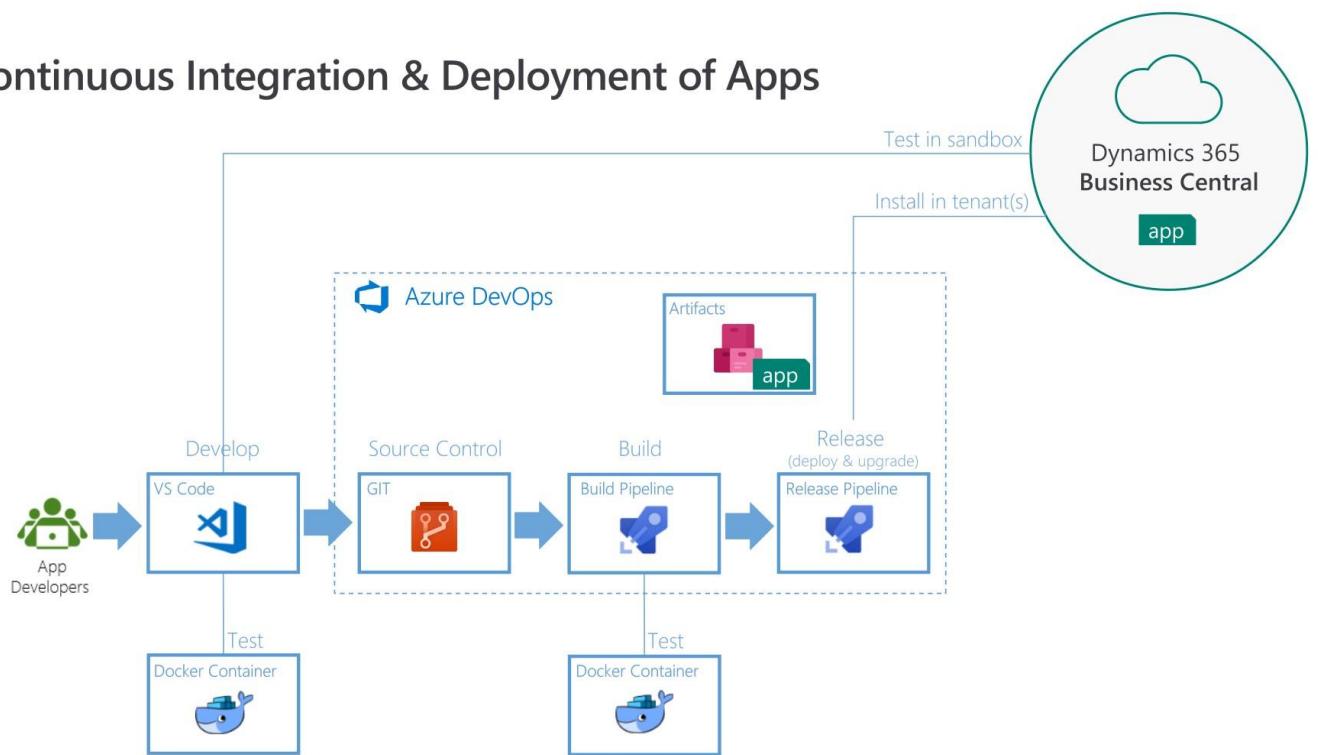


Azure DevOps – CI/CD Workshop

Continuous Integration & Deployment of Apps



This workshop will help you setup a project on Azure DevOps, including Continuous Integration, Continuous Deployment and requirements for the developer to submit Pull Requests when checking in.

Workshop repositories

This workshop uses two repositories. The scripts and pipelines are very similar, but one is a Per Tenant Extension and using the Per Tenant Extension cop + number ranges and the other one is an AppSource app, using the AppSourceCop + appsource number ranges + prefixes and breaking change notifications.

<https://dev.azure.com/businesscentralapps/Old.HelloWorld>

or

<https://github.com/businesscentralapps/Old.HelloWorld>

This is the Per Tenant Extension version of the app.

- Number range defined in app.json is 50100 to 50149.
- PerTenantExtensionCop and UICop are enabled during build (in scripts\settings.json)
- PreviousApps points to .zip file containing previous versions of apps

<https://dev.azure.com/businesscentralapps/Old.HelloWorld.AppSource>

or

<https://github.com/businesscentralapps/Old.HelloWorld.AppSource>

This is the AppSource version of the app.

- Number range defined in app.json is 70074169 to 70074218 (my allocated number range)
- Logo and various URL's in app.json set
- TranslationFile feature enabled
- AppSourceCop and UICop are enabled during build (in scripts\settings.json)
- PreviousApps points to .zip file containing previous versions of apps

The per tenant extension version is the easiest to use for the workshop.

Workshop environment

To complete this workshop, you need a computer with at least 16Gb of RAM running Windows 10 or Windows Server 2019 with the latest Windows updates applied, the latest Visual Studio Code update and the latest Docker version installed and running Windows Containers.

If your own computer doesn't meet these requirements, you can create a virtual machine on Azure using <http://aka.ms/getbc>. After logging into your Azure subscription, you should **at least** specify valid values for these properties:

- Resource group: **<name of resource group>**
- Vm Name: **<name of VM>**
- Accept Eula: **Yes**
- Remote Desktop Access: * **(to allow all IP addresses to connect to remote desktop)**
- Admin Password: **<my admin password>**
- Artifact Url: **bcartifacts/sandbox//us/latest**
- License File Uri: **<secure url to a developer/training licensefile>**
- Final Setup Script Url: **additional-install.ps1 (or full url to script)**
- Contact E Mail for Let's Encrypt: **<my email address>**
- Add Traefik: **Yes (if you want to be able to access containers in the VM from the internet)**

Read this blog post <https://freddysblog.com/2017/02/26/create-a-secure-url-to-a-file/> to learn more about how to create a secure url.

Note the Final Setup Script Url, which will invoke the script [additional-install.ps1](#) script after the VM is final, which will install chocolatey and use that to install Git, Microsoft Edge, Google Chrome, Firefox and some VS Code extensions.

The remaining properties can be left to their default values. Accept the terms and conditions and press **Purchase**.

Now, the deployment starts. It will take around ~30 minutes until the VM is ready. You can monitor the deployment status on the landing page (<http://<vmname>.<region>.cloudapp.azure.com>). The Landing page should state: Installation Complete before you continue.

Prerequisites

Software

If you are using a workshop environment created by <https://aka.ms/getbc>, you will have this software installed already automatically. If you are using your own computer/laptop, you need to check/install this software:

- BcContainerHelper PowerShell module
- Az PowerShell module
- VS Code
- AL Language Extension
- Git
- Microsoft Edge or Google Chrome

Get an Azure DevOps Account

The Azure DevOps account and organization is where you will create your projects and store your source code. Open <https://azure.microsoft.com/en-us/services/devops/> to create a free account. You will be able to create public or private projects in Azure DevOps.

Install and configure GIT

Git is the source code management tool used by Visual Studio Code to connect to your Azure DevOps repository.

If GIT isn't already installed, navigate to <https://www.git-scm.com/download/win> and click the download link to download and install Git. Select Visual Studio Code as Git's default editor during installation Wizard and select to commit as-is and checkout as-is when asked.

Configure your username and email in git by starting a **Command Prompt** and type:

```
git config --global user.name "<your username>"  
git config --global user.email "<your email>"  
git config --global credential.helper manager
```

You should use the same email here, as the one used for your Azure DevOps Account.

Create a Key vault for your secrets

Secrets belong in key vaults and since this workshop will contain license file secrets, passwords and a shared access signature for insider builds, the key vault is needed.

Navigate to <https://portal.azure.com> and login to your subscription. Click Create a resource and select Key vault. Fill out the values as needed and create the Key Vault.

Navigate to the Key Vault resource and navigate to secrets:

Click Generate/Import and fill out the values:

And press create.

For this workshop you need to create a licensefile secret with a secure url to your license file and a password secret.

In PowerShell, accessing these secrets are very simple using the Az PowerShell module.

First, you need to connect your Windows User to your Azure Account, run:

[Connect-AzAccount](#)

login to your Azure Account and now you can use:

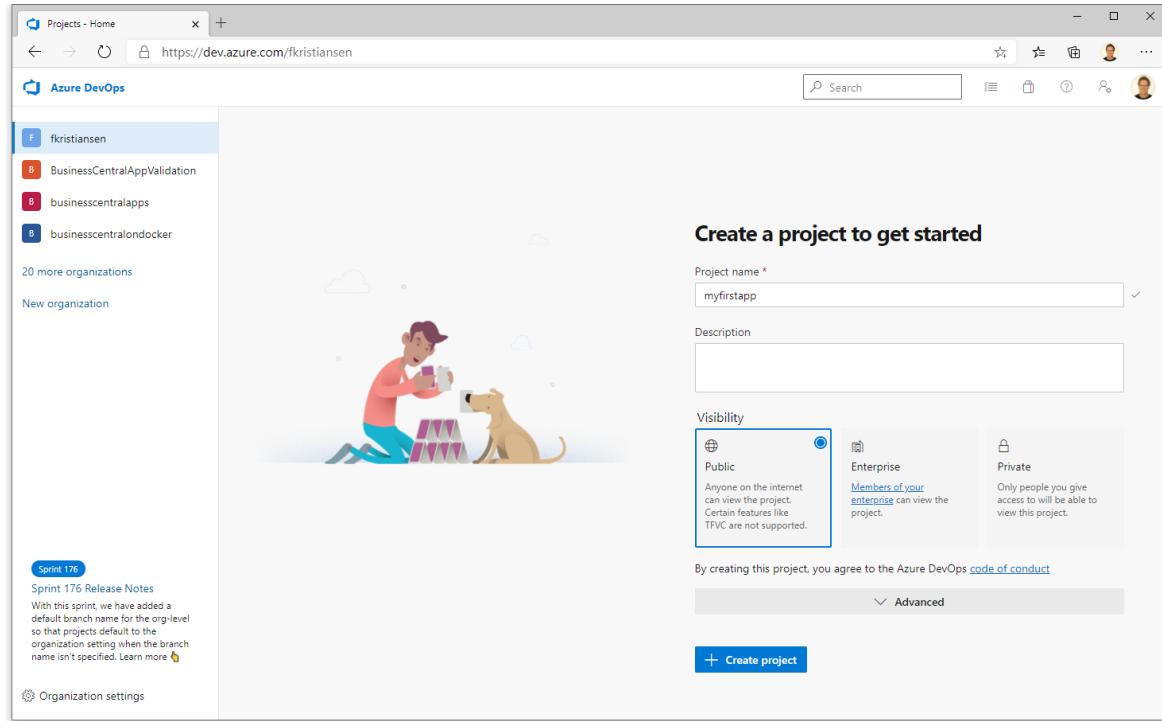
```
$vaultName = "BuildVariables"
$passwordSecret = Get-AzKeyVaultSecret -VaultName $vaultName -Name "Password"
$password = $passwordSecret.SecretValue
```

To read the password as a `SecureString`. If you want to see the actual password, you need to convert the password to text, which can be done using:

```
[Runtime.InteropServices.Marshal]::PtrToStringAuto([Runtime.InteropServices.Marshal]::SecureStringToBSTR($Password))
```

Create your organization and your first project

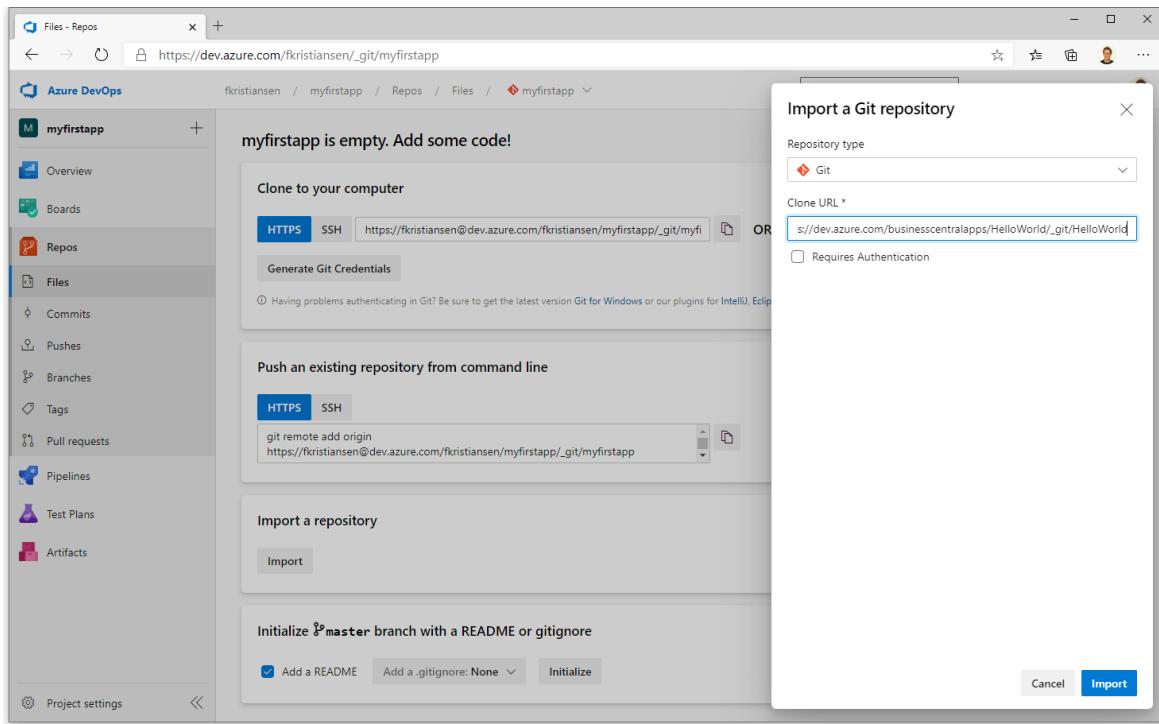
Navigate to <https://devops.azure.com> and login to your DevOps account. Create your organization, which is the location in which you will create your projects. In your organization, create your first project:



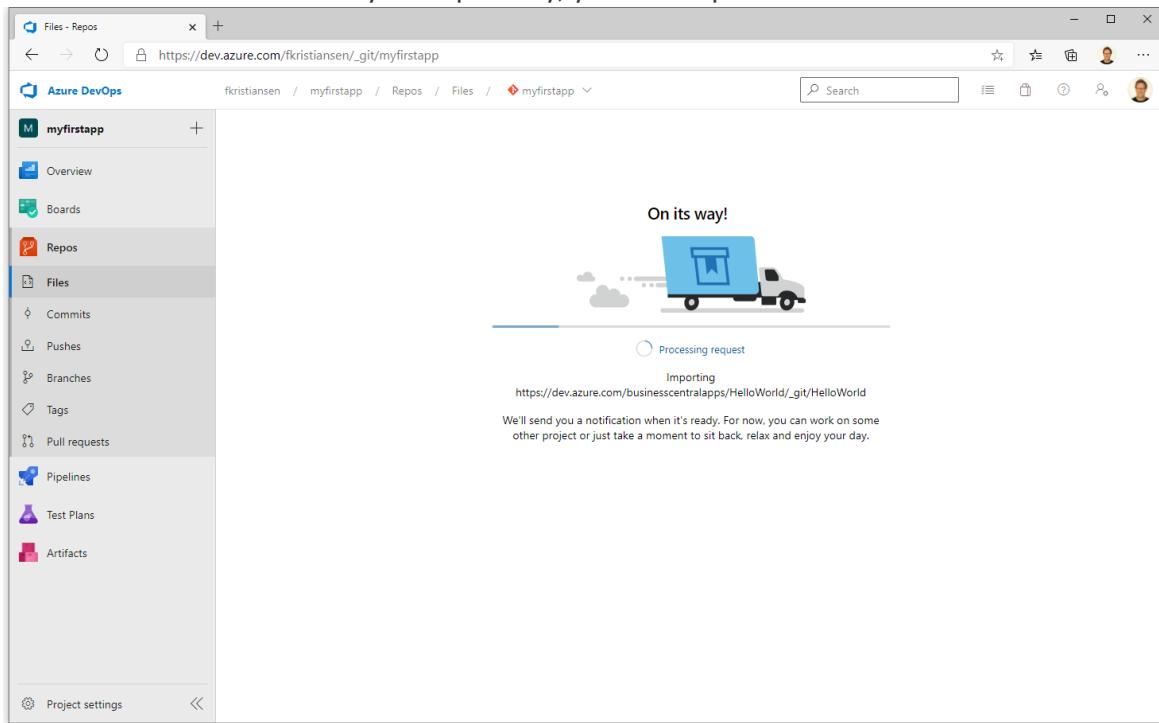
In the project navigate to the **Repos -> Files** area, click **Import** and enter

https://dev.azure.com/businesscentralapps/Old.HelloWorld/_git/Old.HelloWorld (or

Old.HelloWorld.AppSource for the AppSource version) in the Clone URL field. The sample repository is also available on github here: <https://github.com/ BusinessCentralApps/ Old.HelloWorld> (add .AppSource for AppSource)



After the truck has delivered your repository, you can inspect the content.



Inspect the content of the repository

The repository consists of 4 project folders: **base**, **app**, **test** a **scripts** folder.

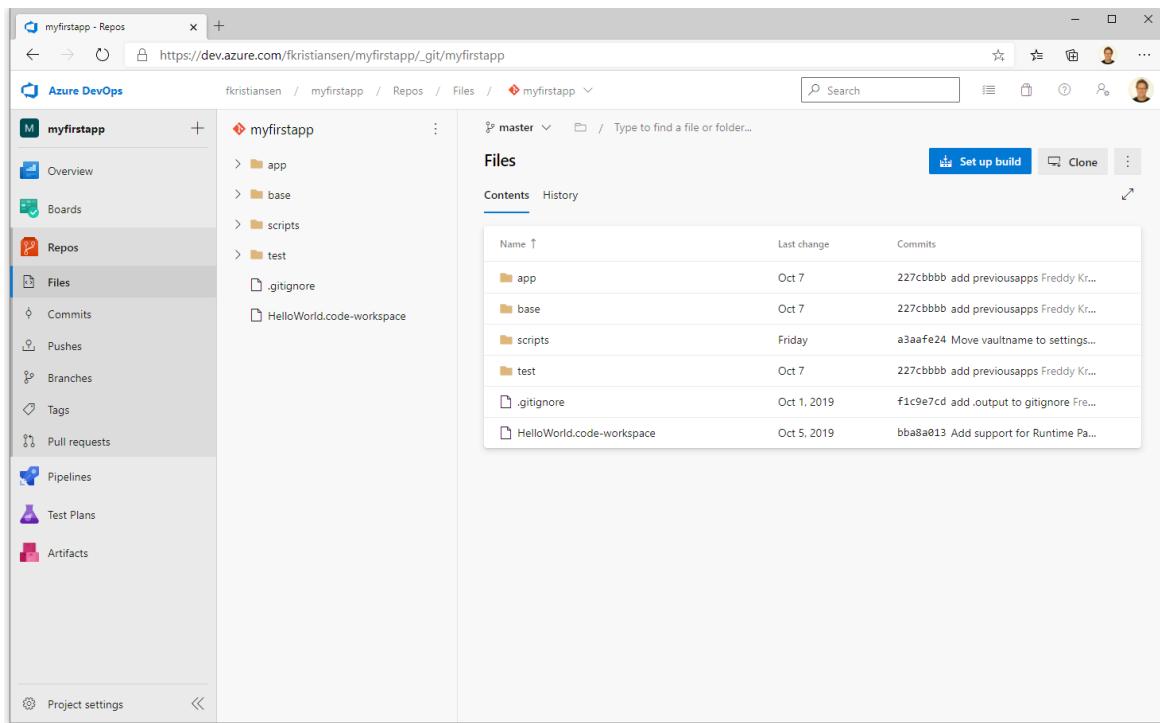
base contains a single codeunit with a single function, which returns **App Published: Hello World Base!**

app has a dependency to **base** and consists of a **Customer List Page Extension**, which will pop up the Hello World message on the **OnOpenPage Trigger**.

test is the test app with a dependency to **app**, containing a **single test**, which opens the **Customer List** and tests that the **Hello World message appears**. **scripts** is a set of scripts/files used for **CI/CD** and setup of dev environments.

The **.gitignore** file is known to everybody who are using GIT as a description of which files GIT should ignore.

The **HelloWorld.code-workspace** is the workspace you want to open with VS Code.



The screenshot shows the Azure DevOps interface for a repository named 'myfirstapp'. The left sidebar is the 'Azure DevOps' navigation bar with various options like Overview, Boards, Repos, Files, Commits, Pushes, Branches, Tags, Pull requests, Pipelines, Test Plans, and Artifacts. The 'Files' option is selected. The main content area shows the 'myfirstapp' folder structure with subfolders 'app', 'base', 'scripts', 'test', and files '.gitignore' and 'HelloWorld.code-workspace'. The 'HelloWorld.code-workspace' file is highlighted. The 'Contents' tab is selected, showing a table with columns 'Name', 'Last change', and 'Commits'. The table data is as follows:

Name	Last change	Commits
app	Oct 7	227cbbbb add previousapps Freddy Kr...
base	Oct 7	227cbbbb add previousapps Freddy Kr...
scripts	Friday	a3aafe24 Move vaultname to settings...
test	Oct 7	227cbbbb add previousapps Freddy Kr...
.gitignore	Oct 1, 2019	f1c9e7cd add .output to gitignore Fre...
HelloWorld.code-workspace	Oct 5, 2019	bba8a013 Add support for Runtime Pa...

Note: The template will constantly be changed/improved, and the content of the template repository might vary.

If you want to add multiple apps to the project, the idea is to create folders for each app in the root folder.

Clone the project

In order to work with the project, we need to clone the project to our work machine. You can use the Workshop VM as work machine, or you can use your personal computer/laptop.

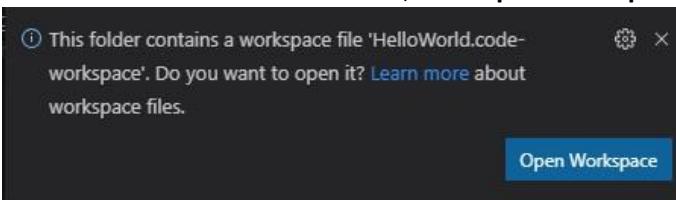


The screenshot shows the 'Clone repository' dialog. It has two main sections: 'Clone Git repository using command line or IDE' and 'IDE'. Under 'Clone Git repository using command line or IDE', there are 'Command line' and 'Generate Git credentials' buttons. Under 'Command line', 'HTTPS' is selected. The URL 'https://freddykristiansen@dev.azure.com/freddykri...' is shown. Under 'IDE', 'Clone in VS Code' is selected. A note at the bottom says: 'Having problems authenticating in Git? Be sure to get the latest version of [Git for Windows](#) or our plugins for [IntelliJ](#), [Eclipse](#), [Android Studio](#) or [Windows command line](#)'.

