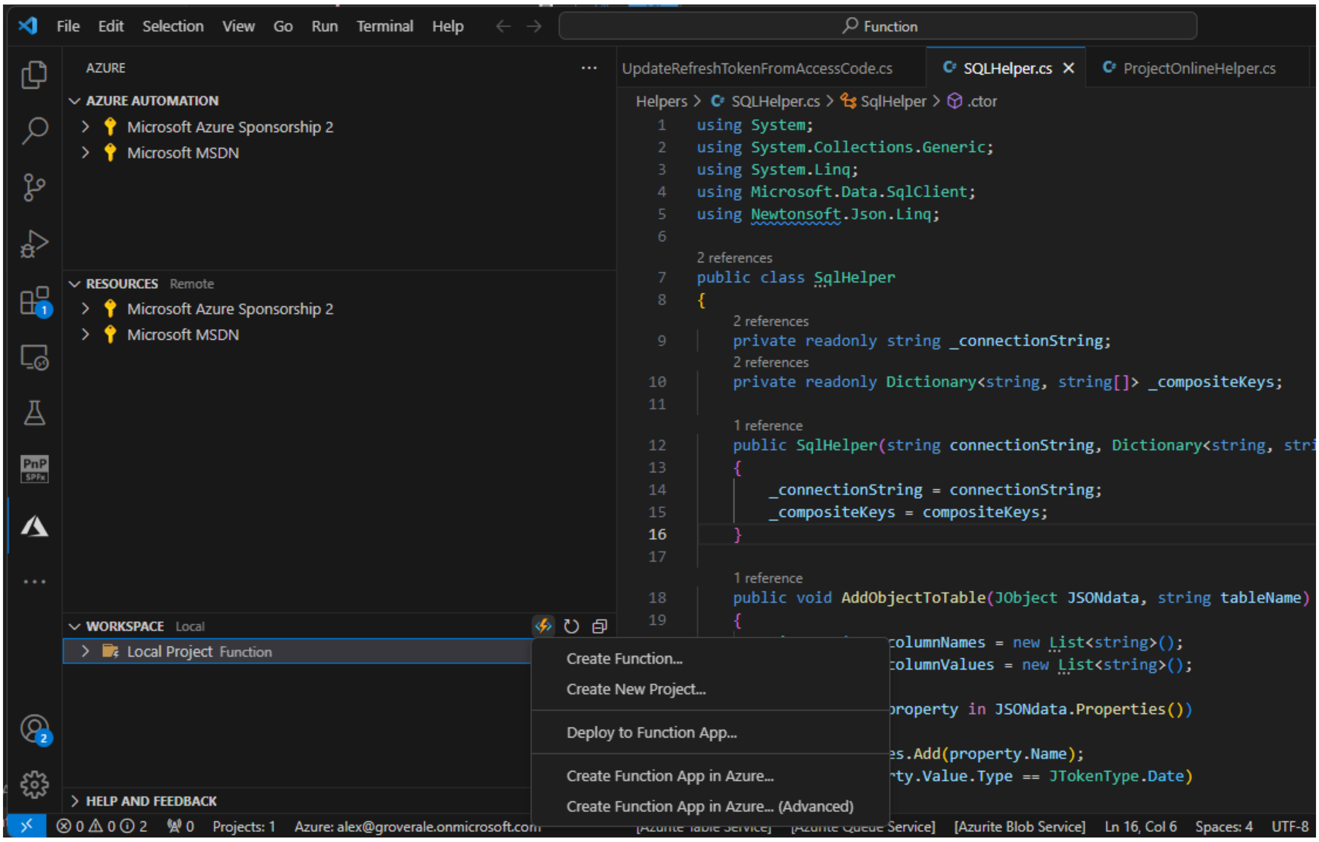
Function code deployment

The function code can be found here [groveale/project-online-api (github.com)](#)

It include two functions. One is configured to be executed daily. The other is a HTTP endpoint that should be called form a PowerShell script (also in the repo) to add a refresh token to the KeyVault