Make sure VS Code is running and click the **Clone** button in the upper right Corner and select **Clone in VS Code**.

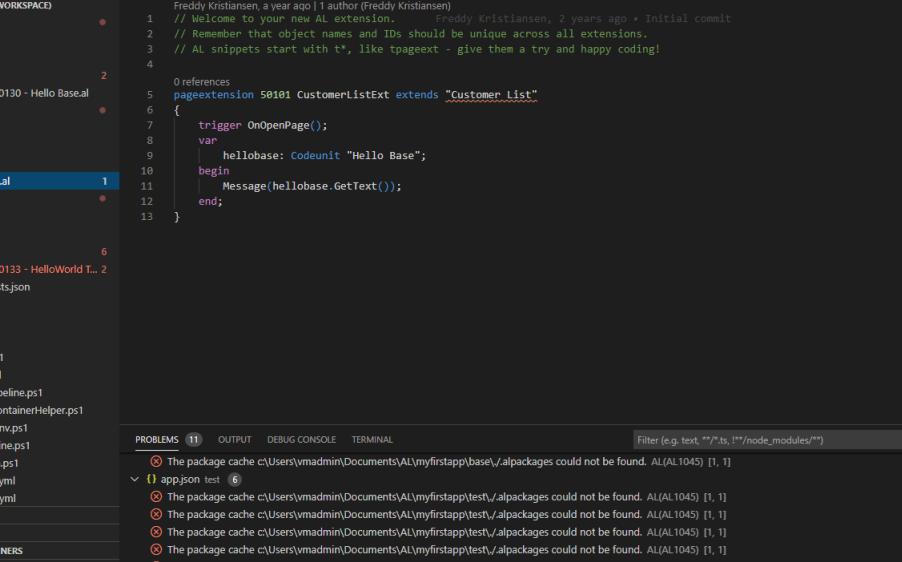
Allow the browser to Open VS Code and select a location (f.ex. `Documents\AL`) for the repository and sign-in to your Azure DevOps account if asked to do so. Say Yes to open the repository.

After opening the repository, VS Code asks whether you want to open the workspace file, click **Open Workspace**.



The screenshot shows a dialog box in VS Code. The text inside the box says: 'This folder contains a workspace file 'HelloWorld.code-workspace'. Do you want to open it? [Learn more about workspace files.](#)' At the bottom of the dialog is a blue button labeled 'Open Workspace'.

And you should “almost” be ready to start working:



The screenshot shows the Visual Studio Code interface with the following details:

- File Explorer:** Shows the project structure under "Helloworld (Workspace)". The "HelloWorld.al" file is selected and has a status bar indicating 1 error.
- Editor:** Displays the content of the "HelloWorld.al" file. The code defines a page extension "CustomerListExt" that triggers on "OnOpenPage". It uses the "HelloBase" codeunit to get text.
- Problems Panel:** Shows 11 errors, all of which are AL1045: "The package cache c:\Users\vmadmin\Documents\AL\myfirstapp\base\.\alpackages could not be found." These errors are associated with the "app.json" and "HelloWorld.al" files.
- Terminal:** Shows the command "HelloWorld.al" was run.
- Status Bar:** Shows the file was created by "Freddy Kristiansen, a year ago" and has 2 changes.

Make it your project

The project has been setup with some default **object ids**, **app ids**, **publisher** and **name**. Since this workshop is going to deploy your app to a cloud tenant, you should change these.

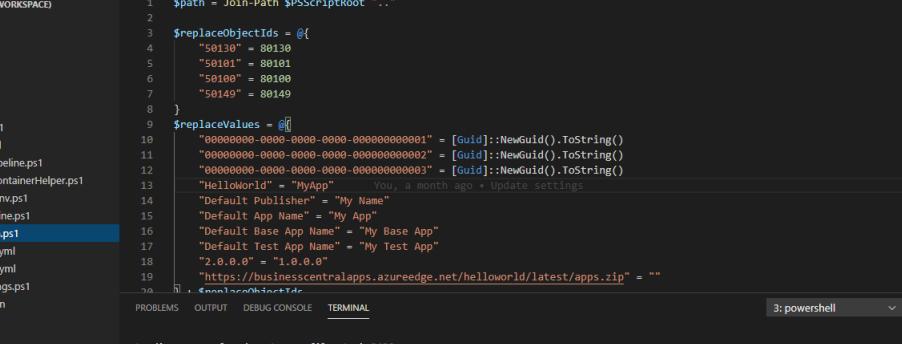
You can either fix the solution manually or you can run a small script to fix the solution automatically.

Fix the solution manually

You can modify `app.json` in the base project, the app project and the test project manually, setting the app id, app name, app publisher and app versions. Remember that the app project has a dependency on the base project and the test project has a dependency on the app project. Your ids must match. If you are using the AppSource version of `HelloWorld`, you should also modify the object ids.

Fix the solution automatically

In VS Code, open the **scripts** folder and open the **MySolution.ps1** file.



The screenshot shows the Visual Studio Code interface with the following details:

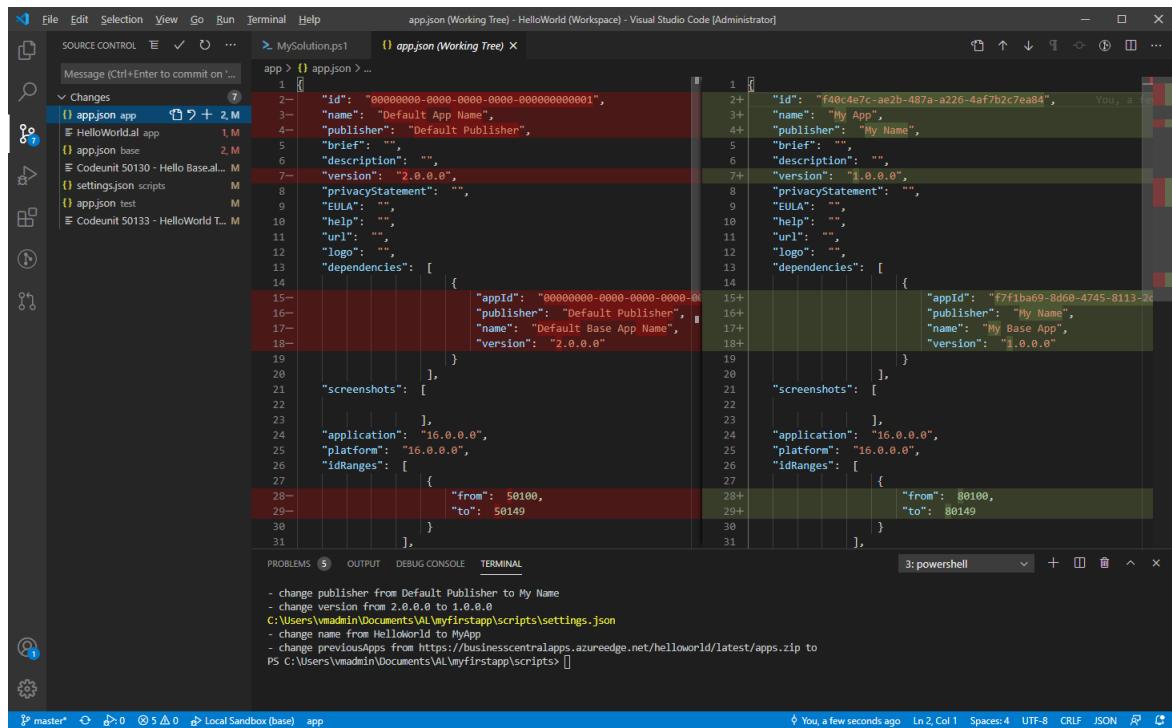
- File Explorer:** Shows the workspace structure. The `scripts` folder contains the `MySolution.ps1` script, which is currently selected.
- Terminal:** The title bar indicates the terminal is running PowerShell. The command `.\MySolution.ps1` has been run, and the output shows the creation of a new Azure App Service named "MyApp" with a GUID-based ID.
- Code Editor:** The `MySolution.ps1` script is displayed. It uses PowerShell's `Join-Path` cmdlet to set the `$path` variable to the parent directory of the script. It then defines a hashtable `$replicaObjectIds` mapping replica IDs to their corresponding GUIDs. The script then creates a new Azure App Service using the `Set-AzWebApp` cmdlet with the `-Name` parameter set to "MyApp".

```
MySolution.ps1 -> PowerShell -> .\MySolution.ps1
```

```
1 $path = Join-Path $PSScriptRoot ".."
2
3 $replicaObjectIds = @{
4     "50130" = 80130
5     "50101" = 80101
6     "50100" = 80100
7     "50149" = 80149
8 }
9 $replicaValues = @{
10     "00000000-0000-0000-0000-000000000001" = [Guid]::NewGuid().ToString()
11     "00000000-0000-0000-0000-000000000002" = [Guid]::NewGuid().ToString()
12     "00000000-0000-0000-0000-000000000003" = [Guid]::NewGuid().ToString()
13     "HelloWorld" = "MyApp"
14     "Default Publisher" = "My Name"
15     "Default App Name" = "My App"
16     "Default Base App Name" = "My Base App"
17     "Default Test App Name" = "My Test App"
18     "2.0.0.0" = "1.0.0.0"
19     "https://businesscentralapps.azureedge.net/helloworld/latest/apps.zip" = ""
20 }
```

Modify **\$replaceValues** array if needed to indicate the values you want to use on the right side, save the script and run the script. **Note:** Running the script can be tricky, I cannot get F5 to work..., so right-click the PowerShell file and **Open in Integrated Terminal** – or use **Ctrl+Shift+P** and **Open Current File in PowerShell ISE**.

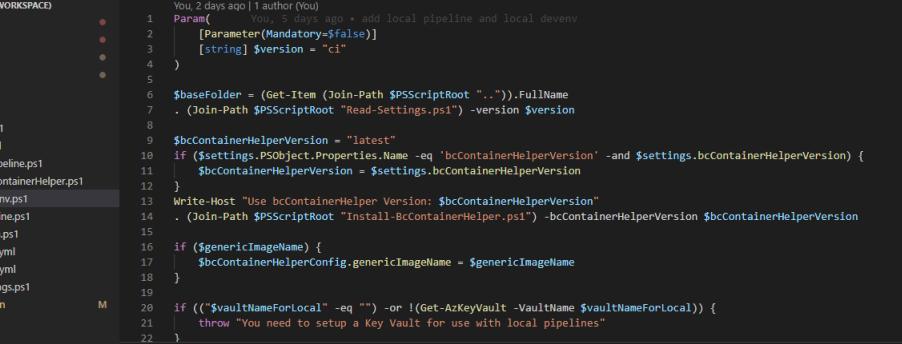
You should see an output indicating which files are modified and if you click the source control symbol, you should see which files have changed and by clicking a file, you can see the changes.



```
MySolution.ps1  app.json (Working Tree) - HelloWorld (Workspace) - Visual Studio Code [Administrator]
File Edit Selection View Go Run Terminal Help
Message (Ctrl+Enter to commit on ...)

Changes
app.json app  2, M
HelloWorld.al app  1, M
app.json base  2, M
Codeunit 50130 - Hello Base.al... M
settings.json scripts  M
app.json test  M
Codeunit 50133 - HelloWorld T... M

1  [ 1
2- "id": "0000000-0000-0000-000000000001", 2+ "id": "f40c4e7c-ae2b-487a-a226-4af7b2c7ea84",
3- "name": "Default App Name", 3+ "name": "My App",
4- "publisher": "Default Publisher", 4+ "publisher": "My Name",
5- "brief": "", 5- "description": "",
6- "description": "", 6- "version": "2.0.0.0",
7- "version": "2.0.0.0", 7+ "version": "1.0.0.0",
8- "privacyStatement": "", 8- "privacyStatement": "",
9- "EULA": "", 9- "EULA": "",
10- "help": "", 10- "help": "",
11- "url": "", 11- "url": "",
12- "logo": "", 12- "logo": "",
13- "dependencies": [ 13- "dependencies": [
14-   { 14+   {
15-     "appId": "0000000-0000-0000-0000-000000000000", 15+   "appId": "f7fibaa9-8d60-4745-8113-2c
16-     "publisher": "Default Publisher", 16+   "publisher": "My Name",
17-     "name": "Default Base App Name", 17+   "name": "My Base App",
18-     "version": "2.0.0.0" 18+   "version": "1.0.0.0"
19-   } 19-   }
20- ], 20- ],
21- "screenshots": [ 21- ],
22- ], 22- ],
23- "application": "16.0.0.0", 23- ],
24- "platform": "16.0.0.0", 24- ],
25- "idRanges": [ 25- ],
26-   { 26- },
27-     "from": 50100, 27-   {
28-     "to": 50149 28+   "from": 80100,
29-   } 29+   "to": 80149
30- ], 30- ],
31- ] 31- ],
32- ] 32- ],
33- ] 33- ],
34- ] 34- ],
35- ] 35- ],
36- ] 36- ],
37- ] 37- ],
38- ] 38- ],
39- ] 39- ],
40- ] 40- ],
41- ] 41- ],
42- ] 42- ],
43- ] 43- ],
44- ] 44- ],
45- ] 45- ],
46- ] 46- ],
47- ] 47- ],
48- ] 48- ],
49- ] 49- ],
50- ] 50- ],
51- ] 51- ],
52- ] 52- ],
53- ] 53- ],
54- ] 54- ],
55- ] 55- ],
56- ] 56- ],
57- ] 57- ],
58- ] 58- ],
59- ] 59- ],
60- ] 60- ],
61- ] 61- ],
62- ] 62- ],
63- ] 63- ],
64- ] 64- ],
65- ] 65- ],
66- ] 66- ],
67- ] 67- ],
68- ] 68- ],
69- ] 69- ],
70- ] 70- ],
71- ] 71- ],
72- ] 72- ],
73- ] 73- ],
74- ] 74- ],
75- ] 75- ],
76- ] 76- ],
77- ] 77- ],
78- ] 78- ],
79- ] 79- ],
80- ] 80- ],
81- ] 81- ],
82- ] 82- ],
83- ] 83- ],
84- ] 84- ],
85- ] 85- ],
86- ] 86- ],
87- ] 87- ],
88- ] 88- ],
89- ] 89- ],
90- ] 90- ],
91- ] 91- ],
92- ] 92- ],
93- ] 93- ],
94- ] 94- ],
95- ] 95- ],
96- ] 96- ],
97- ] 97- ],
98- ] 98- ],
99- ] 99- ],
100- ] 100- ],
101- ] 101- ],
102- ] 102- ],
103- ] 103- ],
104- ] 104- ],
105- ] 105- ],
106- ] 106- ],
107- ] 107- ],
108- ] 108- ],
109- ] 109- ],
110- ] 110- ],
111- ] 111- ],
112- ] 112- ],
113- ] 113- ],
114- ] 114- ],
115- ] 115- ],
116- ] 116- ],
117- ] 117- ],
118- ] 118- ],
119- ] 119- ],
120- ] 120- ],
121- ] 121- ],
122- ] 122- ],
123- ] 123- ],
124- ] 124- ],
125- ] 125- ],
126- ] 126- ],
127- ] 127- ],
128- ] 128- ],
129- ] 129- ],
130- ] 130- ],
131- ] 131- ],
132- ] 132- ],
133- ] 133- ],
134- ] 134- ],
135- ] 135- ],
136- ] 136- ],
137- ] 137- ],
138- ] 138- ],
139- ] 139- ],
140- ] 140- ],
141- ] 141- ],
142- ] 142- ],
143- ] 143- ],
144- ] 144- ],
145- ] 145- ],
146- ] 146- ],
147- ] 147- ],
148- ] 148- ],
149- ] 149- ],
150- ] 150- ],
151- ] 151- ],
152- ] 152- ],
153- ] 153- ],
154- ] 154- ],
155- ] 155- ],
156- ] 156- ],
157- ] 157- ],
158- ] 158- ],
159- ] 159- ],
160- ] 160- ],
161- ] 161- ],
162- ] 162- ],
163- ] 163- ],
164- ] 164- ],
165- ] 165- ],
166- ] 166- ],
167- ] 167- ],
168- ] 168- ],
169- ] 169- ],
170- ] 170- ],
171- ] 171- ],
172- ] 172- ],
173- ] 173- ],
174- ] 174- ],
175- ] 175- ],
176- ] 176- ],
177- ] 177- ],
178- ] 178- ],
179- ] 179- ],
180- ] 180- ],
181- ] 181- ],
182- ] 182- ],
183- ] 183- ],
184- ] 184- ],
185- ] 185- ],
186- ] 186- ],
187- ] 187- ],
188- ] 188- ],
189- ] 189- ],
190- ] 190- ],
191- ] 191- ],
192- ] 192- ],
193- ] 193- ],
194- ] 194- ],
195- ] 195- ],
196- ] 196- ],
197- ] 197- ],
198- ] 198- ],
199- ] 199- ],
200- ] 200- ],
201- ] 201- ],
202- ] 202- ],
203- ] 203- ],
204- ] 204- ],
205- ] 205- ],
206- ] 206- ],
207- ] 207- ],
208- ] 208- ],
209- ] 209- ],
210- ] 210- ],
211- ] 211- ],
212- ] 212- ],
213- ] 213- ],
214- ] 214- ],
215- ] 215- ],
216- ] 216- ],
217- ] 217- ],
218- ] 218- ],
219- ] 219- ],
220- ] 220- ],
221- ] 221- ],
222- ] 222- ],
223- ] 223- ],
224- ] 224- ],
225- ] 225- ],
226- ] 226- ],
227- ] 227- ],
228- ] 228- ],
229- ] 229- ],
230- ] 230- ],
231- ] 231- ],
232- ] 232- ],
233- ] 233- ],
234- ] 234- ],
235- ] 235- ],
236- ] 236- ],
237- ] 237- ],
238- ] 238- ],
239- ] 239- ],
240- ] 240- ],
241- ] 241- ],
242- ] 242- ],
243- ] 243- ],
244- ] 244- ],
245- ] 245- ],
246- ] 246- ],
247- ] 247- ],
248- ] 248- ],
249- ] 249- ],
250- ] 250- ],
251- ] 251- ],
252- ] 252- ],
253- ] 253- ],
254- ] 254- ],
255- ] 255- ],
256- ] 256- ],
257- ] 257- ],
258- ] 258- ],
259- ] 259- ],
260- ] 260- ],
261- ] 261- ],
262- ] 262- ],
263- ] 263- ],
264- ] 264- ],
265- ] 265- ],
266- ] 266- ],
267- ] 267- ],
268- ] 268- ],
269- ] 269- ],
270- ] 270- ],
271- ] 271- ],
272- ] 272- ],
273- ] 273- ],
274- ] 274- ],
275- ] 275- ],
276- ] 276- ],
277- ] 277- ],
278- ] 278- ],
279- ] 279- ],
280- ] 280- ],
281- ] 281- ],
282- ] 282- ],
283- ] 283- ],
284- ] 284- ],
285- ] 285- ],
286- ] 286- ],
287- ] 287- ],
288- ] 288- ],
289- ] 289- ],
290- ] 290- ],
291- ] 291- ],
292- ] 292- ],
293- ] 293- ],
294- ] 294- ],
295- ] 295- ],
296- ] 296- ],
297- ] 297- ],
298- ] 298- ],
299- ] 299- ],
300- ] 300- ],
301- ] 301- ],
302- ] 302- ],
303- ] 303- ],
304- ] 304- ],
305- ] 305- ],
306- ] 306- ],
307- ] 307- ],
308- ] 308- ],
309- ] 309- ],
310- ] 310- ],
311- ] 311- ],
312- ] 312- ],
313- ] 313- ],
314- ] 314- ],
315- ] 315- ],
316- ] 316- ],
317- ] 317- ],
318- ] 318- ],
319- ] 319- ],
320- ] 320- ],
321- ] 321- ],
322- ] 322- ],
323- ] 323- ],
324- ] 324- ],
325- ] 325- ],
326- ] 326- ],
327- ] 327- ],
328- ] 328- ],
329- ] 329- ],
330- ] 330- ],
331- ] 331- ],
332- ] 332- ],
333- ] 333- ],
334- ] 334- ],
335- ] 335- ],
336- ] 336- ],
337- ] 337- ],
338- ] 338- ],
339- ] 339- ],
340- ] 340- ],
341- ] 341- ],
342- ] 342- ],
343- ] 343- ],
344- ] 344- ],
345- ] 345- ],
346- ] 346- ],
347- ] 347- ],
348- ] 348- ],
349- ] 349- ],
350- ] 350- ],
351- ] 351- ],
352- ] 352- ],
353- ] 353- ],
354- ] 354- ],
355- ] 355- ],
356- ] 356- ],
357- ] 357- ],
358- ] 358- ],
359- ] 359- ],
360- ] 360- ],
361- ] 361- ],
362- ] 362- ],
363- ] 363- ],
364- ] 364- ],
365- ] 365- ],
366- ] 366- ],
367- ] 367- ],
368- ] 368- ],
369- ] 369- ],
370- ] 370- ],
371- ] 371- ],
372- ] 372- ],
373- ] 373- ],
374- ] 374- ],
375- ] 375- ],
376- ] 376- ],
377- ] 377- ],
378- ] 378- ],
379- ] 379- ],
380- ] 380- ],
381- ] 381- ],
382- ] 382- ],
383- ] 383- ],
384- ] 384- ],
385- ] 385- ],
386- ] 386- ],
387- ] 387- ],
388- ] 388- ],
389- ] 389- ],
390- ] 390- ],
391- ] 391- ],
392- ] 392- ],
393- ] 393- ],
394- ] 394- ],
395- ] 395- ],
396- ] 396- ],
397- ] 397- ],
398- ] 398- ],
399- ] 399- ],
400- ] 400- ],
401- ] 401- ],
402- ] 402- ],
403- ] 403- ],
404- ] 404- ],
405- ] 405- ],
406- ] 406- ],
407- ] 407- ],
408- ] 408- ],
409- ] 409- ],
410- ] 410- ],
411- ] 411- ],
412- ] 412- ],
413- ] 413- ],
414- ] 414- ],
415- ] 415- ],
416- ] 416- ],
417- ] 417- ],
418- ] 418- ],
419- ] 419- ],
420- ] 420- ],
421- ] 421- ],
422- ] 422- ],
423- ] 423- ],
424- ] 424- ],
425- ] 425- ],
426- ] 426- ],
427- ] 427- ],
428- ] 428- ],
429- ] 429- ],
430- ] 430- ],
431- ] 431- ],
432- ] 432- ],
433- ] 433- ],
434- ] 434- ],
435- ] 435- ],
436- ] 436- ],
437- ] 437- ],
438- ] 438- ],
439- ] 439- ],
440- ] 440- ],
441- ] 441- ],
442- ] 442- ],
443- ] 443- ],
444- ] 444- ],
445- ] 445- ],
446- ] 446- ],
447- ] 447- ],
448- ] 448- ],
449- ] 449- ],
450- ] 450- ],
451- ] 451- ],
452- ] 452- ],
453- ] 453- ],
454- ] 454- ],
455- ] 455- ],
456- ] 456- ],
457- ] 457- ],
458- ] 458- ],
459- ] 459- ],
460- ] 460- ],
461- ] 461- ],
462- ] 462- ],
463- ] 463- ],
464- ] 464- ],
465- ] 465- ],
466- ] 466- ],
467- ] 467- ],
468- ] 468- ],
469- ] 469- ],
470- ] 470- ],
471- ] 471- ],
472- ] 472- ],
473- ] 473- ],
474- ] 474- ],
475- ] 475- ],
476- ] 476- ],
477- ] 477- ],
478- ] 478- ],
479- ] 479- ],
480- ] 480- ],
481- ] 481- ],
482- ] 482- ],
483- ] 483- ],
484- ] 484- ],
485- ] 485- ],
486- ] 486- ],
487- ] 487- ],
488- ] 488- ],
489- ] 489- ],
490- ] 490- ],
491- ] 491- ],
492- ] 492- ],
493- ] 493- ],
494- ] 494- ],
495- ] 495- ],
496- ] 496- ],
497- ] 497- ],
498- ] 498- ],
499- ] 499- ],
500- ] 500- ],
501- ] 501- ],
502- ] 502- ],
503- ] 503- ],
504- ] 504- ],
505- ] 505- ],
506- ] 506- ],
507- ] 507- ],
508- ] 508- ],
509- ] 509- ],
510- ] 510- ],
511- ] 511- ],
512- ] 512- ],
513- ] 513- ],
514- ] 514- ],
515- ] 515- ],
516- ] 516- ],
517- ] 517- ],
518- ] 518- ],
519- ] 519- ],
520- ] 520- ],
521- ] 521- ],
522- ] 522- ],
523- ] 523- ],
524- ] 524- ],
525- ] 525- ],
526- ] 526- ],
527- ] 527- ],
528- ] 528- ],
529- ] 529- ],
530- ] 530- ],
531- ] 531- ],
532- ] 532- ],
533- ] 533- ],
534- ] 534- ],
535- ] 535- ],
536- ] 536- ],
537- ] 537- ],
538- ] 538- ],
539- ] 539- ],
540- ] 540- ],
541- ] 541- ],
542- ] 542- ],
543- ] 543- ],
544- ] 544- ],
545- ] 545- ],
546- ] 546- ],
547- ] 547- ],
548- ] 548- ],
549- ] 549- ],
550- ] 550- ],
551- ] 551- ],
552- ] 552- ],
553- ] 553- ],
554- ] 554- ],
555- ] 555- ],
556- ] 556- ],
557- ] 557- ],
558- ] 558- ],
559- ] 559- ],
560- ] 560- ],
561- ] 561- ],
562- ] 562- ],
563- ] 563- ],
564- ] 564- ],
565- ] 565- ],
566- ] 566- ],
567- ] 567- ],
568- ] 568- ],
569- ] 569- ],
570- ] 570- ],
571- ] 571- ],
572- ] 572- ],
573- ] 573- ],
574- ] 574- ],
575- ] 575- ],
576- ] 576- ],
577- ] 577- ],
578- ] 578- ],
579- ] 579- ],
580- ] 580- ],
581- ] 581- ],
582- ] 582- ],
583- ] 583- ],
584- ] 584- ],
585- ] 585- ],
586- ] 586- ],
587- ] 587- ],
588- ] 588- ],
589- ] 589- ],
590- ] 590- ],
591- ] 591- ],
592- ] 592- ],
593- ] 593- ],
594- ] 594- ],
595- ] 595- ],
596- ] 596- ],
597- ] 597- ],
598- ] 598- ],
599- ] 599- ],
600- ] 600- ],
601- ] 601- ],
602- ] 602- ],
603- ] 603- ],
604- ] 604- ],
605- ] 605- ],
606- ] 606- ],
607- ] 607- ],
608- ] 608- ],
609- ] 609- ],
610- ] 610- ],
611- ] 611- ],
612- ] 612- ],
613- ] 613- ],
614- ] 614- ],
615- ] 615- ],
616- ] 616- ],
617- ] 617- ],
618- ] 618- ],
619- ] 619- ],
620- ] 620- ],
621- ] 621- ],
622- ] 622- ],
623- ] 623- ],
624- ] 624- ],
625- ] 625- ],
626- ] 626- ],
627- ] 627- ],
628- ] 628- ],
629- ] 629- ],
630- ] 630- ],
631- ] 631- ],
632- ] 632- ],
633- ] 633- ],
634- ] 634- ],
635- ] 635- ],
636- ] 636- ],
637- ] 637- ],
638- ] 638- ],
639- ] 639- ],
640- ] 640- ],
641- ] 641- ],
642- ] 642- ],
643- ] 643- ],
644- ] 644- ],
645- ] 645- ],
646- ] 646- ],
647- ] 647- ],
648- ] 648- ],
649- ] 649- ],
650- ] 650- ],
651- ] 651- ],
652- ] 652- ],
653- ] 653- ],
654- ] 654- ],
655- ] 655- ],
656- ] 656- ],
657- ] 657- ],
658- ] 658- ],
659- ] 659- ],
660- ] 660- ],
661- ] 661- ],
662- ] 662- ],
663- ] 663- ],
664- ] 664- ],
665- ] 665- ],
666- ] 666- ],
667- ] 667- ],
668- ] 668- ],
669- ] 669- ],
670- ] 670- ],
671- ] 671- ],
672- ] 672- ],
673- ] 673- ],
674- ] 674- ],
675- ] 675- ],
676- ] 676- ],
677- ] 677- ],
678- ] 678- ],
679- ] 679- ],
680- ] 680- ],
681- ] 681- ],
682- ] 682- ],
683- ] 683- ],
684- ] 684- ],
685- ] 685- ],
686- ] 686- ],
687- ] 687- ],
688- ] 688- ],
689- ] 689- ],
690- ] 690- ],
691- ] 691- ],
692- ] 692- ],
693- ] 693- ],
694- ] 694- ],
695- ] 695- ],
696- ] 696- ],
697- ] 697- ],
698- ] 698- ],
699- ] 699- ],
700- ] 700- ],
701- ] 701- ],
702- ] 702- ],
703- ] 703- ],
704- ] 704- ],
705- ] 705- ],
706- ] 706- ],
707- ] 707- ],
708- ] 708- ],
709- ] 709- ],
710- ] 710- ],
711- ] 711- ],
712- ] 712- ],
713- ] 713- ],
714- ] 714- ],
715- ] 715- ],
716- ] 716- ],
717- ] 717- ],
718- ] 718- ],
719- ] 719- ],
720- ] 720- ],
721- ] 721- ],
722- ] 722- ],
723- ] 723- ],
724- ] 724- ],
725- ] 725- ],
726- ] 726- ],
727- ] 727- ],
728- ] 728- ],
729- ] 729- ],
730- ] 730- ],
731- ] 731- ],
732- ] 732- ],
733- ] 733- ],
734- ] 734- ],
735- ] 735- ],
736- ] 736- ],
737- ] 737- ],
738- ] 738- ],
739- ] 739- ],
740- ] 740- ],
741- ] 741- ],
742- ] 742- ],
743- ] 743- ],
744- ] 744- ],
745- ] 745- ],
746- ] 746- ],
747- ] 747- ],
748- ] 748- ],
749- ] 749- ],
750- ] 750- ],
751- ] 751- ],
752- ] 752- ],
753- ] 753- ],
754- ] 754- ],
755- ] 755- ],
756- ] 756- ],
757- ] 757- ],
758- ] 758- ],
759- ] 759- ],
760- ] 760- ],
761- ] 761- ],
762- ] 762- ],
763- ] 763- ],
764- ] 764- ],
765- ] 765- ],
766- ] 766- ],
767- ] 767- ],
768- ] 768- ],
769- ] 769- ],
770- ] 770- ],
771- ] 771- ],
772- ] 772- ],
773- ] 773- ],
774- ] 774- ],
775- ] 775- ],
776- ] 776- ],
777- ] 777- ],
778- ] 778- ],
779- ] 779- ],
780- ] 780- ],
781- ] 781- ],
782- ] 782- ],
783- ] 783- ],
784- ] 784- ],
785- ] 785- ],
786- ] 786- ],
787- ] 787- ],
788- ] 788- ],
789- ] 789- ],
790- ] 790- ],
791- ] 791- ],
792- ] 792- ],
793- ] 793- ],
794- ] 794- ],
795- ] 795- ],
796- ] 796- ],
797- ] 797- ],
798- ] 798- ],
799- ] 799- ],
800- ] 800- ],
801- ] 801- ],
802- ] 802- ],
803- ] 803- ],
804- ] 804- ],
805- ] 805- ],
806- ] 806- ],
807- ] 807- ],
808- ] 808- ],
809- ] 809- ],
810- ] 810- ],
811- ] 811- ],
812- ] 812- ],
813- ] 813- ],
814- ] 814- ],
815- ] 815- ],
816- ] 816- ],
817- ] 817- ],
818- ] 818- ],
819- ] 819- ],
820- ] 820- ],
821- ] 821- ],
822- ] 822- ],
823- ] 823- ],
824- ] 824- ],
825- ] 825- ],
826- ] 826- ],
827- ] 827- ],
828- ] 828- ],
829- ] 829- ],
830- ] 830- ],
831- ] 831- ],
832- ] 832- ],
833- ] 833- ],
834- ] 834- ],
835- ] 835- ],
836- ] 836- ],
837- ] 837- ],
838- ] 838- ],
839- ] 839- ],
840- ] 840- ],
841- ] 841- ],
842- ] 842- ],
843- ] 843- ],
844- ] 844- ],
845- ] 845- ],
846- ] 846- ],
847- ] 847- ],
848- ] 848- ],
849- ] 849- ],
850- ] 850- ],
851- ] 851- ],
852- ] 852- ],
853- ] 853- ],
854- ] 854- ],
855- ] 855- ],
856- ] 856- ],
857- ] 857- ],
858- ] 858- ],
859- ] 859- ],
860- ] 860- ],
861- ] 861- ],
862- ] 862- ],
863- ] 863- ],
864- ] 864- ],
865- ] 865- ],
866- ] 866- ],
867- ] 867- ],
868- ] 868- ],
869- ] 869- ],
870- ] 870- ],
871- ] 871- ],
872- ] 872- ],
873- ] 873- ],
874- ] 874- ],
875- ] 875- ],
876- ] 876- ],
877- ] 877- ],
878- ] 878- ],
879- ] 879- ],
880- ] 880- ],
881- ] 881- ],
882- ] 882- ],
883- ] 883- ],
884- ] 884- ],
885- ] 885- ],
886- ] 886- ],
887- ] 887- ],
888- ] 888- ],
889- ] 889- ],
890- ] 890- ],
891- ] 891- ],
892- ] 892- ],
893- ] 893- ],
894- ] 894- ],
895- ] 895- ],
896- ] 896- ],
897- ] 897- ],
898- ] 898- ],
899- ] 899- ],
900- ] 900- ],
901- ] 901- ],
902- ] 902- ],
903- ] 903- ],
904- ] 904- ],
905- ] 905- ],
906- ] 906- ],
907- ] 907- ],
908- ] 908- ],
909- ] 909- ],
910- ] 910- ],
911- ] 911- ],
912- ] 912- ],
913- ] 913- ],
914- ] 914- ],
915- ] 915- ],
916- ] 916- ],
917- ] 917- ],
918- ] 918- ],
919- ] 919- ],
920- ] 920- ],
921- ] 921- ],
922- ] 922- ],
923- ] 923- ],
924- ] 924- ],
925- ] 925- ],
926- ] 926- ],
927- ] 927- ],
928- ] 928- ],
929- ] 929- ],
930- ] 930- ],
931- ] 931- ],
932- ] 932- ],
933- ] 933- ],
934- ] 934- ],
935- ] 935- ],
936- ] 936- ],
937- ] 937- ],
938- ] 938- ],
939- ] 939- ],
940- ] 940- ],
941- ] 941- ],
942- ] 942- ],
943- ] 943- ],
944- ] 944- ],
945- ] 945- ],
946- ] 946- ],
947- ] 947- ],
948- ] 948- ],
949- ] 949- ],
950- ] 950- ],
951- ] 951- ],
952- ] 952- ],
953- ] 953- ],
954- ] 954- ],
955- ] 955- ],
956- ] 956- ],
957- ] 957- ],
958- ] 958-
```



File Edit Selection View Go Run Terminal Help Local-DevEnv.ps1 - HelloWorld (Workspace) - Visual Studio Code [Administrator]

EXPLORER OPEN EDITORS

HELLLOWORLD (WORKSPACE)

- > base
- > app
- > test
- scripts
 - > .snapshots
 - ! Current.yml
 - ! DevOps-Pipeline.ps1
 - ! Install-BcContainerHelper.ps1
 - ! Local-DevEnv.ps1
 - ! Local-Pipeline.ps1
 - ! MySolution.ps1
 - ! NextMajor.yml
 - ! NextMinor.yml
 - > Read-Settings.ps1
- settings.json

MySolution.ps1 > Local-DevEnv.ps1

```
scripts > Local-DevEnv.ps1 > ...
You, 2 days ago | 1 author (You)
1 Param(
2     [Parameter(Mandatory=$false)]
3     [string] $version = "ci"
4 )
5
6 $baseFolder = (Get-Item (Join-Path $PSScriptRoot "")).FullName
7 . (Join-Path $PSScriptRoot "Read-Settings.ps1") -version $version
8
9 $bcContainerHelperVersion = "latest"
10 if ($settings.PSObject.Properties.Name -eq 'bcContainerHelperVersion' -and $settings.bcContainerHelperVersion) {
11     $bcContainerHelperVersion = $settings.bcContainerHelperVersion
12 }
13 Write-Host "Use bcContainerHelper Version: $bcContainerHelperVersion"
14 . (Join-Path $PSScriptRoot "Install-BcContainerHelper.ps1") -bcContainerHelperVersion $bcContainerHelperVersion
15
16 if ($genericImageName) {
17     $bcContainerHelperConfig.genericImageName = $genericImageName
18 }
19
20 if ("$vaultNameForLocal" -eq "") -or !(Get-AzKeyVault -VaultName $vaultNameForLocal)) {
21     throw "You need to setup a Key Vault for use with local pipelines"
22 }
```

PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL

Windows PowerShell

Copyright (C) Microsoft Corporation. All rights reserved.

Loading personal and system profiles took 2487ms.

PS C:\Users\vmadmin\Documents\VAL\myfirstapp\scripts>

PS C:\Users\vmadmin\Documents\VAL\myfirstapp\scripts> connect-azaccount

Account	SubscriptionName	TenantId	Environment
freddykg@microsoft.com	workshops and Demos	72f988bf-86f1-41af-91ab-2d7cd011db47	AzureCloud

PS C:\Users\vmadmin\Documents\VAL\myfirstapp\scripts> |

Now run .\Local-DevEnv.ps1

This script will create a development container, compile and publish all apps using the Dev Endpoint and leave the container running, allowing you to modify and publish individual apps afterwards.

Note: that the local-devenv script assumes that you have a key vault called **BuildVariables** with at least a **licensefile** and a **password** secret.

You should see the script running, creating the container, compiling and publishing apps and modifying the launch.json file to prepare your VS Code for Rapid Application Development.



```
dotnet new -n local-Development
dotnet new -h
dotnet new -i Microsoft.AspNetCore.Razor.Templates::6.0.0
```

```

File Edit Selection View Go Run Terminal Help Local-DevEnv.ps1 - HelloWorld (Workspace) - Visual Studio Code [Administrator]
EXPLORER > MySolution.ps1 > Local-DevEnv.ps1
OPEN EDITORS
HELLOWORLD (WORKSPACE)
> base
> app
> test
> scripts
> snapshots
! Cl.yml
> Cleanup.ps1
! Current.yml
> DevOps-Pipeline.ps1
> Install-BcContainerHelper.ps1
> Local-DevEnv.ps1 M
> Local-Pipeline.ps1
> MySolution.ps1 1, M
! NextMajor.yml
! NextMinor.yml
> Read-Settings.ps1
! settings.json M
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL
1: powershell
[AL] Using Symbols Folder: C:\Users\vmadmin\Documents\AL\myfirstapp\test\alPackages
Disabling SSL Verification
Re-enabling SSL Verification
Compiling...
.alc.exe /project:"c:\sources\test" /packagecachePath:"c:\sources\test\alPackages" /out:"c:\sources\test\My Name_My Test App_1.0.0.0.app" /assemblyprobingsPaths:"c:\Program Files (x86)\Microsoft Dynamics NAV170\RoleTailored Client","c:\Program Files\Microsoft Dynamics NAV\170\Service","c:\Program Files (x86)\Open XML SDK\2.5\lib","c:\Windows\Microsoft.NET\Assembly","c:\Test Assemblies\Mock Assemblies"
Microsoft (R) AL Compiler version 6.0.5.10826
Copyright (C) Microsoft Corporation. All rights reserved

Compilation started for project 'My Test App' containing '1' files at '13:09:47.39'.

Compilation ended at '13:09:57.467'.

C:\Users\vmadmin\Documents\AL\myfirstapp\test\My Name_My Test App_1.0.0.0.app successfully created in 12 seconds
Disabling SSL Verification
Publishing My Name_My Test App_1.0.0.0.app to https://172.17.201.196:7049/BC/dev/apps?schemaUpdateMode=synchronize&tenant=default
Re-enabling SSL Verification
App successfully published
Modifying C:\Users\vmadmin\Documents\AL\myfirstapp\test\vscode\launch.json

Compiling apps, test apps and importing test toolkit took 97 seconds
AL Pipeline finished in 230 seconds
PS C:\Users\vmadmin\Documents\AL\myfirstapp\scripts>

```

You, 2 days ago Ln 22, Col 2 Spaces: 4 UTF-8 with BOM CRLF PowerShell 5.1

Press **Ctrl+Shift+P** and execute Developer: Reload Window to get rid of cached compiler errors.

```

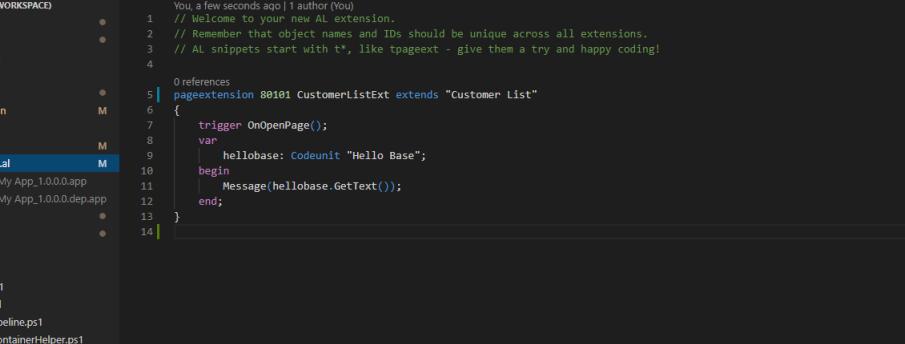
File Edit Selection View Go Run Terminal Help HelloWorld.al - HelloWorld (Workspace) - Visual Studio Code [Administrator]
EXPLORER > HelloWorld.al
OPEN EDITORS
HELLOWORLD (WORKSPACE)
> base
> app
> allPackages
> snapshots
> vscode
! app.json M
> HelloWorld.al M
> My Name_My App_1.0.0.0.app
> test
> scripts
> snapshots
! Cl.yml
> Cleanup.ps1
! Current.yml
> DevOps-Pipeline.ps1
> Install-BcContainerHelper.ps1
> Local-DevEnv.ps1 M
> Local-Pipeline.ps1
> MySolution.ps1 M
! NextMajor.yml
! NextMinor.yml
> Read-Settings.ps1
! settings.json M
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
2: PowerShell Integrated Console v2020.6.0 <=====

PS C:\Users\vmadmin\Documents\AL\myfirstapp\base>

```

You, a few seconds ago Ln 12, Col 9 Spaces: 4 UTF-8 CRLF AL

Now you can navigate to **HelloWorld.al** and press **F5**.