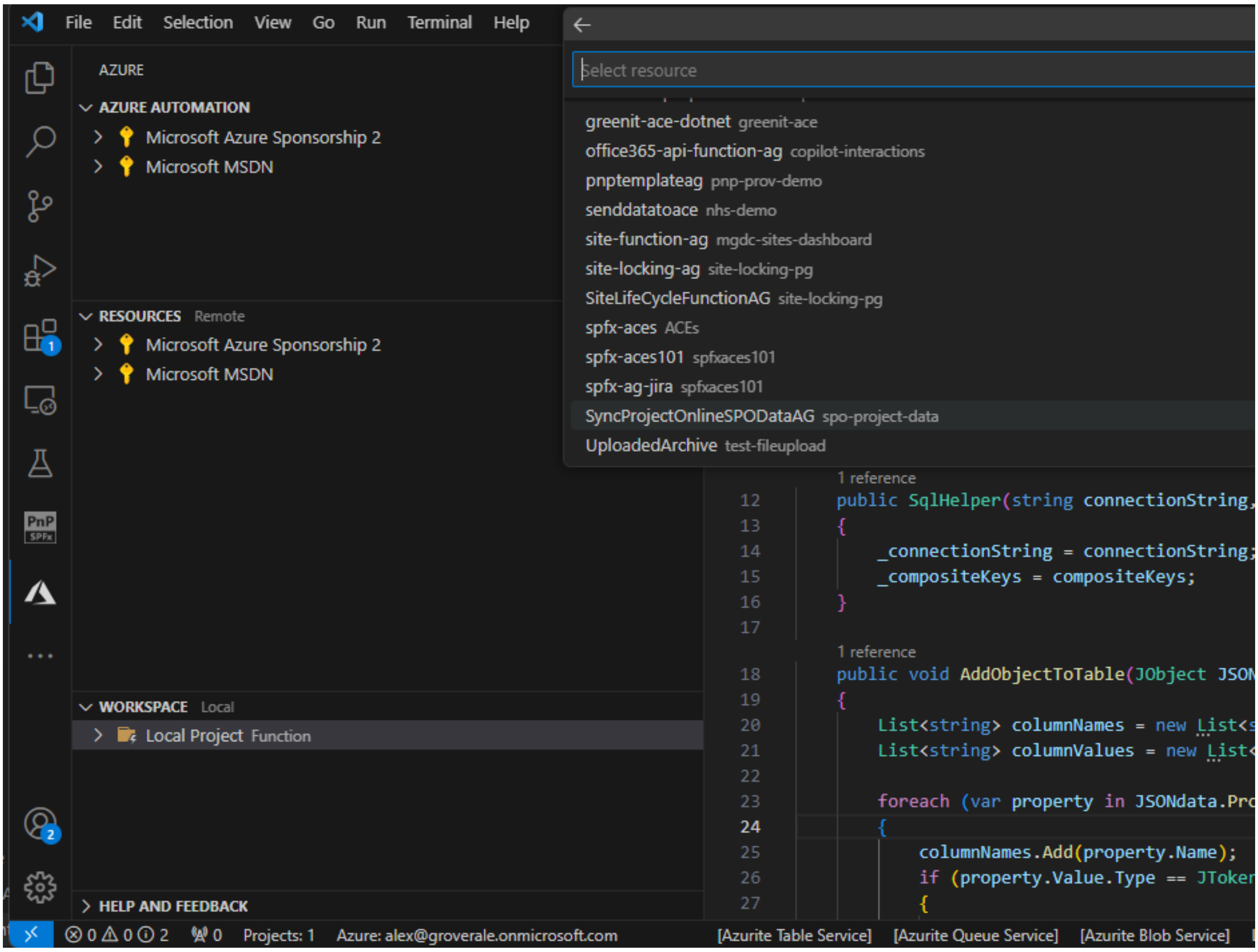
There are numerous ways to deploy function code to Azure. They are detailed here - [Deployment technologies in Azure Functions | Microsoft Learn](#)

An easy option is to use the Azure Function Extension in `vscode`. This requires the users to sign into their Microsoft account that has access to the provisioned Azure resources.



```
using System;
using System.Collections.Generic;
using System.Linq;
using Microsoft.Data.SqlClient;
using Newtonsoft.Json.Linq;

public class SqlHelper
{
    private readonly string _connectionString;
    private readonly Dictionary<string, string[]> _compositeKeys;

    public SqlHelper(string connectionString, Dictionary<string, stri
    {
        _connectionString = connectionString;
        _compositeKeys = compositeKeys;
    }

    public void AddObjectToTable(JObject JSONdata, string tableName)
    {
                columnNames = new List<string>();
                columnValues = new List<string>();

                roperty in JSONdata.Properties())

                es.Add(property.Name);
                rty.Value.Type == JTokenType.Date)
```

This opens a dialogue that enables you to select the Azure Function you have just created



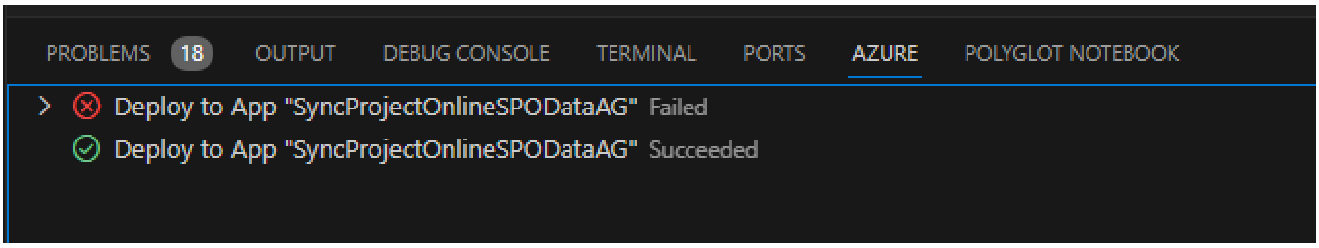
greenit-ace-dotnet greenit-ace
office365-api-function-ag copilot-interactions
pnptemplateag pnp-prov-demo
senddatatoace nhs-demo
site-function-ag mgdc-sites-dashboard
site-locking-ag site-locking-pg
SiteLifeCycleFunctionAG site-locking-pg
spfx-aces ACEs
spfx-aces101 spfxaces101
spfx-ag-jira spfxaces101
SyncProjectOnlineSPODataAG spo-project-data
UploadedArchive test-fileupload

```
1 reference
public SqlHelper(string connectionString,
{
    _connectionString = connectionString;
    _compositeKeys = compositeKeys;
}

1 reference
public void AddObjectToTable(JObject JSON
{
    List<string> columnNames = new List<s
    List<string> columnValues = new List<

    foreach (var property in JSONdata.Pro
    {
        columnNames.Add(property.Name);
        if (property.Value.Type == JToken
        {
```