File Edit Selection View Go Run Terminal Help HelloWorld.al - HelloWorld (Workspace) - Visual Studio Code [Administrator]

EXPLORER

OPEN EDITORS

HELLOWORLD (WORKSPACE)

- > base
- app
 - > _alPackages
 - > _snapshots
 - > .vscode
 - launch.json
 - radjson
 - app.json
- >HelloWorld.al
- > test
- > scripts
 - > snapshots
 - Clyml
 - Cleanup.ps1
 - Currentlyml
 - DevOps-Pipeline.ps1
 - Install-9cContainerHelper.ps1
 - Local-DevEnv.ps1
 - Local-Pipelines.ps1
 - MySolution.ps1
 - NextMajor.yml
 - NextMinor.yml
 - Read-Settings.ps1
 - settings.json
- > OUTLINE
- > TIMELINE
- DOCKER CONTAINERS
- DOCKER IMAGES
- AZURE CONTAINER REGISTRY
- DOCKER HUB
- SUGGESTED DOCKER HUB IMAGES

HELLOWORLD.al ×

app > HelloWorld.al > ...

```
You, a few seconds ago | 1 author (You)
1  // Welcome to your new AL extension.
2  // Remember that object names and IDs should be unique across all extensions.
3  // AL snippets start with t*, like tpageext - give them a try and happy coding!
4

0 references
5  pageextension 88101 CustomerListExt extends "Customer List"
6  {
7      trigger OnOpenPage();
8      var
9          hellobase: Codeunit "Hello Base";
10     begin
11         Message(hellobase.GetText());
12     end;
13 }
14 |
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Filter (e.g. text | exclude)

[2020-10-25 14:17:34.64] Targeting server '<https://fkworkshop.northeurope.cloudapp.azure.com>', server instance 'myapp-cidev' and tenant 'default'.
[2020-10-25 14:17:34.74] Using user name and password authentication. User name used is: 'admin'.
[2020-10-25 14:17:34.76] Sending request to <https://fkworkshop.northeurope.cloudapp.azure.com/myapp-cidev/dev/metadata?tenant=default>
[2020-10-25 14:17:35.41] Publishing package to tenant 'default'.
[2020-10-25 14:17:35.34] Targeting server '<https://fkworkshop.northeurope.cloudapp.azure.com>', server instance 'myapp-cidev' and tenant 'default'.
[2020-10-25 14:17:35.34] Using user name and password authentication. User name used is: 'admin'.
[2020-10-25 14:17:35.34] Sending request to <https://fkworkshop.northeurope.cloudapp.azure.com/myapp-cidev/dev/apps?tenant=default&schemaUpdateMo>
[2020-10-25 14:17:35.34] Success: The package 'My Name My App 1.0.0.0.dep.app' has been published to the server.

Ln 14 Col 1 Spaces: 4 UTF-8 CRLF AL

And then it should automatically launch the Web Client.

The screenshot shows the Dynamics 365 Business Central interface. At the top, there are navigation links for Finance, Cash Management, Sales, Purchasing, Setup & Extensions, and a search bar. Below the header, a list of customers is displayed with columns for No. (Customer ID), Name, Responsibility Center, Location Code, Phone No., and Contact. The first customer, Adatum Corporation, is selected. A modal dialog box is open in the center, displaying the message "App Published: Hello World Base!". In the bottom right corner of the modal, there is an "OK" button. To the right of the customer list, there is a sidebar titled "Sell-to Customer Sales History" with a table showing various sales stages: Ongoing Sales Quotes, Ongoing Sales Blanket Orders, Ongoing Sales Orders, Ongoing Sales Invoices, Ongoing Sales Return Orders, Ongoing Sales Credit Memos, Posted Sales Shipments, Posted Sales Invoices, Posted Sales Return Receipts, and Posted Sales Credit Memos. At the bottom of the screen, there is a "Customer Statistics" section.

Check-in your changes

Typically, we do not check-in `launch.json`, in VS Code, you can add these to `.gitignore` in order to avoid tracking them if you like, in this workshop we will just check in everything.

Click the Source Control icon and press + on the changes line to stage all your changes.

```
version: "0.2.0",
"configurations": [
  {
    "type": "al",
    "request": "launch",
    "name": "Local Sandbox",
    "server": "http://helloworld-current",
    "serverInstance": "BC",
    "port": 7089,
    "tenant": "default",
    "authentication": "UserPassword",
    "startupObjectID": 22,
    "startupObjectType": "Page",
    "breakOnErrors": true
  }
]
```

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | Filter (e.g. text | exclude)

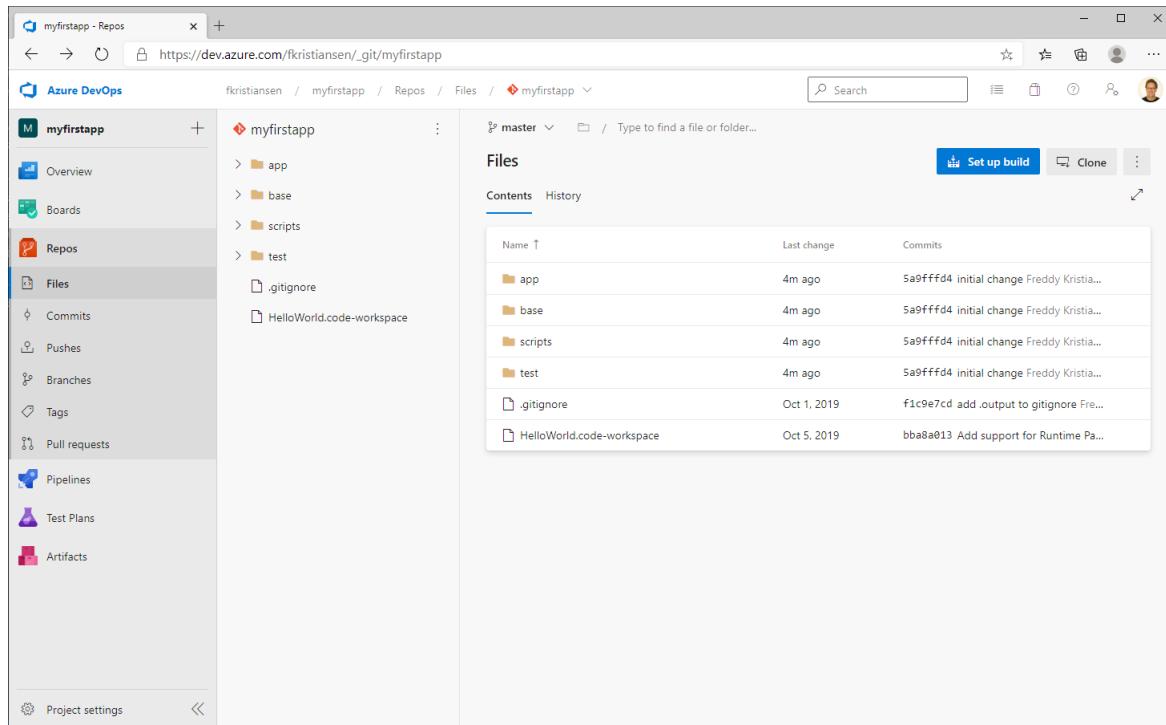
```
[2020-10-25 15:56:40.21] Publishing AL application using launch configuration 'Local Sandbox'.
[2020-10-25 15:56:40.67] Targeting server 'https://fkworkshop.northeurope.cloudapp.azure.com', server instance 'myapp-cidev' and tenant 'default'.
[2020-10-25 15:56:40.69] Using user name and password authentication. User name used is: 'admin'.
[2020-10-25 15:56:40.71] Sending request to https://fkworkshop.northeurope.cloudapp.azure.com/myapp-cidev/dev/metadata?tenant=default
[2020-10-25 15:56:41.33] Publishing package to tenant 'default'.
[2020-10-25 15:56:41.33] Targeting server 'https://fkworkshop.northeurope.cloudapp.azure.com', server instance 'myapp-cidev' and tenant 'default'.
[2020-10-25 15:56:41.33] Using user name and password authentication. User name used is: 'admin'.
[2020-10-25 15:56:41.34] Sending request to https://fkworkshop.northeurope.cloudapp.azure.com/myapp-cidev/dev/apps?tenant=default&schemaUpdateMode=synchronize&dependencyPublishingOptions=default
[2020-10-25 15:56:43.50] Success: The package 'My Name_My App_1.0.0.0.dep.app' has been published to the server.
```

Staged changes are changes you want to commit. Enter a commit message and hit Commit (or use **Ctrl+Enter**). In the bottom left corner, identify the **Synchronize Changes** symbol and hit that. You might be asked to login to your devops account in order to push your changes.



Create a Build Pipeline

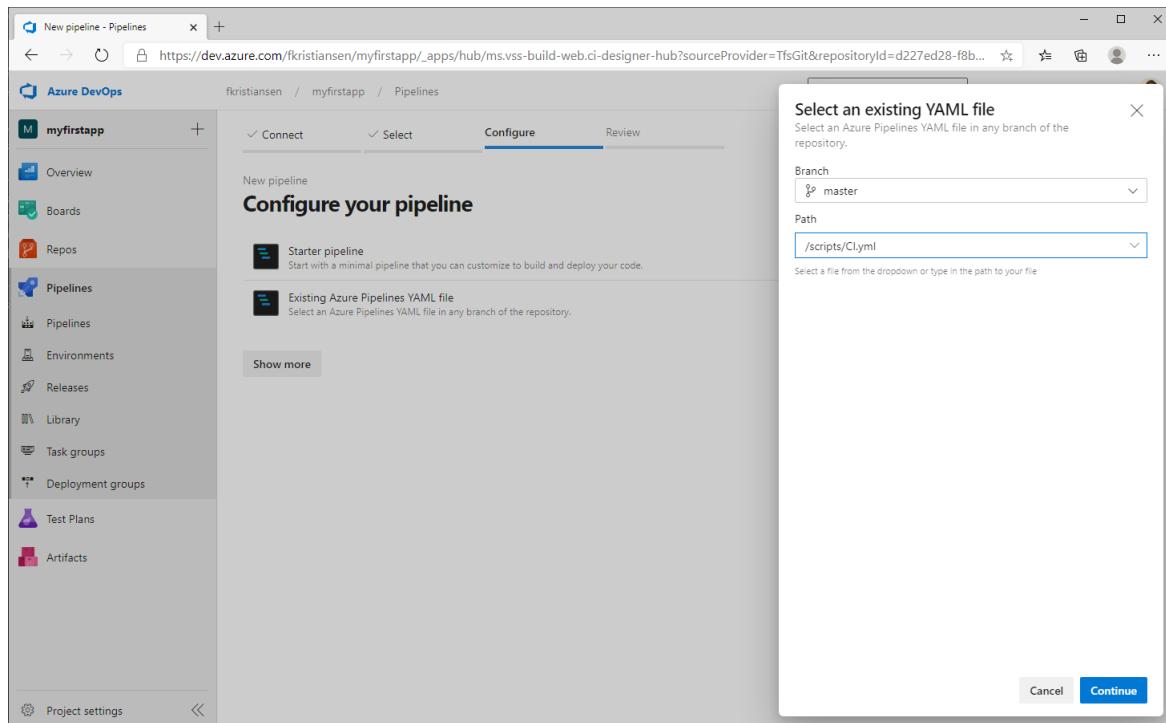
In Azure DevOps, under your Project, **Repos** -> **Files**, you will see a button called **Set up build**. Click it.



The screenshot shows the 'Files' section of the Azure DevOps interface for the 'myfirstapp' repository. The left sidebar shows navigation options like Overview, Boards, Repos, Files (which is selected), Commits, Pushes, Branches, Tags, Full requests, Pipelines, Test Plans, and Artifacts. The 'Project settings' link is at the bottom of the sidebar. The main area displays a list of files and folders: 'app', 'base', 'scripts', 'test', '.gitignore', and 'HelloWorld.code-workspace'. A 'Set up build' button is prominently displayed in the top right of the file list area. The URL in the browser is https://dev.azure.com/fkristiansen/_git/myfirstapp.

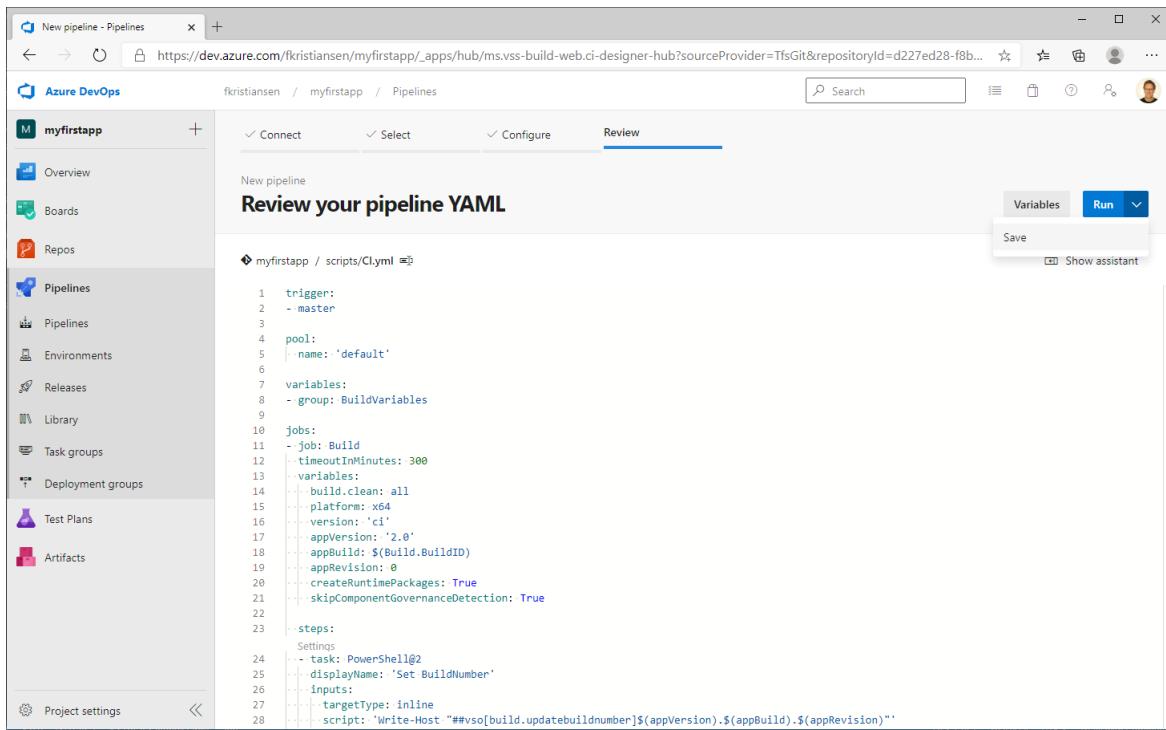
In the **Configure your pipeline**, select **Existing Azure Pipelines YAML file**.

Select **./.azureDevOps/Cl.yml** in the path for the YAML file and press **Continue**.



The screenshot shows the 'Configure your pipeline' step in the Azure DevOps pipeline creation wizard. The left sidebar shows 'Pipelines' selected. The main area has tabs: 'Connect', 'Select', 'Configure' (which is selected), and 'Review'. Under 'Configure', there are two options: 'Starter pipeline' and 'Existing Azure Pipelines YAML file'. The 'Existing Azure Pipelines YAML file' option is selected, and a modal window titled 'Select an existing YAML file' is open. The modal shows a dropdown for 'Branch' set to 'master' and a text input for 'Path' containing '/scripts/Cl.yml'. At the bottom of the modal are 'Cancel' and 'Continue' buttons. The URL in the browser is https://dev.azure.com/fkristiansen/myfirstapp/_apps/hub/ms.vss-build-web.ci-designer-hub?sourceProvider=TfsGit&repositoryId=d227ed28-f8b....

In the **Review your Pipeline YAML**, click the arrow next to the **Run** button and **Save the pipeline**.



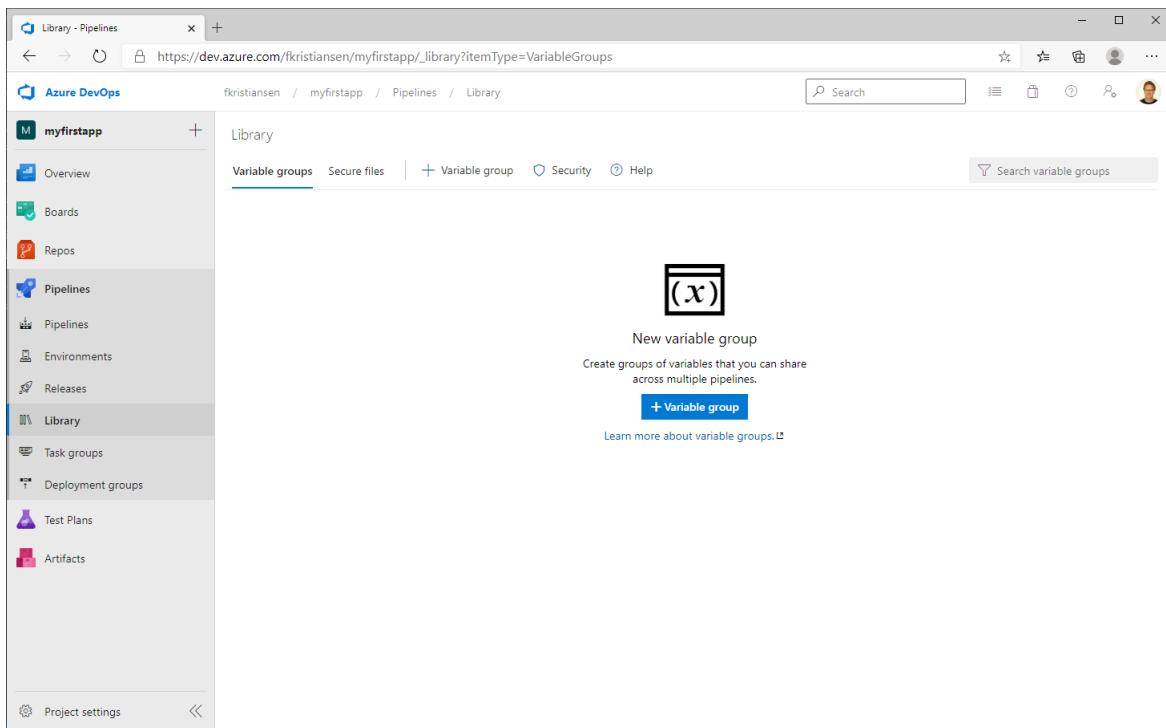
```
trigger:
- master

pool:
  name: 'default'