⊗ 0 ⚠ 0 ⓘ 2 0 Projects: 1 Azure: alex@groverale.onmicrosoft.com [Azurite Table Service] [Azurite Queue Service] [Azurite Blob Service]

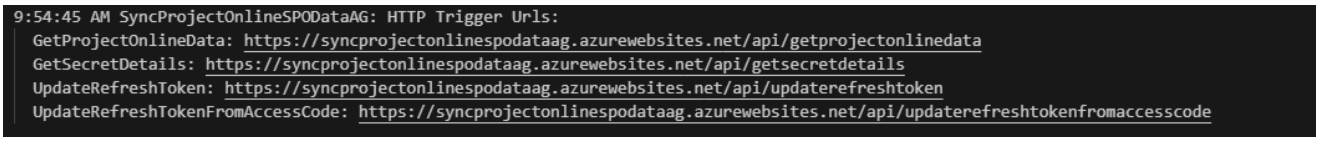
The `vscode` extension now builds and compiles the code and uploads it to the hosted Azure Function.

Check the Azure Tab in the terminal for deployment status. My first attempt failed, but my second was successful



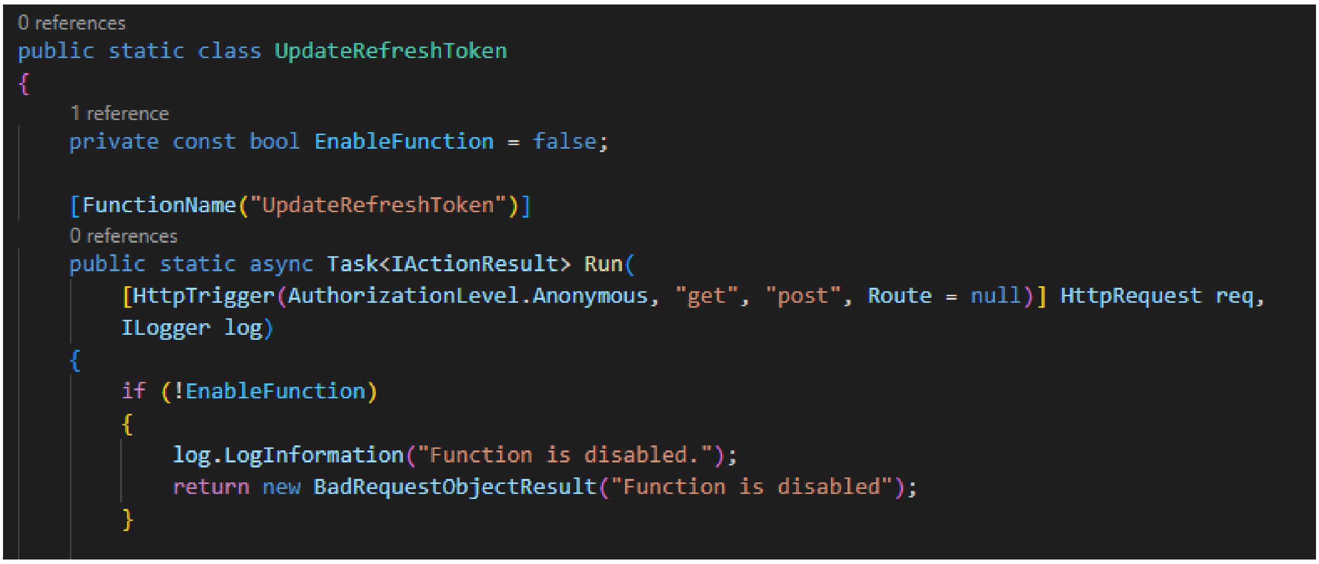
⊗ Deploy to App "SyncProjectOnlineSPODataAG" Failed
✓ Deploy to App "SyncProjectOnlineSPODataAG" Succeeded

For a successful deployment the output tab on the terminal will contain details of the URLs for the functions.



```
9:54:45 AM SyncProjectOnlineSPODataAG: HTTP Trigger Urls:
  GetProjectOnlineData: https://syncprojectonlinespodataag.azurewebsites.net/api/getprojectonlinedata
  GetSecretDetails: https://syncprojectonlinespodataag.azurewebsites.net/api/getsecretdetails
  UpdateRefreshToken: https://syncprojectonlinespodataag.azurewebsites.net/api/updaterefreshtoken
  UpdateRefreshTokenFromAccessCode: https://syncprojectonlinespodataag.azurewebsites.net/api/updaterefreshtokenfromaccesscode
```

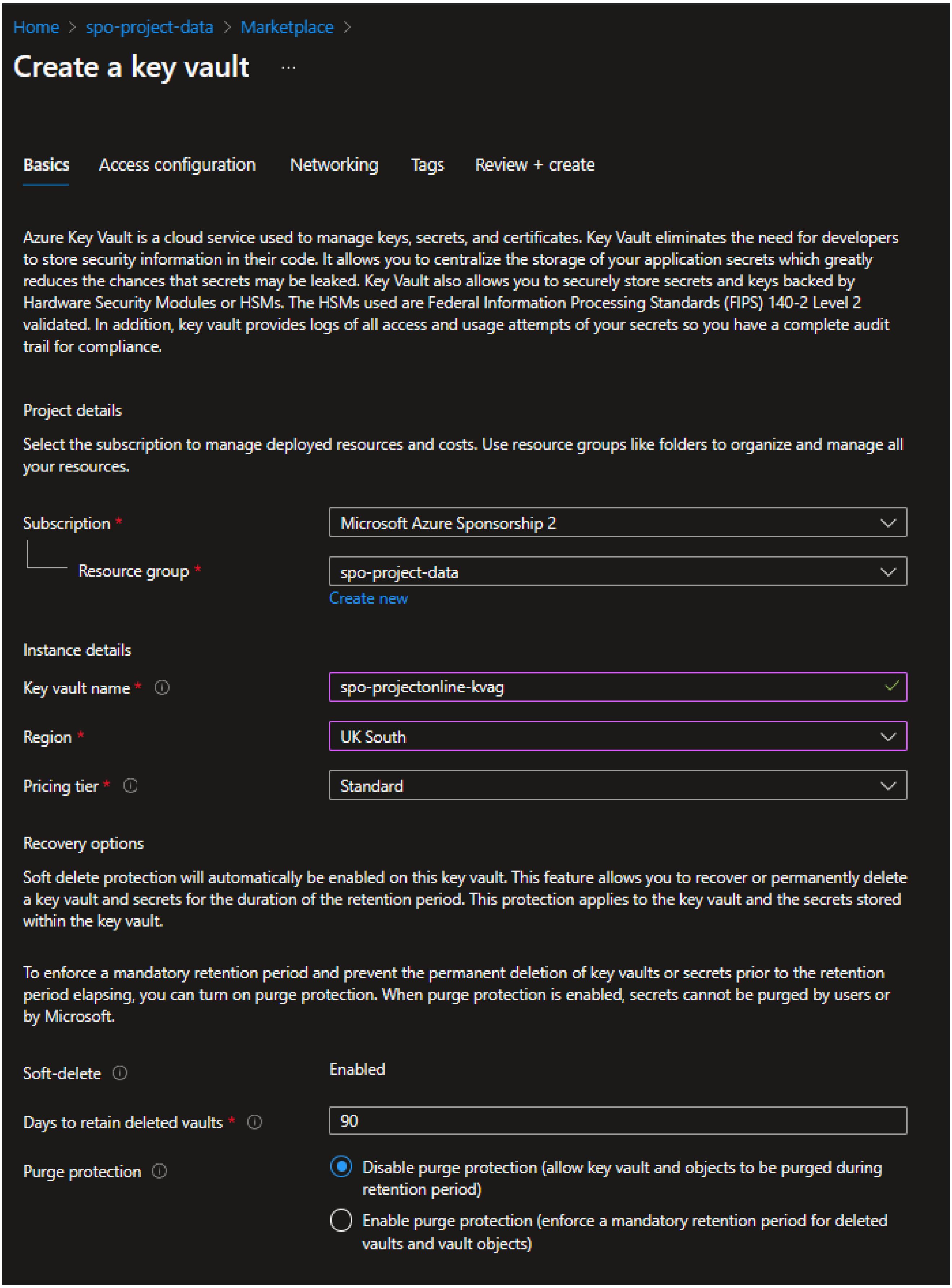
FWI the `GetSecretDetails` and `UpdateRefreshToken` functions are disabled. They are still in the repo for completeness and can be used as a reference resource



```
0 references
public static class UpdateRefreshToken
{
    1 reference
    private const bool EnableFunction = false;

    [FunctionName("UpdateRefreshToken")]
    0 references
    public static async Task<IActionResult> Run(
        [HttpTrigger(AuthorizationLevel.Anonymous, "get", "post", Route = null)] HttpRequest req,
        ILogger log)
    {
        if (!EnableFunction)
        {
            log.LogInformation("Function is disabled.");
            return new BadRequestObjectResult("Function is disabled");
        }
```

Azure Key Vault

Create a KeyVault in azure with the following config



Create a key vault ...

Azure Key Vault is a cloud service used to manage keys, secrets, and certificates. Key Vault eliminates the need for developers to store security information in their code. It allows you to centralize the storage of your application secrets which greatly reduces the chances that secrets may be leaked. Key Vault also allows you to securely store secrets and keys backed by Hardware Security Modules or HSMs. The HSMs used are Federal Information Processing Standards (FIPS) 140-2 Level 2 validated. In addition, key vault provides logs of all access and usage attempts of your secrets so you have a complete audit trail for compliance.

Project details

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * Microsoft Azure Sponsorship 2

Resource group * spo-project-data

Create new

Instance details

Key vault name * spo-projectonline-kvag ✓

Region * UK South

Pricing tier * Standard

Recovery options

Soft delete protection will automatically be enabled on this key vault. This feature allows you to recover or permanently delete a key vault and secrets for the duration of the retention period. This protection applies to the key vault and the secrets stored within the key vault.

To enforce a mandatory retention period and prevent the permanent deletion of key vaults or secrets prior to the retention period elapsing, you can turn on purge protection. When purge protection is enabled, secrets cannot be purged by users or by Microsoft.