variables:
- group: BuildVariables

jobs:
- job: Build
  timeoutInMinutes: 300
  variables:
    build.clean: all
    platform: x64
    version: 'ci'
    appVersion: '2.0'
    appBuild: $(Build.BuildID)
    appRevision: 0
    createRuntimePackages: True
    skipComponentGovernanceDetection: True
  steps:
    - task: PowerShell@2
      displayName: 'Set BuildNumber'
      inputs:
        targetType: inline
        script: 'Write-Host "#vso[build.updatebuildnumber]$(appVersion).$(appBuild).$(appRevision)"'
```

The pipeline needs access to your Key Vault secrets. Navigate to Library under Pipelines:



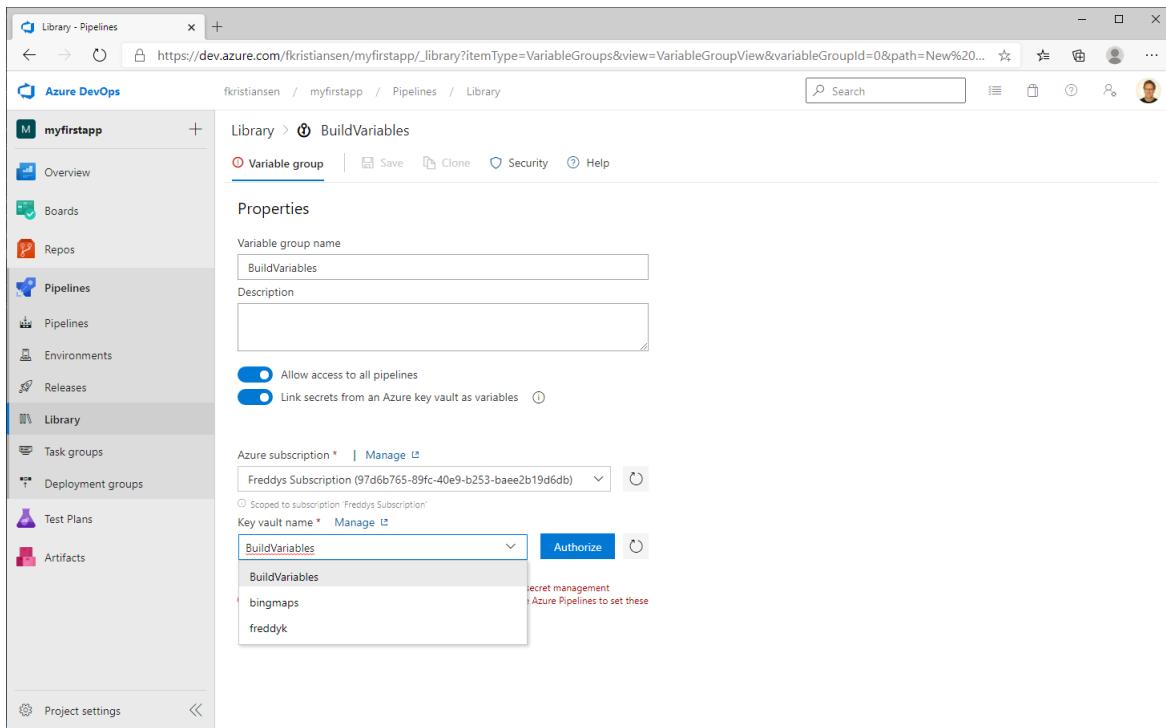
Variable groups

New variable group

Create groups of variables that you can share across multiple pipelines.

+ Variable group

Create a variable group called BuildVariables. Link Secrets from an Azure Key Vault as variables and authorize the pipeline to access your subscription and your Key Vault:



Variable group name: BuildVariables

Description:

Allow access to all pipelines

Link secrets from an Azure key vault as variables

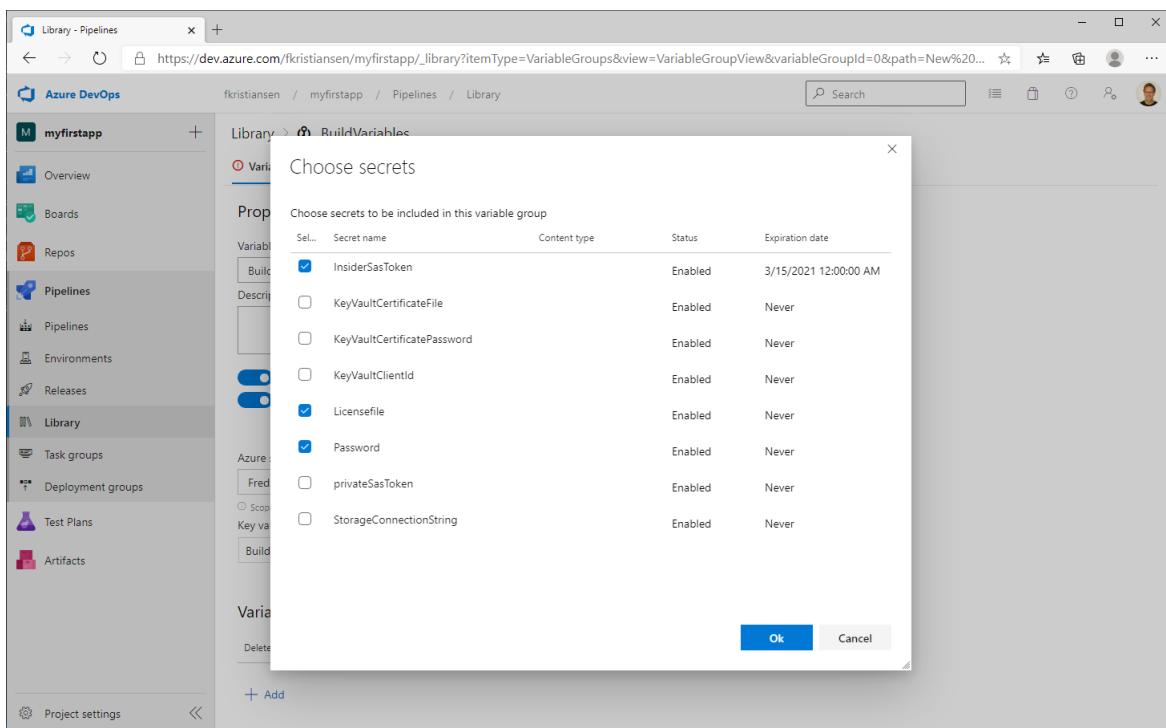
Azure subscription: Freddy's Subscription (97d6b765-89fc-40e9-b253-baee2b19d6db)

Key vault name: BuildVariables

bingmaps

freddyk

After that, add the variables needed by the pipeline:

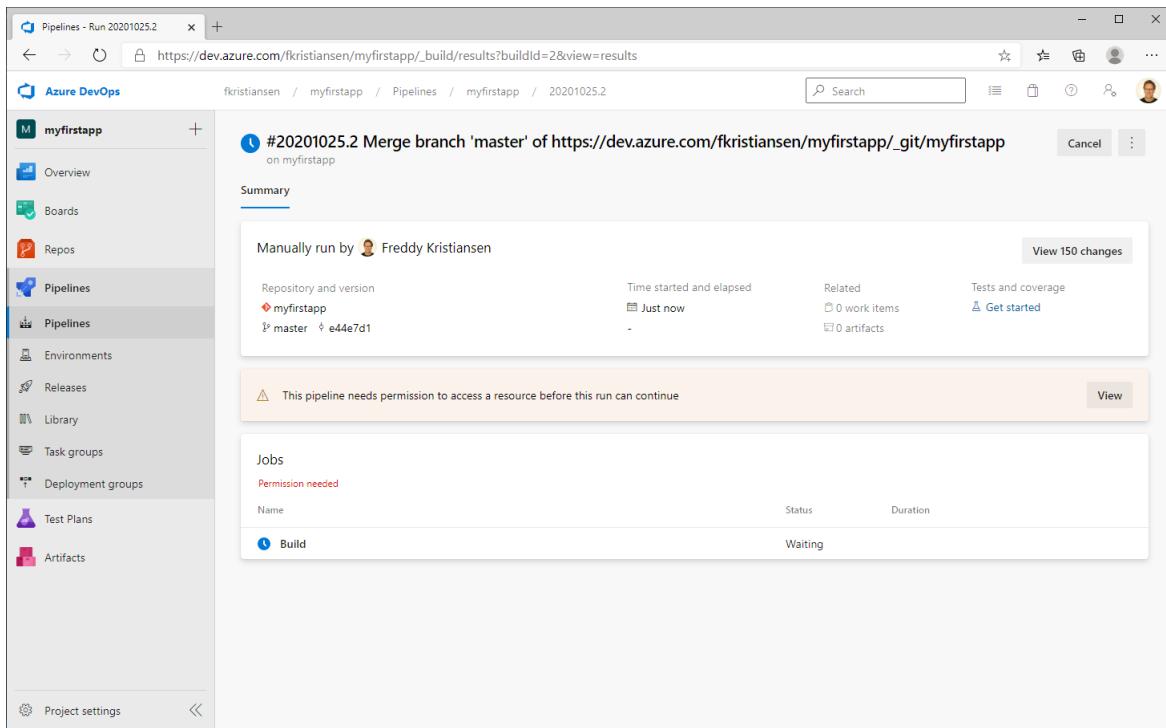


Secret name	Content type	Status	Expiration date
InsiderSasToken		Enabled	3/15/2021 12:00:00 AM
KeyVaultCertificateFile		Enabled	Never
KeyVaultCertificatePassword		Enabled	Never
KeyVaultClientId		Enabled	Never
Licensefile		Enabled	Never
Password		Enabled	Never
privateSasToken		Enabled	Never
StorageConnectionString		Enabled	Never

Click Ok and Save!

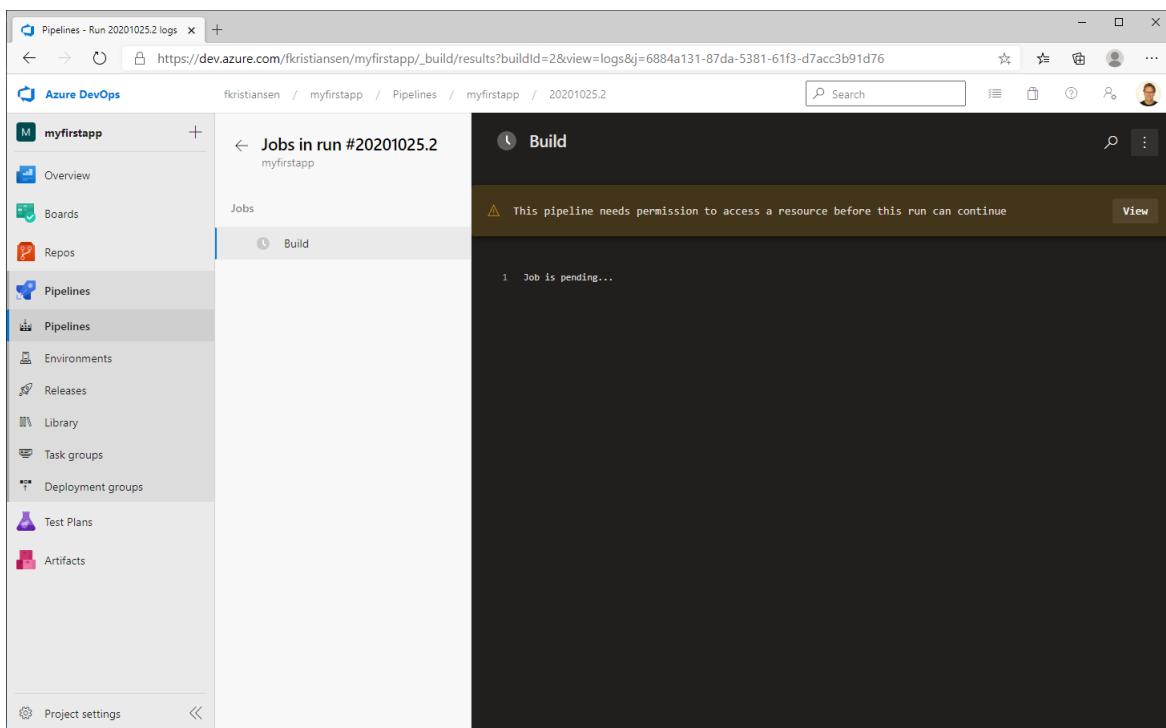
Go back to pipelines, click your pipeline and select **Run Pipeline**.

The pipeline will ask for permissions to access a resource (your Key Vault) before running. This can either be in the pipeline window, like here:



A screenshot of the Azure DevOps Pipelines summary page. The pipeline is named 'myfirstapp' and is currently 'Waiting'. The summary table shows the repository as 'myfirstapp' on the 'master' branch, started 'Just now', with 0 work items and 0 artifacts. A note indicates that the pipeline needs permission to access a resource. The 'Jobs' section shows a single 'Build' job listed.

or inside the actual build window, like here:



A screenshot of the Azure DevOps Build window for run #20201025.2. The build is named 'Build' and is currently 'pending'. A note indicates that the pipeline needs permission to access a resource. The build window shows a single job entry.

Click view and grant permissions.

Follow the progress of the pipeline by clicking the pipeline:

```

83 Test application folders
84 - D:\a\1\s\test
85
86
87
88
89
90
91
92
93
94
95 Pulling mcr.microsoft.com/dynamicsnav:10.0.17763.1457-generic
96
97 Pulling generic image took 167 seconds
98
99
100
101
102
103
104
105
106
107
108 WARNING: Container name should not exceed 15 characters
109 BeContainerHelper is version 1.0.11
110 BeContainerHelper is running as administrator
111 Host is Microsoft Windows Server 2019 Datacenter - Itsc2019
112 Docker Client Version is 19.03.12
113 Docker Server Version is 19.03.12
114 Downloading application artifact /sandbox/17.0.17126.17791/us
115 Downloading C:\Users\VssAdministrator\AppData\Local\Temp\32076f5c-1535-471b-a7a0-4370af5ae286.zip

```

The Run Pipeline is the task, which basically performs the same steps as you did when running the local devenv pipeline earlier and you will see similar output.

```

231
232 Files:
233 http://hostedagent-myapp-ci:8080/Alllanguage.vsix
234
235 Container Total Physical Memory is 7.0Gb
236 Container Free Physical Memory is 2.7Gb
237
238 Initialization took 88 seconds
239 Ready for connections!
240 Reading CustomSettings.config from hostedagent-myapp-ci
241 Creating Desktop Shortcuts for hostedagent-myapp-ci
242 Container hostedagent-myapp-ci successfully created
243
244 Use:
245 Get-BcContainerEventLog -containerName hostedagent-myapp-ci to retrieve a snapshot of the event log from the container
246 Get-BcContainerDebugInfo -containerName hostedagent-myapp-ci to get debug information about the container
247 Enter-BcContainer -containerName hostedagent-myapp-ci to open a PowerShell prompt inside the container
248 Remove-BcContainer -containerName hostedagent-myapp-ci to remove the container again
249 docker logs hostedagent-myapp-ci to retrieve information about URL's again
250
251 Creating container took 759 seconds
252
253
254
255
256
257
258
259
260
261
262 Using Version 1.0.2.0
263 Using Symbols Folder: D:\a\1\s\packages

```

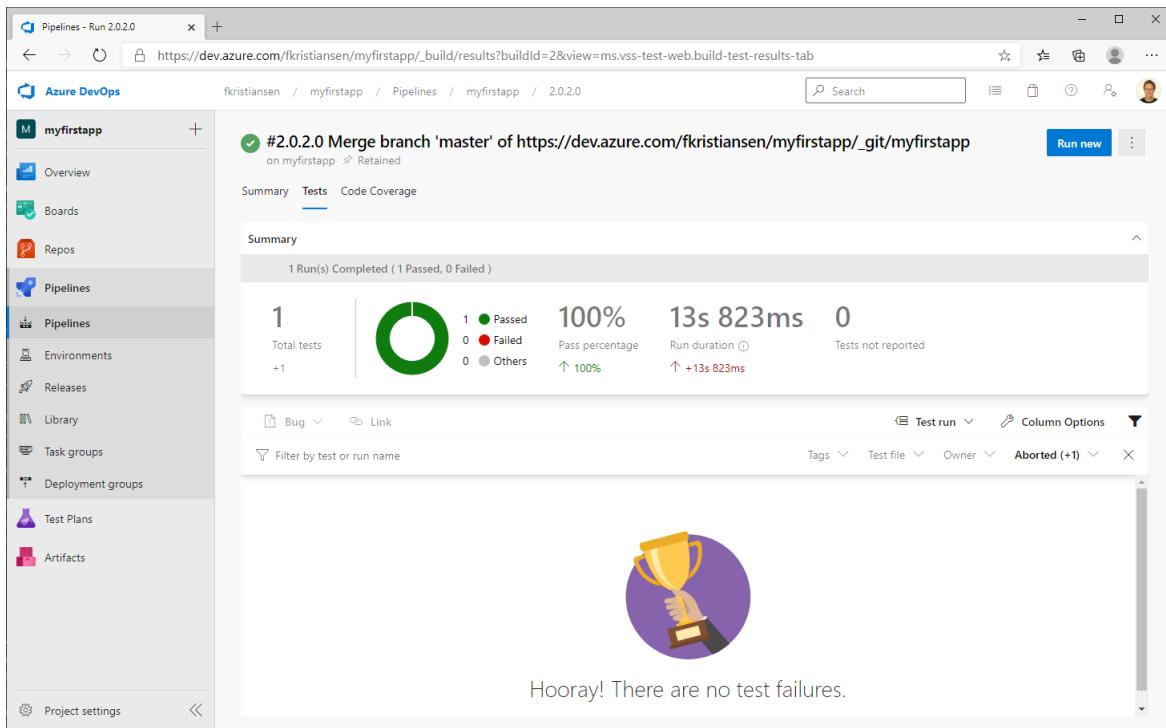
The pipeline is by default setup to use Azure Hosted Agents, which will take approx. 20 minutes to complete a build and nothing can really be reused between builds.

You can also setup your own build agents (either on an Azure VM or a local computer) for use with the build, which will greatly increase speed but will also come with a cost of running and maintaining that machine.

When the build is complete, you should see:

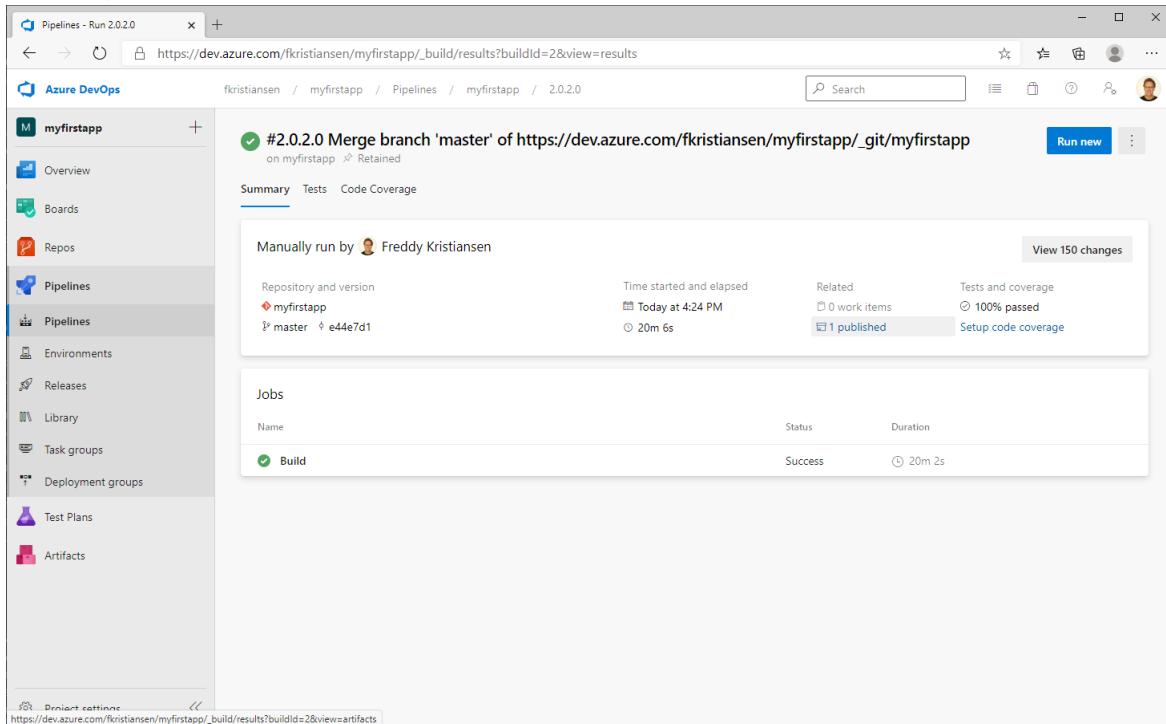
And clicking **view raw log** on the **Run Pipeline** will give you a nice output of the pipeline, which also can be used to search for into, and should always be included when creating issues on Run-Pipeline.

You can inspect the test results:



The screenshot shows the Azure DevOps Test Results page for a build. The summary indicates 1 test completed (1 Passed, 0 Failed, 0 Others), with a 100% pass percentage and a run duration of 13s 823ms. A trophy icon is displayed, and the message "Hooray! There are no test failures." is shown.

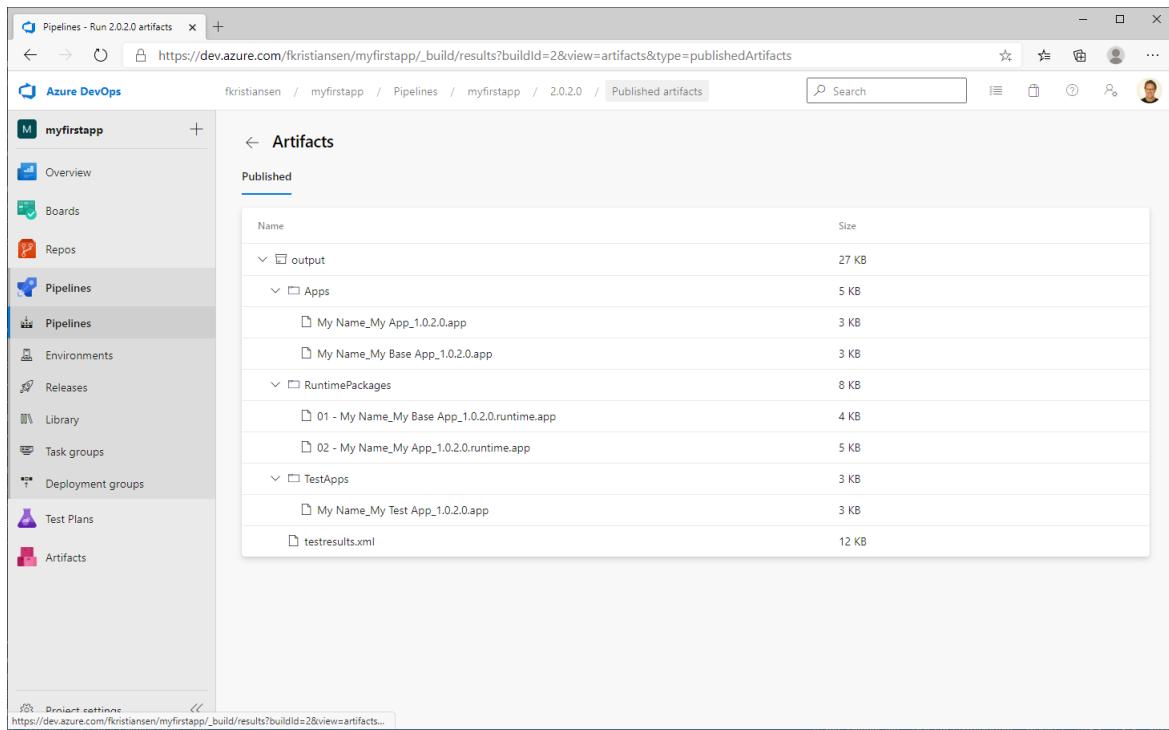
And in the build summary, you can download the published app artifacts (not to be confused with Business Central artifacts):



The screenshot shows the Azure DevOps Build Summary page for a successful build. It displays the repository and version (myfirstapp, master, e44e7d1), the time started and elapsed (Today at 4:24 PM, 20m 6s), and the tests and coverage (100% passed). The Jobs section shows a successful Build job. The URL at the bottom is https://dev.azure.com/fkristiansen/myfirstapp/_build/results?buildId=2&view=artifacts.

The app is available as app and runtime package. The test app and the test results are also published.

Note that the runtime packages are prefixed with a number series, which indicates the order in which they should be installed (dependencies first), as Sort-AppFilesByDependencies doesn't work on runtime packages.



The screenshot shows the Azure DevOps interface for a project named 'myfirstapp'. The left sidebar is a navigation menu with the following items: Overview, Boards, Repos, Pipelines (selected), Pipelines, Environments, Releases, Library, Task groups, Deployment groups, Test Plans, and Artifacts. The main content area is titled 'Artifacts' and shows a table of published artifacts for build 2.0.0.0. The table has two columns: 'Name' and 'Size'. The artifacts listed are:

Name	Size
output	27 KB
Apps	5 KB
My Name_My App_1.0.2.0.app	3 KB
My Name_My Base App_1.0.2.0.app	3 KB
RuntimePackages	8 KB
01 - My Name_My Base App_1.0.2.0.runtime.app	4 KB
02 - My Name_My App_1.0.2.0.runtime.app	5 KB
TestApps	3 KB
My Name_My Test App_1.0.2.0.app	3 KB
testresults.xml	12 KB

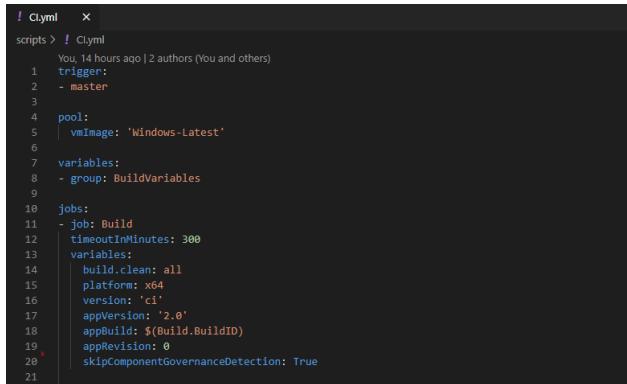
Congratulations – you have run your first Build pipeline.

Inspect the pipeline

Let's have a look at the pipeline and some of the steps.

Initialization

In CI.yml you have some settings and variables at the start:



```
! CI.yml
scripts > ! CI.yml
You: 14 hours ago | 2 authors (You and others)
1 trigger:
2 - master
3
4 pool:
5 | vImage: 'Windows-Latest'
6
7 variables:
8 - group: BuildVariables
9
10 jobs:
11 - job: Build
12 | timeoutInMinutes: 300
13 | variables:
14 |   build.clean: all
15 |   platform: x64
16 |   version: 'ci'
17 |   appVersion: '2.0'
18 |   appBuild: ${Build.BuildID}
19 |   appRevision: 0
20 |   skipComponentGovernanceDetection: True
21
```

Trigger: master means that the pipeline will trigger whenever a change is made to the master branch.

Pool: vImage: 'windows-latest' means that the pipeline will use Azure Hosted agents of the latest Windows version. Change this to **name: 'default'** if you want to use self-hosted agents in the default pool. (You can use <https://aka.ms/getbuildagent> to create a self-hosted agent and add it to the default pool).

Variables: group: buildVariables will request access to the variable group buildVariables and make the variables defined in that available in the pipeline.

timeoutInMinutes: 300 determines that the build will run in up to 300 minutes before timing out.

build.clean: all means that DevOps will clean all symbols caches and binary folders before every build – do not attempt to reuse anything.

platform: x64 indicates that we want to run 64 bit mode (not 32 bit x86) **pool: name: Default** says that we want to use a build agent from the Default pool (in which we places our agent)

version: 'ci' determines which settings from **settings.json** to use. The settings points out which Business Central artifacts to use for the build, which apps to install, the App Folders, the Test Folders, which test framework to install, which cops to enable etc. etc.

AppVersion, AppBuild and AppRevision determines the version numbering. There are a lot of different ways to make version numbering of your app. I have implemented one very simple model, where the three variables **appVersion**, **appBuild** and **appRevision** will be combined into a version number of the build. **appBuild** is set to the **Build_BuildID** which is a unique build ID (auto incrementing) for this Organization.

The individual apps compiled by the pipeline will get the same **appBuild** and **appRevision**, but will keep their **appMajor** and **appMinor** from **app.json**. This allows for dependency apps to be compiled from various pipelines and get unique build numbers.

Steps

Set BuildNumber

Inline PowerShell script to set the build number for Azure DevOps.

Run Pipeline

Invoke the function DevOps-Pipeline to perform the actual pipeline. InsiderSasToken, LicenseFile and code signing certificate are transferred from variable group to function in environment variables as the build agent won't have

access to key vaults, the pipeline does. AppBuild and AppRevision are transferred as parameters together with version, which indicates whether this should be a ci build, a current build, a nextminor build or a nextmajor build.

The DevOps-Pipeline will also check that the right version of BcContainerHelper is installed and imported and it will read the settings file to determine the right settings for invoking the Run-APipeline in BcContainerHelper

[Publish Test Results](#)

Publish the test results, which are saves in XML files by the pipeline. The format used for the test output is Junit.

[Publish Artifacts](#)

Publish the build artifacts (not to be confused with artifacts for running a container) to devops. This is the build result. The apps, the runtime packages, the test apps and the test results.

[Cleanup](#)

Cleanup the environment by invoking cleanup.ps1. When using hosted agents, this is really not necessary, they get cleaned up automatically. When using self-hosted agents, the cleanup function will remove containers, artifacts and images left over after failed builds. Artifacts and images, which haven't been used for 2 days are removed.

Settings

The settings file in the scripts folder determines the settings for running the pipeline. The file is read and used only by the Read-Settings file, which reads the file and sets a number of variables based on settings.json. Read-Settings takes a version parameter indicating which version to build (ci, current, nextminor or nextmajor). Settings are read from the specific version section if available, else from the main section.

This means that if you want to run **AppSourceCop** in all versions but the **ci** version – you would set this to **true in the main section and false in the ci section**.

[Read-Settings](#) will set the following variables

agentName is blank if running local or set to \$ENV:AGENT_NAME if running a devops agent.

pipelineName is the name of the pipeline, including the version (e.g. HelloWorld-ci)

containerName is set to the name of the build container. The agent name is included in the name if running a devops agent.

The following list are the settings, which will be turned into variables with the same name

installApps is a comma separated list of secure url's to dependencies (apps or .zip files containing apps) to be installed before compiling apps and test apps. The apps can be apps or runtime packages. If they are apps, they are sorted after dependencies before installing. Runtime packages are sorted and installed alphabetically.

previousApps is a comma separated list of secure url's to previous versions of the apps (or .zip files containing previous versions of the apps) to be used as previous versions of apps for **AppSourceCop** breaking change detection and upgrade test. When **previousApps** are specifying, these apps are published and installed before the newly build apps and upgrade is run before the tests.

appFolders is a comma separated list of folders which should be compiled as apps. The folders will be sorted after dependencies before compiled, published and installed. Apps in app folders are compiled before the test framework/libraries are published/installed and they are signed (if signing certificate is specified) unless doNotSignApp is set to true.

appSourceCopMandatoryAffixes is a comma separated list of affixes to be used as mandatory affixes in AppSourceCop settings.

appSourceCopSupportedCountries is a comma separated list of supported countries to be used as supported countries in AppSourceCop settings.

testFolders is a comma separated list of folders which should be compiled and used as test apps. The folders will be sorted after dependencies before compiled, published and installed. Apps in test folders are compiled after the test framework/libraries are published/installed and tests in these apps are used for test execution.

memoryLimit determines the amount of memory available in the build container during the pipeline run.

additionalCountries is a comma separated list of country codes. During the pipeline, a container with this local version of Business Central will be spun up and the apps produced will be published, installed and tested. Test results will be gathered for all versions. Default is no additional countries.

genericImageName is the generic image name to use for creating the container. Default is the default generic image in BcContainerHelper configuration.

vaultNameForLocal is the name of the key vault to use for secrets like licensefile, insiderSasToken and passwords. This setting is only used in **Local-DevEnv.ps1** and **Local-Pipeline.ps1**.

bcContainerHelperVersion determines which version of BcContainerHelper to use. Latest is the default setting, which probably is fine for most. Preview means grab the latest preview version and a specific version number will grab that exact version from the PowerShell Gallery. This setting can also be a local path on the build agent or a URL to a github repository, where the desired version of the containerhelper can be downloaded.

installTestFramework is a Boolean setting determining whether to install the Test Framework before compiling the test apps.

installTestLibraries is a Boolean setting, determining whether to install the Test Libraries before compiling the test apps. Test Libraries includes the Test Framework.

installPerformanceToolkit is a Boolean setting, determining whether to install the Performance Toolkit before compiling the test apps. Performance toolkit include the Test Framework.

enableCodeCop is a Boolean setting, determining whether Code Cop is enabled.

enableAppSourceCop is a Boolean setting, determining whether AppSource Cop is enabled.

enablePerTenantExtensionCop is a Boolean setting, determining whether Per Tenant Extension Cop is enabled.

enableUICop is a Boolean setting, determining whether UI Cop is enabled.

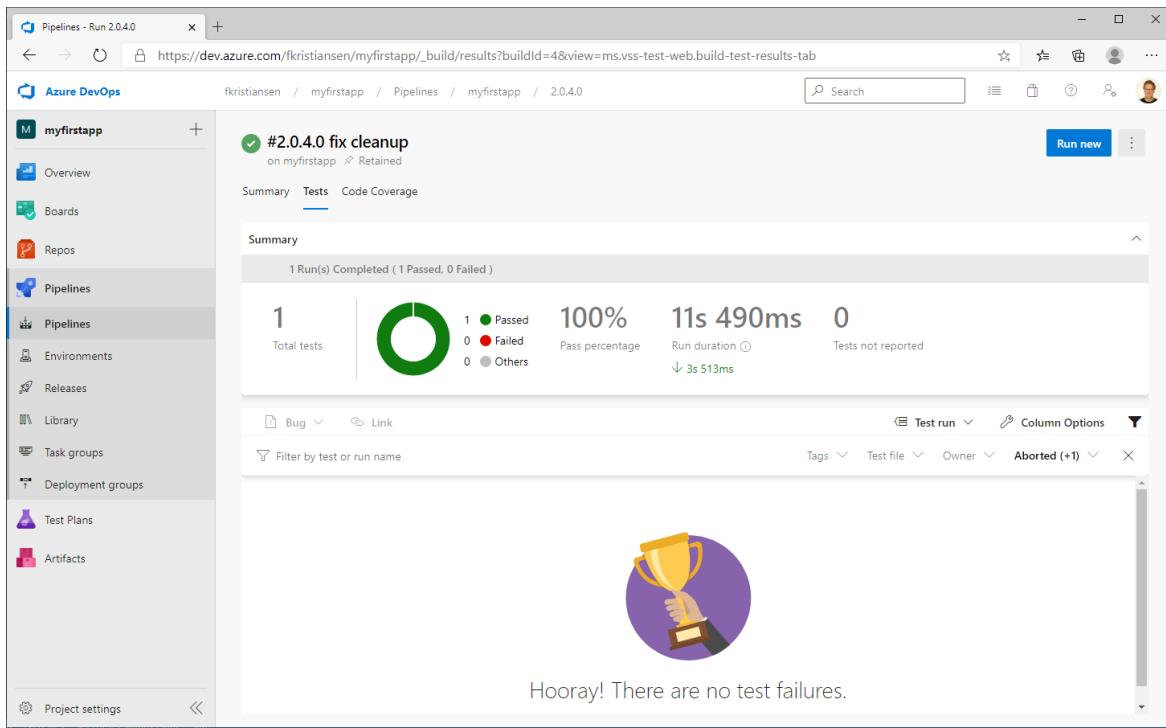
doNotSignApps is a Boolean setting which can be set to true if you do not want to sign Apps.

doNotRunTests is a Boolean setting, which can be set to true if you do not want to run Tests.

cachelImage is a Boolean setting, which determines whether an image will be cached before creating the build container. By default cachelImage is set to true on ci pipelines which is typically reusing the same Business Central version and false on other pipelines as versions changes a lot.

Publishing Test Results

The **Publish Test Results** step will publish the **JUnit** compatible test results file to **Azure DevOps**, giving you the opportunity to investigate failing tests, see stack traces and creating work item for fixing failing tests.



Pipelines - Run 2.0.4.0

https://dev.azure.com/fkristiansen/myfirstapp/_build/results?buildId=4&view=ms.vss-test-web.build-test-results-tab

myfirstapp

#2.0.4.0 fix cleanup

on myfirstapp Retained

Run new

Summary Tests Code Coverage

Summary

1 Run(s) Completed (1 Passed, 0 Failed)

1 Total tests

1 Passed

0 Failed

0 Others

100% Pass percentage

11s 490ms Run duration

↓ 3s 513ms Tests not reported

Bug Link

Filter by test or run name

Test run Column Options

Tags Test file Owner Aborted (+1)

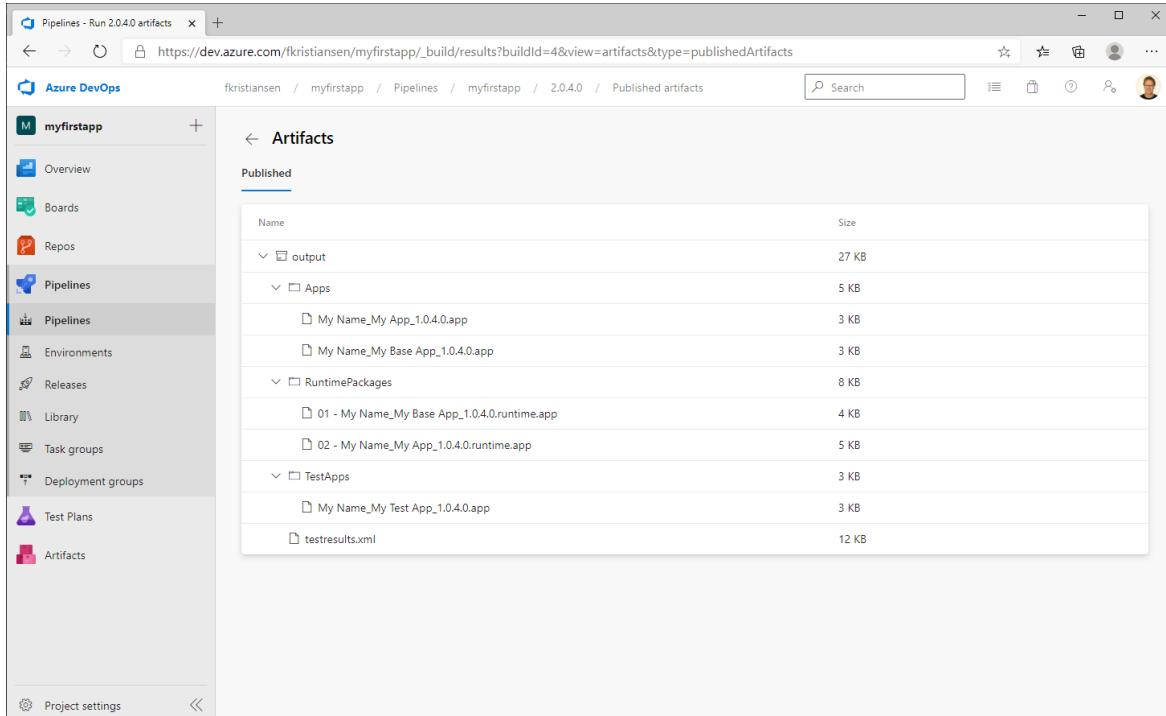


Hooray! There are no test failures.

Publish Artifacts

The **Publish Artifacts** step will publish the generated build artifacts to make them available for a release pipeline. The artifacts contains the **.app files generated (including tests)**, runtime packages of the same apps and the test results.

The artifacts are easily consumed by release pipelines.



Pipelines - Run 2.0.4.0 artifacts

https://dev.azure.com/fkristiansen/myfirstapp/_build/results?buildId=4&view=artifacts&type=publishedArtifacts

myfirstapp

Published

Name	Size
output	27 KB
Apps	5 KB
My Name_My App_1.0.4.0.app	3 KB
My Name_My Base App_1.0.4.0.app	3 KB
RuntimePackages	8 KB
01 - My Name_My Base App_1.0.4.0.runtime.app	4 KB
02 - My Name_My App_1.0.4.0.runtime.app	5 KB
TestApps	3 KB
My Name_My Test App_1.0.4.0.app	3 KB
testresults.xml	12 KB

Cleanup

The Cleanup step will invoke the **cleanup.ps1** script, which will remove the container (if present) and cleanup up artifacts cache and container images, which hasn't been used the last 2 days.

Create a Release Pipeline

I will be working with a few different release pipelines.

1. Two release pipelines for releasing the build artifacts to blob storage (preview and production). You can also use Azure DevOps Artifacts, but what I really like about blob storage is, that I can secure access to the artifacts as I decide and I can download the artifacts using a simple Url (no PowerShell commands with special Az modules needed). The preview release pipeline should be invoked after every successful build, the production one should run on demand.
2. Two release pipelines for releasing a Per Tenant Extension to an online environment for a customer to test in sandbox environment or run in production. The pipeline releasing to a sandbox environment could be setup to run after every successful build, the one releasing to production should run on demand.
3. **Later versions** of this workshop will also include how to **deploy the App directly to AppSource** and through that **to end-customers** who have installed your app.

Releasing to Blob Storage

The way I have structured the blob storage for my apps is:

<https://storageaccount/appname/version/apps.zip>

appname could be **bingmaps**, **bingmaps-preview**, **helloworld** or **helloworld-preview**. Version is either a **specific version number** or **latest**.

Examples:

<https://businesscentralapps.blob.core.windows.net/bingmaps/16.0.10208.0/apps.zip>

gives me version 16.0.10208.0 of the BingMaps app, and

<https://businesscentralapps.blob.core.windows.net/bingmaps-preview/latest/runtimepackages.zip>

gives me the runtime packages from the latest version of the bingmaps preview, and

<https://businesscentralapps.blob.core.windows.net/helloworld/latest/apps.zip>

gives me the latest production release of the helloworld app.

You might have noticed that I always use the latest production release of my apps as **previousapps** setting for AppSourceCop to get breaking change notification. So, let's setup the two release pipelines.

Create a Blob Storage and get the Connection String

First we need a Blob Storage and a connection string. The Blog Storage Account is created in the Azure Portal and under shared access signature you define the access requirements for the connection string and generate the connection string:

After pressing **Generate SAS and connection string**, you get three values:

Connection string
`BlobEndpoint=https://businesscentralapps.blob.core.windows.net/;QueueEndpoint=https://businesscentralapps.queue.core.windows.net/;FileEndpoint=https://businesscentralapps.file.core.windows.net/;SasToken=?sv=2019-12-12&ss=b&srt=sco&sp=rw&st=2020-10-26T10:59:00Z&se=2020-10-26T10:50:26Z&spr=https&sig=Z6OYznraRFPjITXN4obXdmxw0Bm...`

SAS token
`?sv=2019-12-12&ss=b&srt=sco&sp=rw&st=2020-10-26T10:59:00Z&se=2020-10-26T10:50:26Z&spr=https&sig=Z6OYznraRFPjITXN4obXdmxw0Bm...`

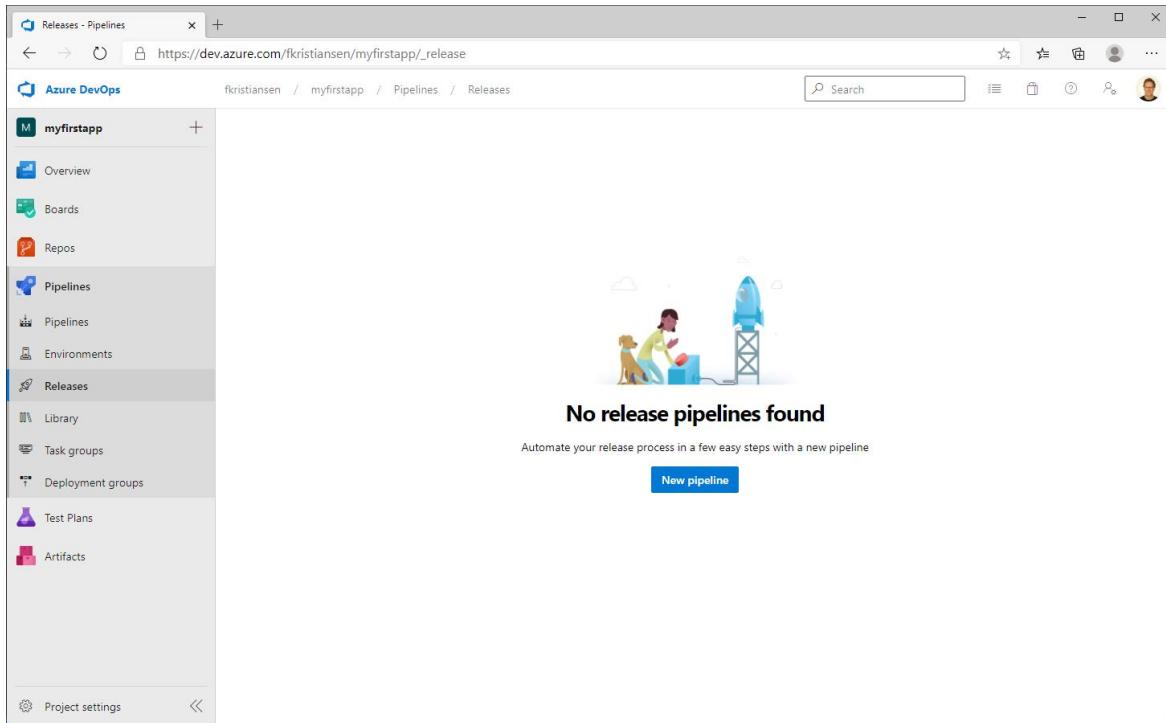
Blob service SAS URL
`https://businesscentralapps.blob.core.windows.net/?sv=2019-12-12&ss=b&srt=sco&sp=rw&st=2020-10-26T10:59:00Z&se=2020-10-26T10:50:26Z...`

Click the **Copy to Clipboard** next to the **Connection String** and add this as a secret in your key vault from the first section. In the library section under Pipelines, add the new secret to the secrets available to the pipelines, press **Ok** and **Save**.

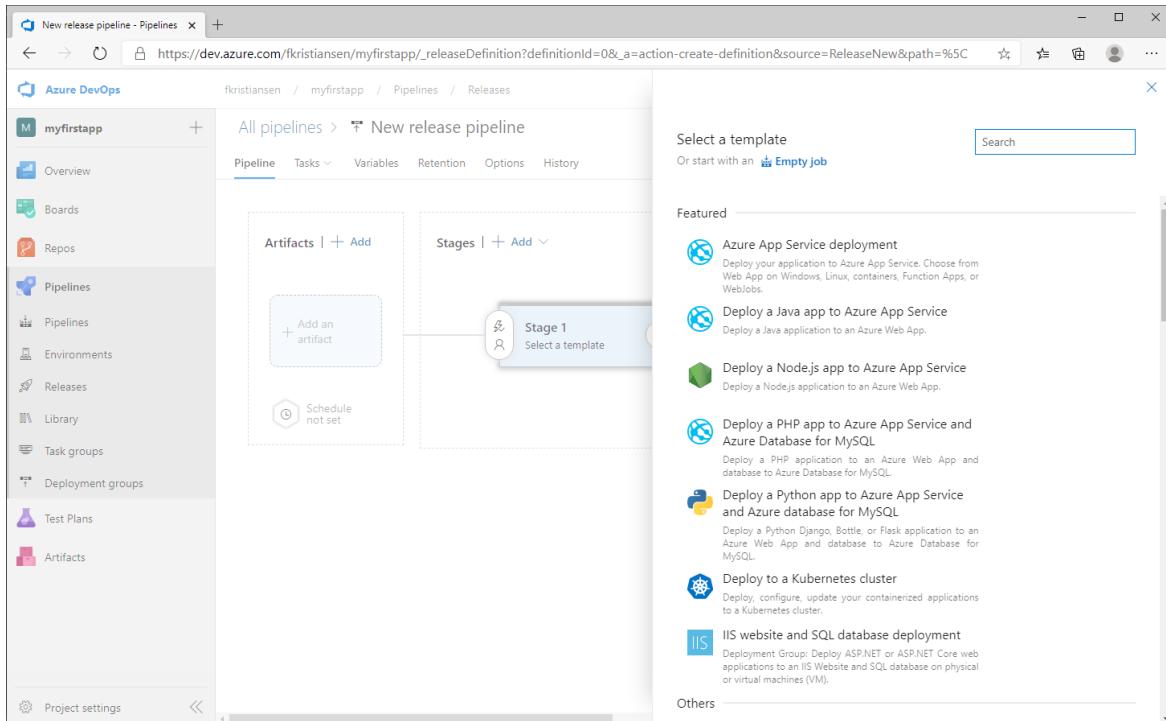
Secret name	Content type	Status	Expiration date
InsiderSasToken		Enabled	3/15/2021 12:00:00 AM
KeyVaultCertificateFile		Enabled	Never
KeyVaultCertificatePassword		Enabled	Never
KeyVaultClientId		Enabled	Never
Licensefile		Enabled	Never
Password		Enabled	Never
privateSasToken		Enabled	Never
StorageConnectionString		Enabled	Never

Creating the release Pipeline

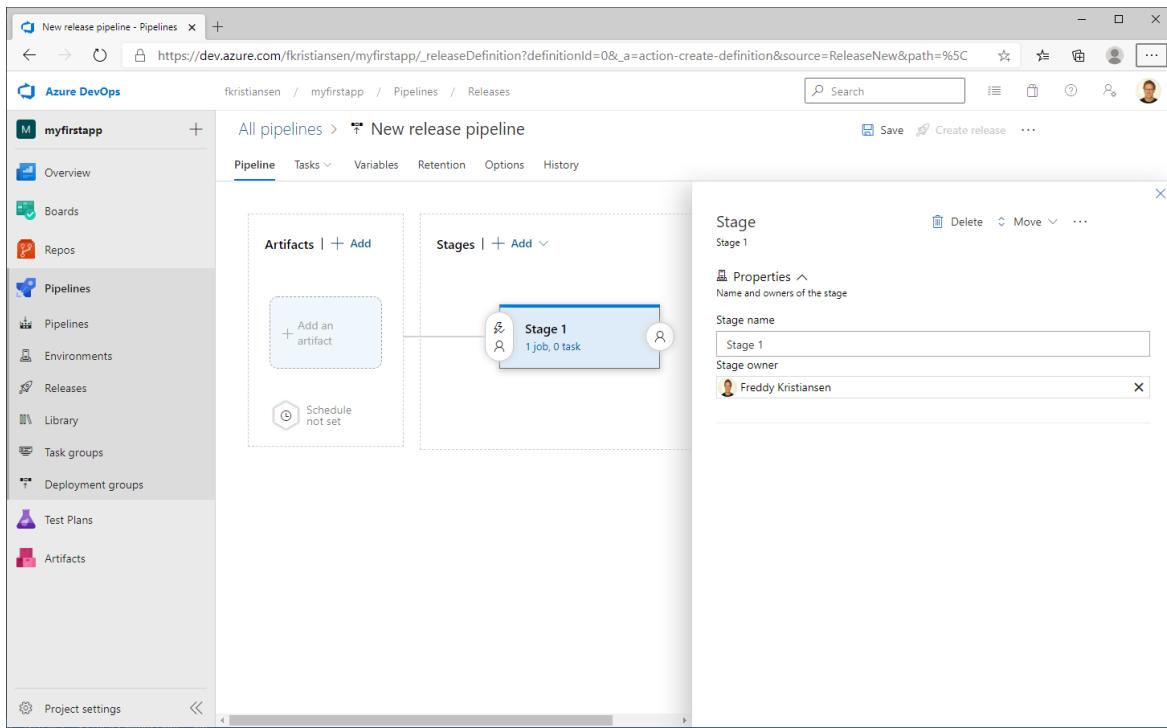
In Azure DevOps, under **Pipelines**, click **Releases** and select **New pipeline**.



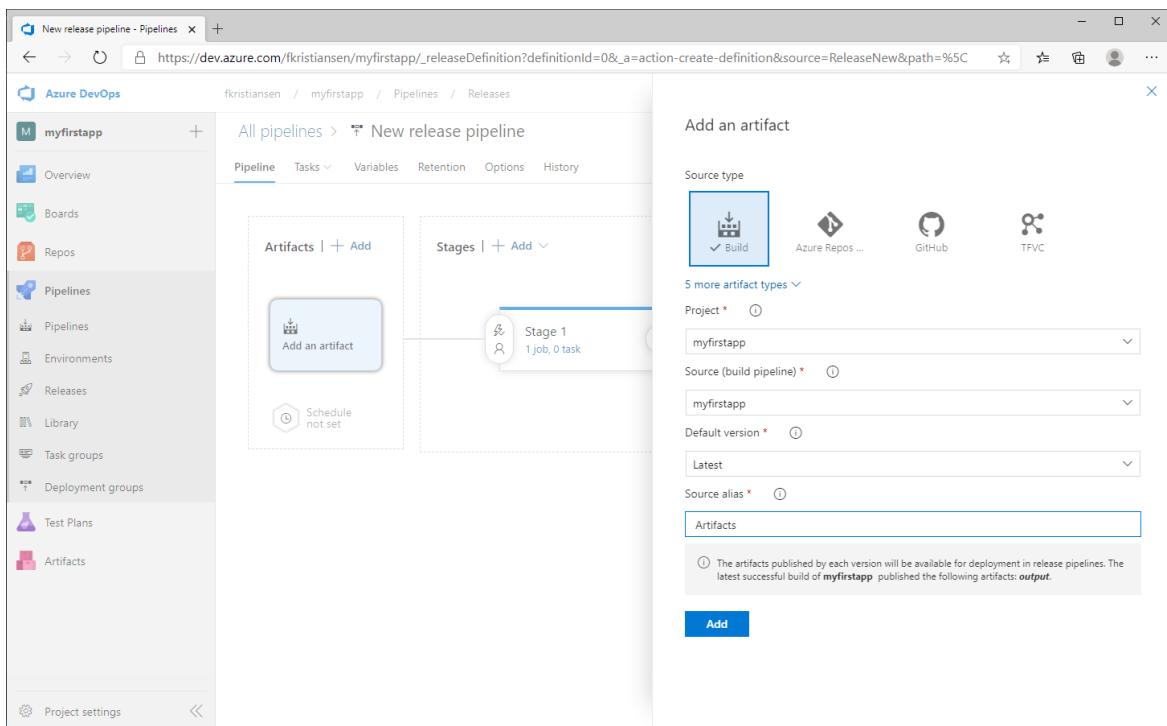
Select **Empty job** as template



Click **Add an artifact**



Use the **build pipeline** as source and name the artifacts **Artifacts**.



Click **Tasks** and click the **+** next to the **Agent Job**. Search for **PowerShell** and add a **PowerShell** task to the job.

The **PowerShell Script task** displays that some settings needs attention. Click the **PowerShell Script task**. Select **Inline** and Paste the following code into the script editor:

```
Write-Host "Installing BcContainerHelper"
Install-Module BcContainerHelper -Force

Write-Host "Publishing to Storage"
Publish-BuildOutputToStorage
  -storageConnectionString "$(StorageConnectionString)"
  -projectId "$($Env:BUILD_PROJECTNAME)-preview".ToLowerInvariant()
  -appVersion "$Env:BUILD_BUILDNUMBER"
  -path "Artifacts\output"
  -setLatest
```

Click **Variables**, **Variable Groups**, select **BuildVariables** and click **Link** to link the variable group to the release pipeline.

Set the name of the release Pipeline to **Release to Storage (preview)**. Click the small lightning bolt icon in artifacts to enable Continuous deployment trigger on the preview release pipeline.

Continuous deployment trigger
Build: Artifacts
Enabled
Creates a release every time a new build is available.

Build branch filters
Type: Build branch
Include: master
Build tags

Pull request trigger
Build: Artifacts
Disabled
Enabling this will create a release every time a selected artifact is available as part of a pull request workflow.

Click **Save**. After successfully saving the release pipeline, click **Create Release** to create a release and you should get your artifacts published to blob storage:

Name	Access Tier	Access Tier Last Modified	Last Modified	Blob Type	Content Type	Size	Status	Remaining Days	Deleted
app.zip	Hot (inferred)		10/26/2020, 1:44:57 PM	Block Blob	application/octet-stream	4.1 KB	Active		
runtimepackages.zip	Hot (inferred)		10/26/2020, 1:44:57 PM	Block Blob	application/octet-stream	8.0 KB	Active		
testapps.zip	Hot (inferred)		10/26/2020, 1:44:58 PM	Block Blob	application/octet-stream	2.5 KB	Active		

Now redo the entire process again, where you remove **-preview** in the script and replace **(preview)** with **(prod)** in the name and I can now use <https://businesscentralapps.blob.core.windows.net/myfirstapp/latest/apps.zip> as previousapps in my settings file.

Releasing a Per Tenant Extension to an online environment

To release per tenant extension apps to an online environment, we will use the service-to-service authentication for automation APIs, which was shipped in Business Central 2020 release wave 2 (v17). AJ did a very detailed description of this feature on his blog: <https://www.kauffmann.nl/2020/09/14/service-to-service-authentication-for-automation-apis-in-business-central/>.

The partner creates an AAD Application for authentication. The Customer registers the partners AAD Application in their Business Central tenant and assigns permissions that the app can do automation and extension management.

So, we need to setup a few things before we can create a release pipeline:

- An AAD Application for authenticating to Business Central
- An online tenant of Business Central v17 or higher
- Permissions in Business Central to allow the AAD App to do Automation and Extension Management

An AAD Application for authenticating to Business Central

Every partner will probably create just one multitenant AAD Application, which they will use for managing extensions or their customers, so I am not going to create a script for this (not at this time 😊)

Open the Azure Portal (<https://portal.azure.com>), search for **App Registrations**, create a **New App Registration**. Give your app a **friendly display name**, set the app to **multitenant**, set the **redirect URI** to <https://businesscentral.dynamics.com/OAuthLanding.htm>. Click Register.

Microsoft Azure (Preview) Report a bug Search resources, services, and docs (G+)

Home > App registrations > Register an application

Name
The user-facing display name for this application (this can be changed later).
Freddys App Publisher App

Supported account types
Who can use this application or access this API?
 Accounts in this organizational directory only (Microsoft only - Single tenant)
 Accounts in any organizational directory (Any Azure AD directory - Multitenant)
 Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)
 Personal Microsoft accounts only

Help me choose...

Redirect URI (optional)
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.
Web

By proceeding, you agree to the Microsoft Platform Policies [?](#)

Register

Make a copy the Application (Client ID).

Microsoft Azure (Preview) Report a bug Search resources, services, and docs (G+)

Home > App registrations > **Freddys App Publisher App**

Overview

Essentials

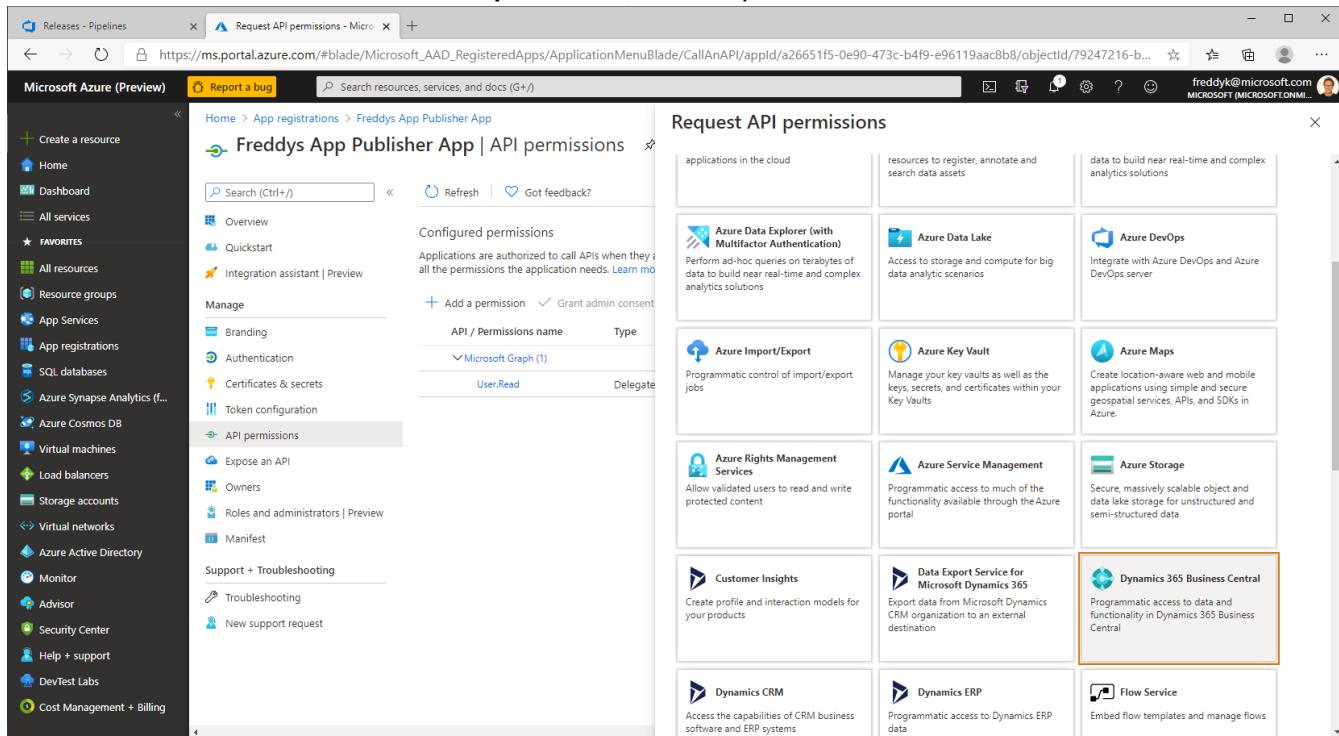
Display name : **Freddys App Publisher App** Supported account types : **Multiple organizations**
 Application (client) ID : **a26651f5-0e90-473c-b4f9-e96119aac8b8** Redirect URIs : **1 web, 0 spa, 0 public client**
 Directory (tenant) ID : **72f988bf-86f1-41af-91ab-2d7cd011db47** Application ID URI : **Add an Application ID URI**
 Object ID : **79247216-b293-435b-81af-39fe4eb6a4c** Managed application in L... : **Freddys App Publisher App**

Call APIs

Documentation

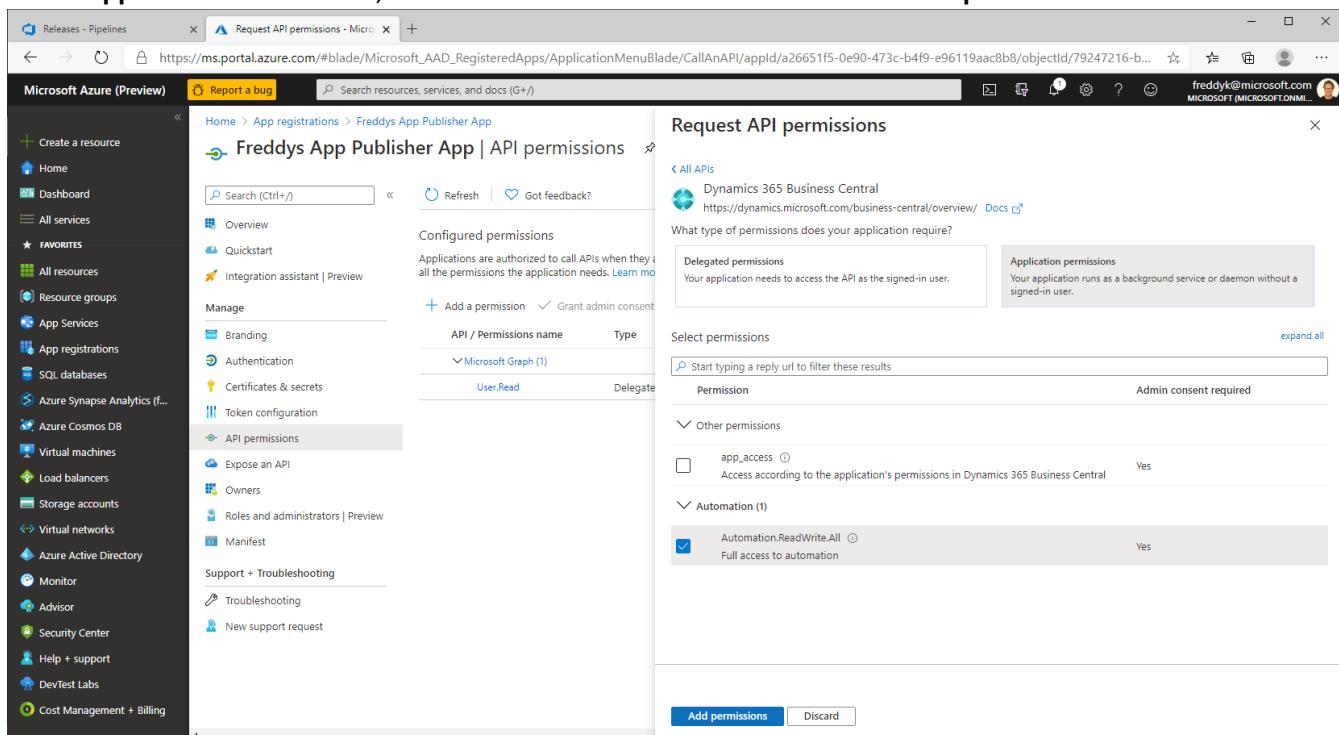
Microsoft identity platform
 Authentication scenarios
 Authentication libraries
 Code samples
 Microsoft Graph
 Glossary
 Help and Support

Click API Permissions and select + Add a permission, locate Dynamics 365 Business Central and select that.



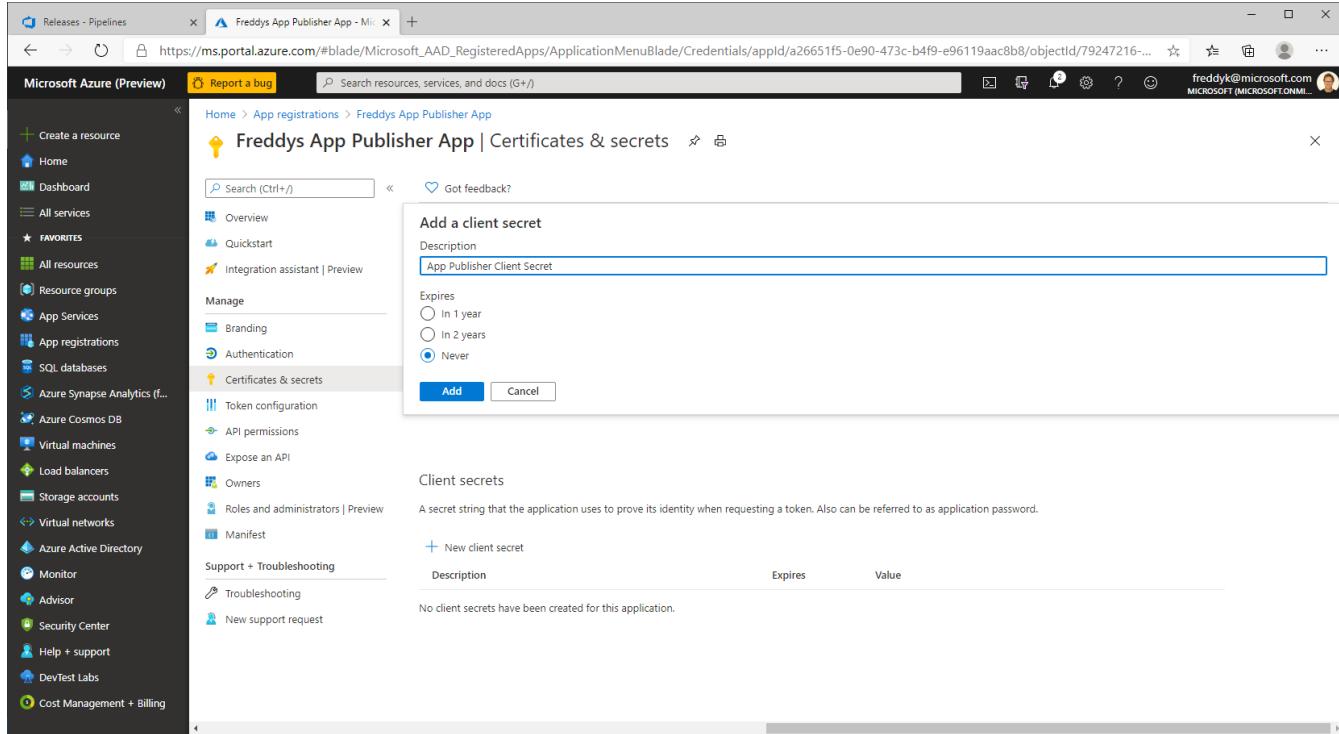
The screenshot shows the 'Request API permissions' page for the 'Freddys App Publisher App'. The left sidebar shows various Azure services. The main area shows 'Configured permissions' and a list of available APIs. The 'Dynamics 365 Business Central' option is highlighted with an orange box.

Select Application Permissions, check Automation.ReadWrite.All and click Add permissions



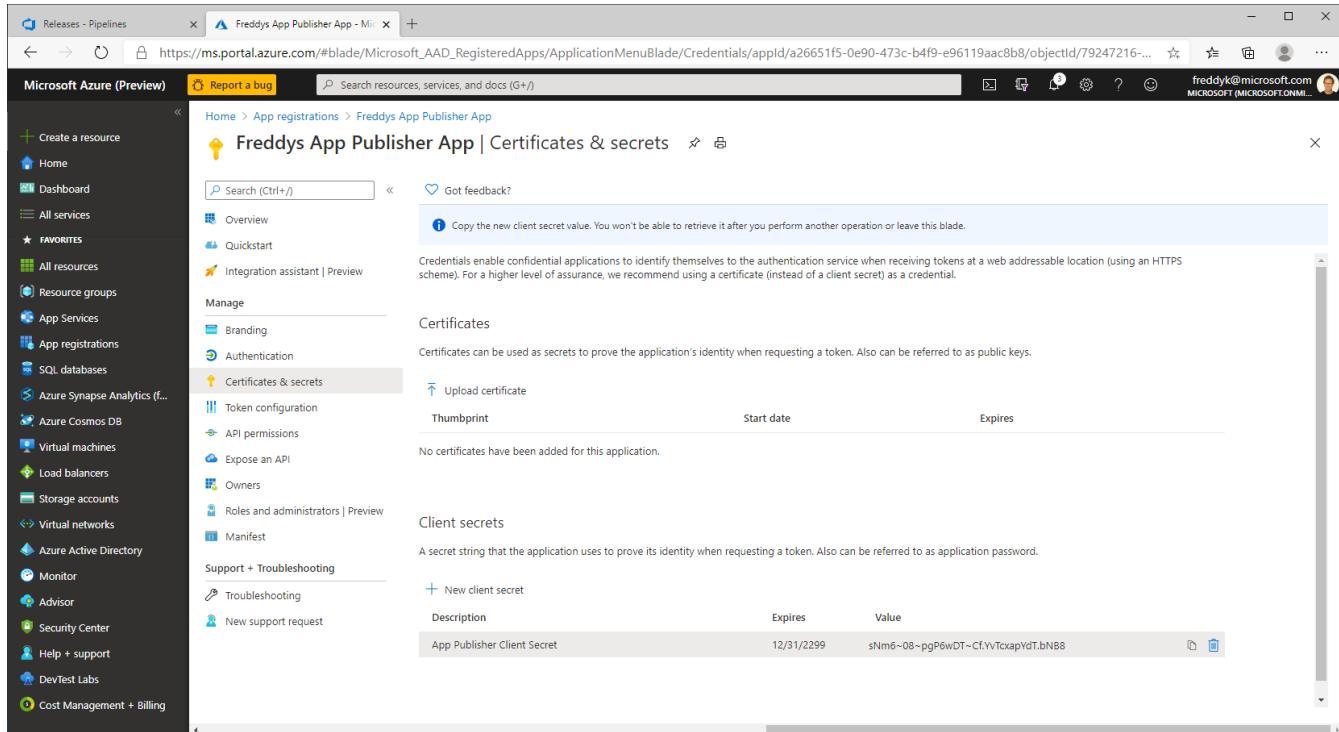
The screenshot shows the 'Request API permissions' page for the 'Freddys App Publisher App'. The left sidebar shows various Azure services. The main area shows 'Configured permissions' and a list of available APIs. The 'Automation.ReadWrite.All' permission is selected and highlighted with a grey box.

Select Certificates & secrets, click + New client secret, give it a name, set the expiration date and click Add