Soft-delete Enabled

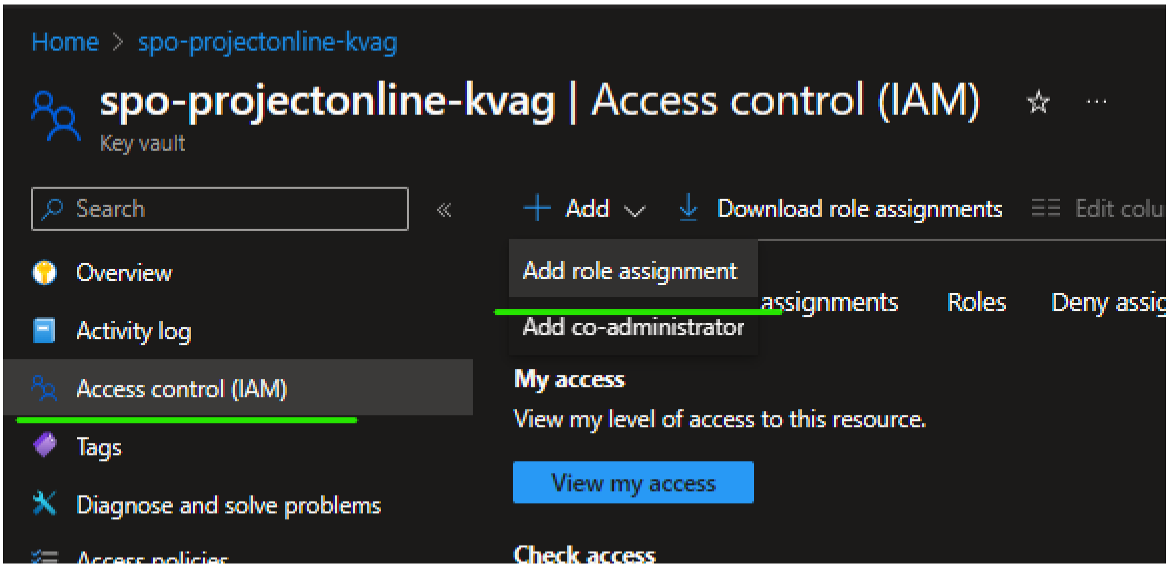
Days to retain deleted vaults * 90

Purge protection

● Disable purge protection (allow key vault and objects to be purged during retention period)

○ Enable purge protection (enforce a mandatory retention period for deleted vaults and vault objects)

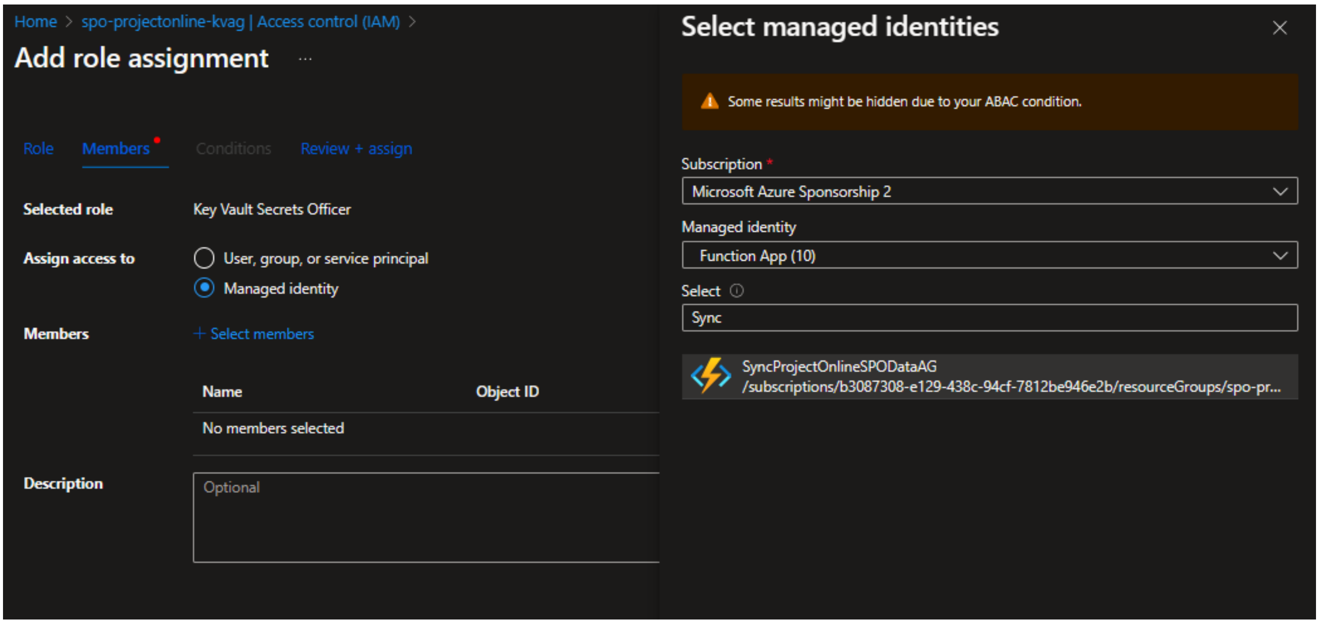
Once provisioned, go to the keyvault and add role based access for our Azure Function



My access

View my level of access to this resource.

Choose `Key Vault Secrets Officer` from the list of roles. The select the managed identity option. Click Select Members and find the Azure function. Add the role


Home > spo-projectonline-kvag | Access control (IAM) >
Add role assignment
Role
Members
Conditions
Review + assign
+ Select members
Select managed identities
×

Selected role Key Vault Secrets Officer

Assign access to
○ User, group, or service principal
● Managed identity

Members

Name	Object ID
No members selected	

Description: Optional

Some results might be hidden due to your ABAC condition.

Subscription *: Microsoft Azure Sponsorship 2

Managed identity: Function App (10)

Select: Sync

SyncProjectOnlineSPODataAG
/subscriptions/b3087308-e129-438c-94cf-7812be946e2b/resourceGroups/spo-pr...

The Azure function now has permission to create / update and retrieve secrets in the KV

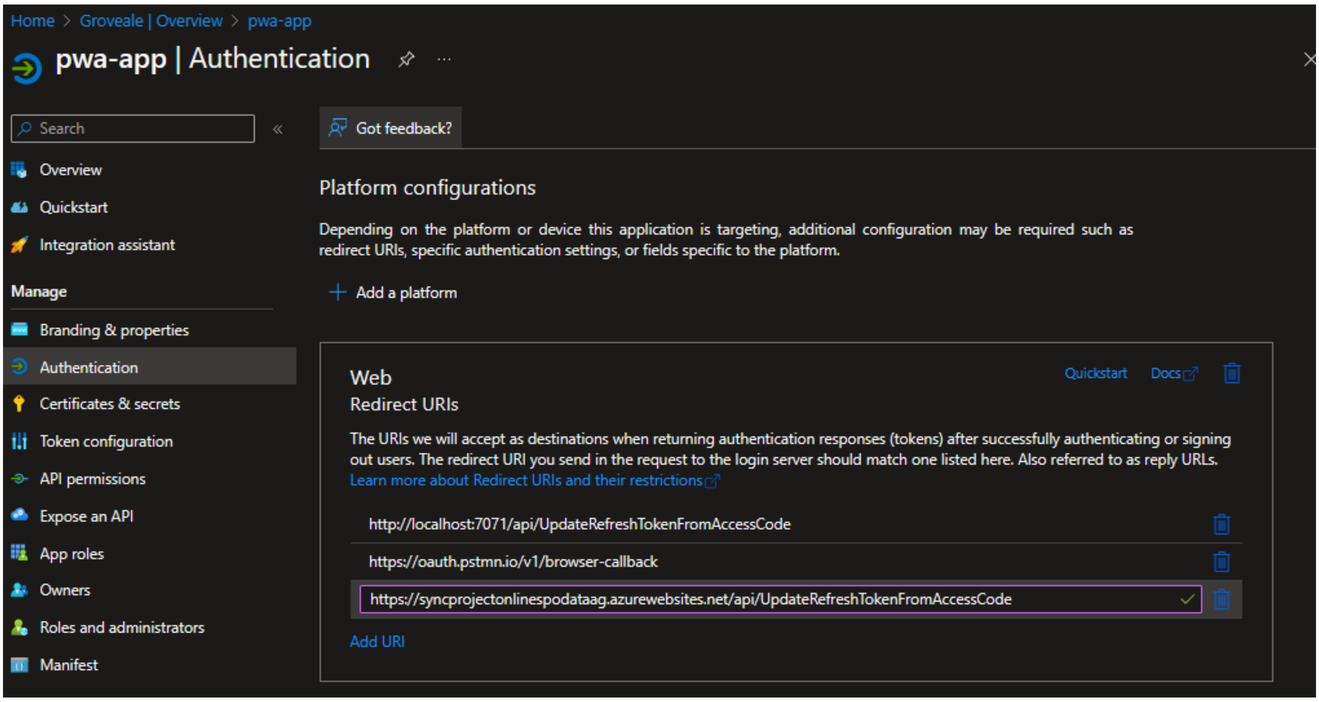
Configure Azure Components

App Registration

Now that they Azure function has been deployed we need to add another redirect URI

This is the URL of the `UpdateRefreshTokenFromAccessCode` function.

```
https://syncprojectonlinespodataag.azurewebsites.net/api/UpdateRefreshTokenFromAccessCode
```



Platform configurations

Depending on the platform or device this application is targeting, additional configuration may be required such as redirect URIs, specific authentication settings, or fields specific to the platform.

Web

Redirect URIs

The URIs we will accept as destinations when returning authentication responses (tokens) after successfully authenticating or signing out users. The redirect URI you send in the request to the login server should match one listed here. Also referred to as reply URLs. Learn more about Redirect URIs and their restrictions

http://localhost:7071/api/UpdateRefreshTokenFromAccessCode

https://oauth.pstmn.io/v1/browser-callback

https://syncprojectonlinespodataag.azurewebsites.net/api/UpdateRefreshTokenFromAccessCode

Function Config

The function app contains many config variables. These can either be entered into the portal or added via the `vscode` extension

```
{
        "clientId": "cd85557e-65a9-4854-b879-2671dfaee51a",
        "clientSecret": "eDs8Q~k7XHscraRjim...",
        "scope": "https://groverale.sharepoint.com/.default",
        "projectOnlineSiteUrl": "https://groverale.sharepoint.com/sites/pwa",
        "tenantId": "75e67881-b174-484b-9d30-c581c7ebc177",
        "fullPull": "false",
        "keyVaultName": "spo-projectonline-kvag",
        "redirectUri":
"https://syncprojectonlinespodataag.azurewebsites.net/api/UpdateRefreshTokenFromAccessCode",
        "sqlConnectionString": "Server=tcp:groveale-sql-
server.database.windows.net,143..;"
}
```

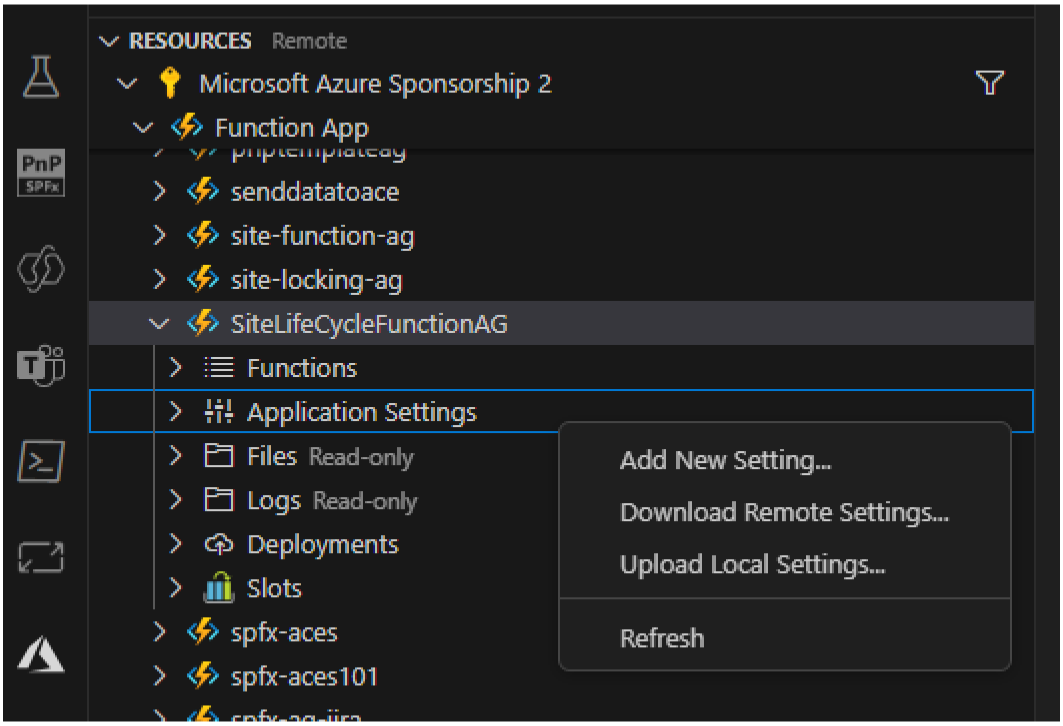
Field Name	Value
clientId	cd85557e-65a9-4854-b879-2671dfaee51a
clientSecret	eDs8Q~k7XHscraRjim...
scope	https://groverale.sharepoint.com/.default

Field Name	Value
projectOnlineSiteUrl	https://groverale.sharepoint.com/sites/pwa
tenantId	75e67881-b174-484b-9d30-c581c7ebc177
fullPull	false
keyVaultName	spo-projectonline-kvag
redirectUri	https://syncprojectonlinespodataag.azurewebsites.net/api/UpdateRefresh
sqlConnectionString	Server=tcp:groveale-sql-server.database.windows.net,143..;

Open up the extension, find the function in your list or resource. Expand and right click the app settings. Clicking upload local settings will upload the settings values from the `local.settings.json` files

Note

The repo includes a `local.settings.json.sample` file. Use this to create a local settings file for your environment



The Azure function is now configured.

Key Vault Initial Config

There is a once time action required to seed the KeyVault secret with a refresh token. This token will be used by the function when it attempts to pull the Project Online data.

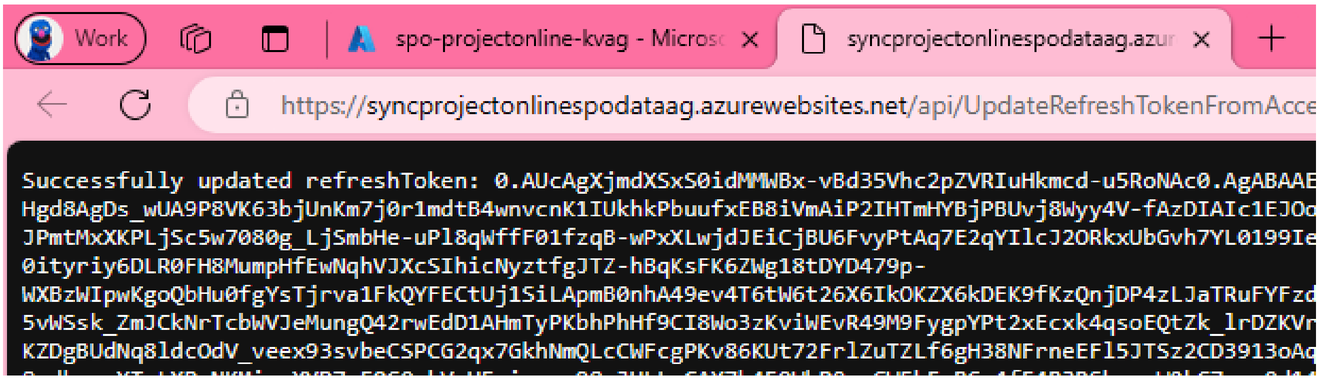
A PowerShell script `LoginAndPostTokenToFunction` is included in the repo to support this activity.

There are three variable that should be updated in this script before executing

```
$client_id = "cd85557e-65a9-4854-b879-2671dfaee51a"
$tenantId = "75e67881-b174-484b-9d30-c581c7ebc177"
$redirect_uri =
"https://syncprojectonlinespodataag.azurewebsites.net/api/UpdateRefreshTokenFromAccessCode"
```

Executing this script will open a browser where you should login with the user that has access to the project online data.

A successful login attempt will add the refresh token to the KeyVault and return the token to the browser


Work
spo-projectonline-kvag - Microsc
syncprojectonlinespodataag.azur
https://syncprojectonlinespodataag.azurewebsites.net/api/UpdateRefreshTokenFromAcce

```
Successfully updated refreshToken: 0.AUcAgXjmdXSxS0idMMWBx-vBd35Vhc2pZVRIuHkmcd-u5RoNAc0.AgABAAE
Hgd8AgDs_wUA9P8VK63bjUnKm7j0r1mdtB4wnvcnK1IUkhkPbuufxEB8iVmAiP2IHTmHYBjPBUvj8Wyy4V-fAzDIAIc1EJOo
JPmtMxXKPLjSc5w7080g_LjSmbHe-uPl8qWffF01fzqB-wPxXLwjdJEiCjBU6FvyPtAq7E2qYIlcJ2ORkxUbGvh7YL0199Ie
0ityriy6DLR0FH8MumpHfEwNqhVJXcSIhicNyztfgJTZ-hBqKsFK6ZWg18tDYD479p-
WXBzWIpwKgoQbHu0fgYsTjrva1FkQYFECtUj1SiLApmB0nhA49ev4T6tW6t26X6IkOKZX6kDEK9fKzQnjDP4zLJaTRuFYFzd
5vWSsk_ZmJCkNrTcbWVJeMungQ42rwEdD1AHmTyPKbhPhHf9CI8Wo3zKviWEvR49M9FygpYPt2xEcxk4qsoEQtZk_lrDZKVr
KZDgBUdNq8ldcOdV_veex93svbeCSPCG2qx7GkhNmQLcCWFcgPKv86KUt72FrlZuTZLf6gH38NFrneEFl5JTSz2CD3913oAq
```

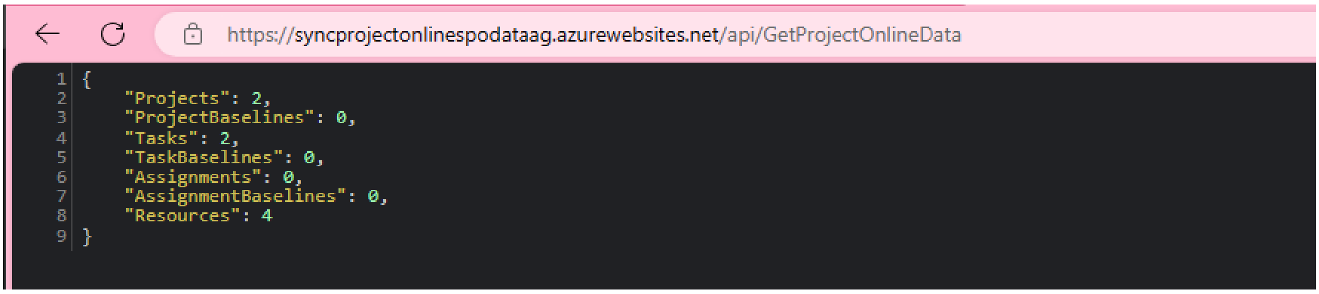
SQL db config

There is a number of SQL create table scripts included in the repo. Please run these to create tables in your prestaging database

Test the Solution

Everything should now be in place for you to test the solution

Simplest way is to enter the URL of the `GetProjectOnlineData` function into your browser


https://syncprojectonlinespodataag.azurewebsites.net/api/GetProjectOnlineData

```
{
    "Projects": 2,
    "ProjectBaselines": 0,
    "Tasks": 2,
    "TaskBaselines": 0,
    "Assignments": 0,
    "AssignmentBaselines": 0,
    "Resources": 4
}
```

This indicates that 2 project items, 2 task items and 4 resource items have been updated in SQL.

Production Considerations

The `GetProjectOnlineData` function should be configured as a TimerTriggered function. This way there would be to HTTP endpoint exposed to trigger a pull of the data.

The `UpdateRefreshTokenFromAccessCode` should be configured to not allow anonymous access. At this stage anyone with the URL could attempt to use it to update the secret in the Key Vault. they would not be able to obtain the refreshToken value but a successful logon attempt would overwrite the key vault value and stop the solution from working