The screenshot shows the Microsoft Azure portal with the URL https://ms.portal.azure.com/#blade/Microsoft_AAD_RegisteredApps/ApplicationMenuBlade/Credentials/appId/a26651f5-0e90-473c-b4f9-e96119aac8b8/objectId/79247216-.... The page title is 'Freddys App Publisher App | Certificates & secrets'. The left sidebar shows various Azure services. The main content area is titled 'Add a client secret' with a description input field containing 'App Publisher Client Secret'. Below it, an 'Expires' section has three radio buttons: 'In 1 year', 'In 2 years', and 'Never' (which is selected). At the bottom are 'Add' and 'Cancel' buttons. To the right, there is a section titled 'Client secrets' with a note: 'A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as application password.' Below this is a table with a single row: '+ New client secret' (button), 'Description' (input: 'App Publisher Client Secret'), 'Expires' (input: '12/31/2299'), and 'Value' (input: empty).

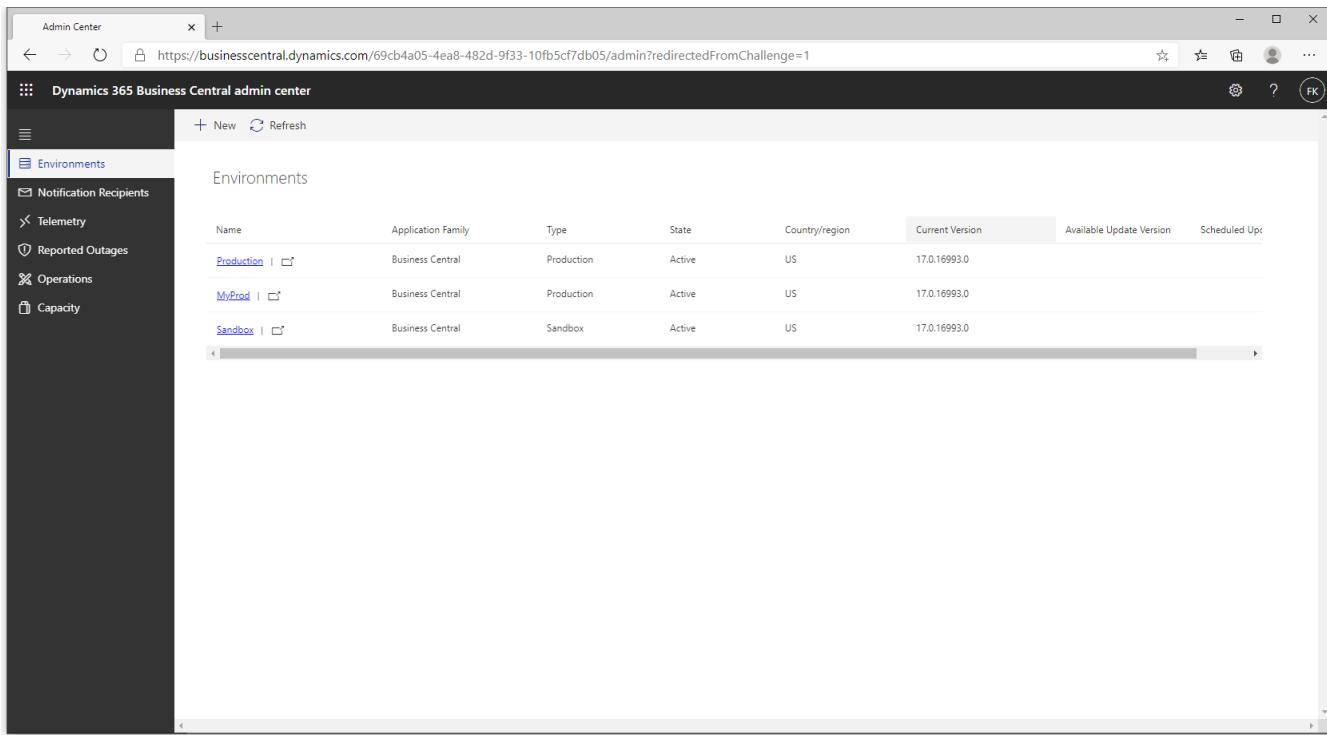
Make a Copy of the Client Secret, it is only shown once



The screenshot shows the Microsoft Azure portal with the same URL and page title as the previous screenshot. The 'Certificates & secrets' blade is displayed. A tooltip at the top right of the table says: 'Copy the new client secret value. You won't be able to retrieve it after you perform another operation or leave this blade.' Below the tooltip, there is a note: 'Credentials enable confidential applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (instead of a client secret) as a credential.' The 'Certificates' section is shown with a table header: 'Upload certificate', 'Thumbprint', 'Start date', and 'Expires'. The 'Client secrets' section shows the newly added secret with the details: 'App Publisher Client Secret', '12/31/2299', and 'sNm6-08~pgP6wDT-CfVvTcxapYdT.bNB8'.

Online tenant of Business Central

Next thing we need is an online tenant of Business Central, and we need to assign permissions to the partner AAD Application. In my admin center, I have created a production environment (MyProd) and a Sandbox environment (Sandbox), which I will use for this workshop.

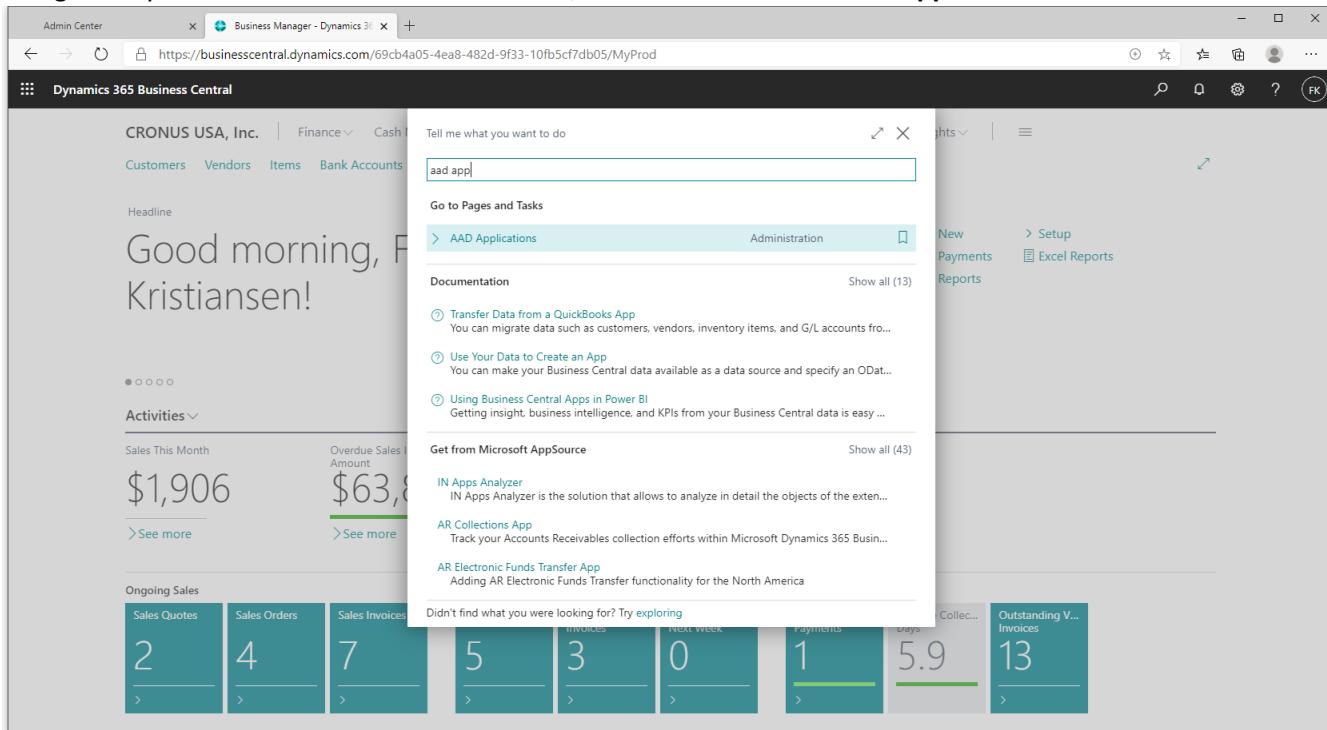


Dynamics 365 Business Central admin center

Environments

Name	Application Family	Type	State	Country/region	Current Version	Available Update Version	Scheduled Upc
Production	Business Central	Production	Active	US	17.0.16993.0		
MyProd	Business Central	Production	Active	US	17.0.16993.0		
Sandbox	Business Central	Sandbox	Active	US	17.0.16993.0		

Permissions in Business Central to allow the AAD App to do Automation and Extension Management
Navigate to your **Business Central environment**, click search and enter **aad app**



Admin Center x Business Manager - Dynamics 365 x

https://businesscentral.dynamics.com/69cb4a05-4ea8-482d-9f33-10fb5cf7db05/MyProd

Dynamics 365 Business Central

CRONUS USA, Inc. | Finance ▾ Cash ▾

Customers Vendors Items Bank Accounts

Headline

Good morning, Kristiansen!

Sales This Month \$1,906

Overdue Sales Amount \$63,8

Activities ▾

Ongoing Sales

Sales Quotes	Sales Orders	Sales Invoices	5	3	0	1	5.9	13
2	4	7	>	>	>	>	>	>

Tell me what you want to do

aad app

Go to Pages and Tasks

> **AAD Applications** Administration

Documentation Show all (13)

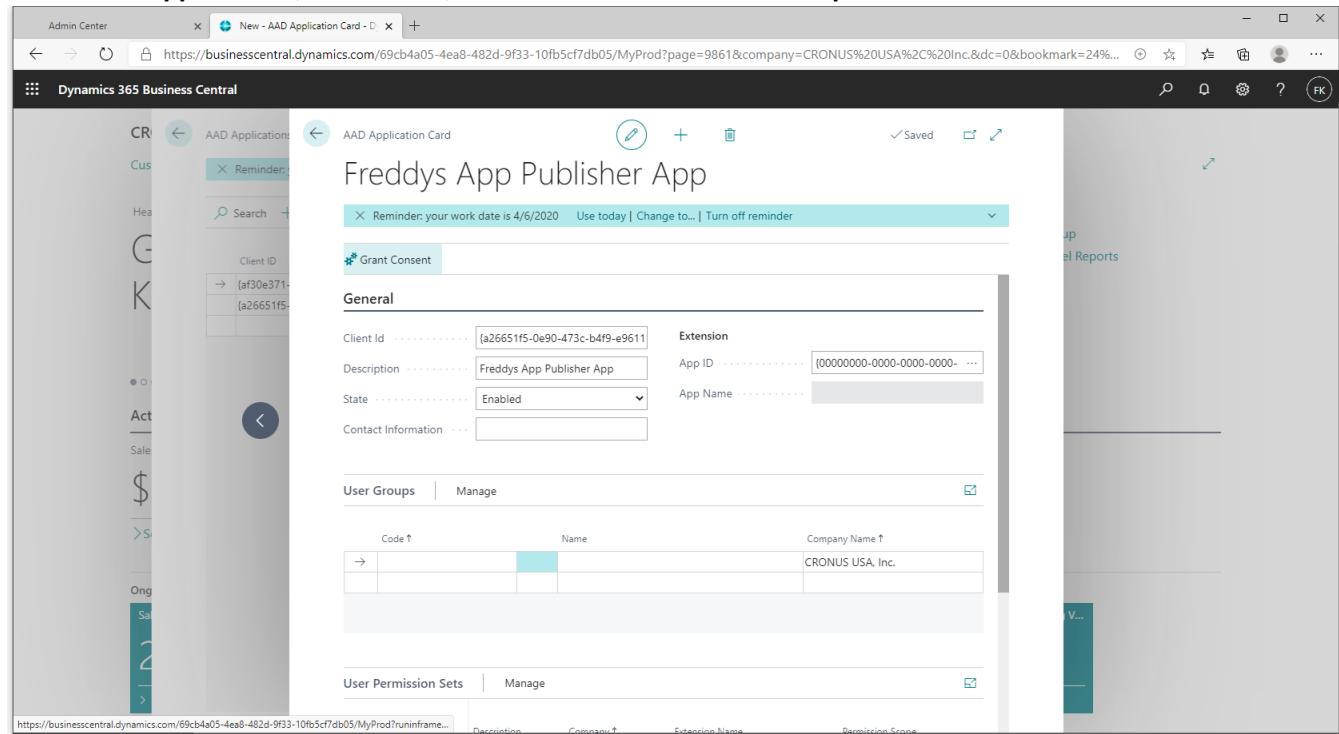
- Transfer Data from a QuickBooks App
- Use Your Data to Create an App
- Using Business Central Apps in Power BI

Get from Microsoft AppSource Show all (43)

- IN Apps Analyzer
- AR Collections App
- AR Electronic Funds Transfer App

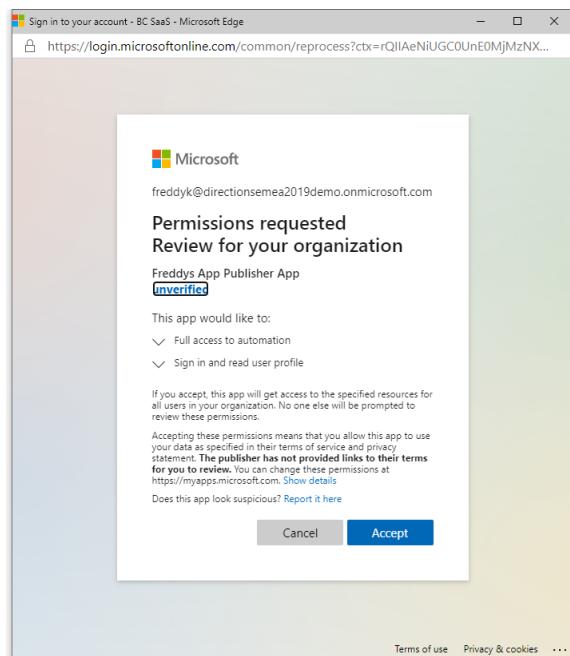
Didn't find what you were looking for? Try exploring

Select AAD Applications, click + New, enter the Client ID and a description and click Grant Consent



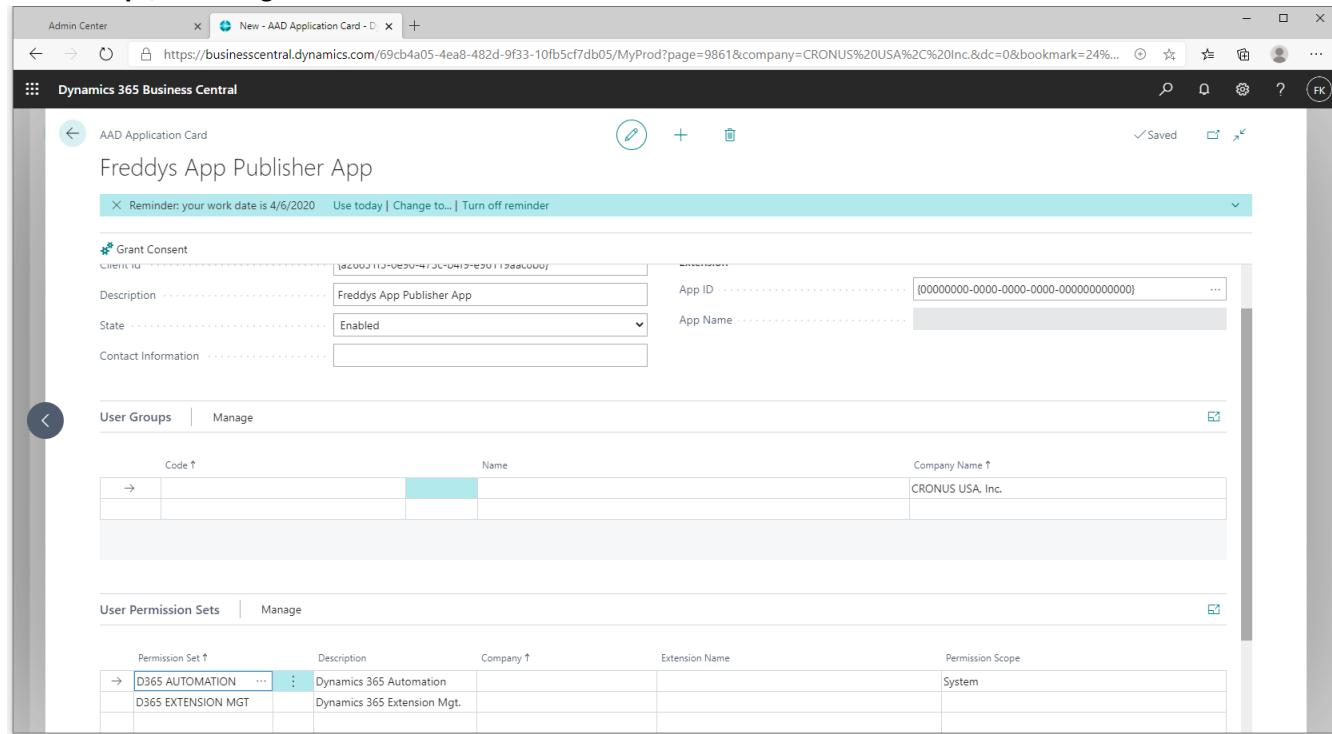
The screenshot shows the Dynamics 365 Business Central Admin Center. On the left, a sidebar lists various modules like CRM, Customers, and Leads. The main area is titled 'AAD Application Card' and shows a card for 'Freddys App Publisher App'. The card includes fields for Client Id (a26651f5-0e90-473c-b4f9-e9611), Description (Freddys App Publisher App), and State (Enabled). It also shows sections for User Groups and User Permission Sets, both of which are currently empty. A reminder at the top indicates 'Reminder: your work date is 4/6/2020'.

You will now be asked to authenticate and give access to Freddys App Publisher App to do automation and sign in and read user profile



The screenshot shows a Microsoft Edge browser window with the title 'Sign in to your account - BC SaaS - Microsoft Edge'. The URL is <https://login.microsoftonline.com/common/reprocess?ctx=rQIAeNiUGC0UnE0MjMzNX...>. The page displays a Microsoft sign-in screen with the text 'Permissions requested' and 'Review for your organization'. It lists the app 'Freddys App Publisher App' as 'unverified'. It then asks for permission to 'Full access to automation' and 'Sign in and read user profile'. A note states that accepting these permissions allows the app to access resources for all users in the organization. The user is given the option to 'Accept' or 'Cancel'.

Click **Accept**, and assign two User Permission Sets: **D365 AUTOMATION** and **D365 EXTENSION MGT**



The screenshot shows the Dynamics 365 Business Central interface for managing AAD Application Cards. The card for 'Freddys App Publisher App' is displayed. In the 'User Groups' section, two permission sets are assigned: 'D365 AUTOMATION' and 'D365 EXTENSION MGT'. In the 'User Permission Sets' section, the same two permission sets are listed with their descriptions and extension names.

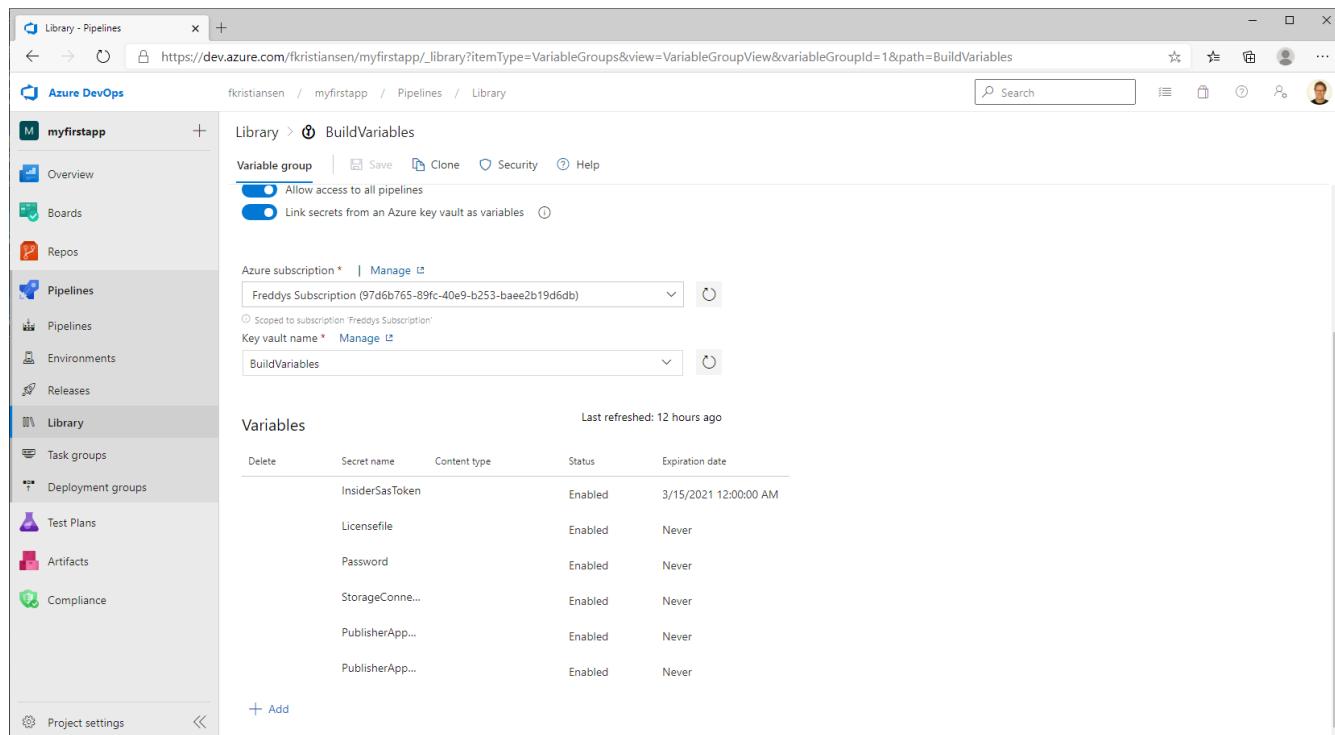
Permission Set	Description	Company	Extension Name	Permission Scope
D365 AUTOMATION	Dynamics 365 Automation			System
D365 EXTENSION MGT	Dynamics 365 Extension Mgt.			

Perform the same steps in the **Sandbox environment**.

Creating the release Pipeline

Open the Azure Portal and locate your Key Vault. Add secrets for **PublisherAppClientId** (Client ID from your AAD App) and **PublisherAppClientSecret** (Client Secret from your AAD App).

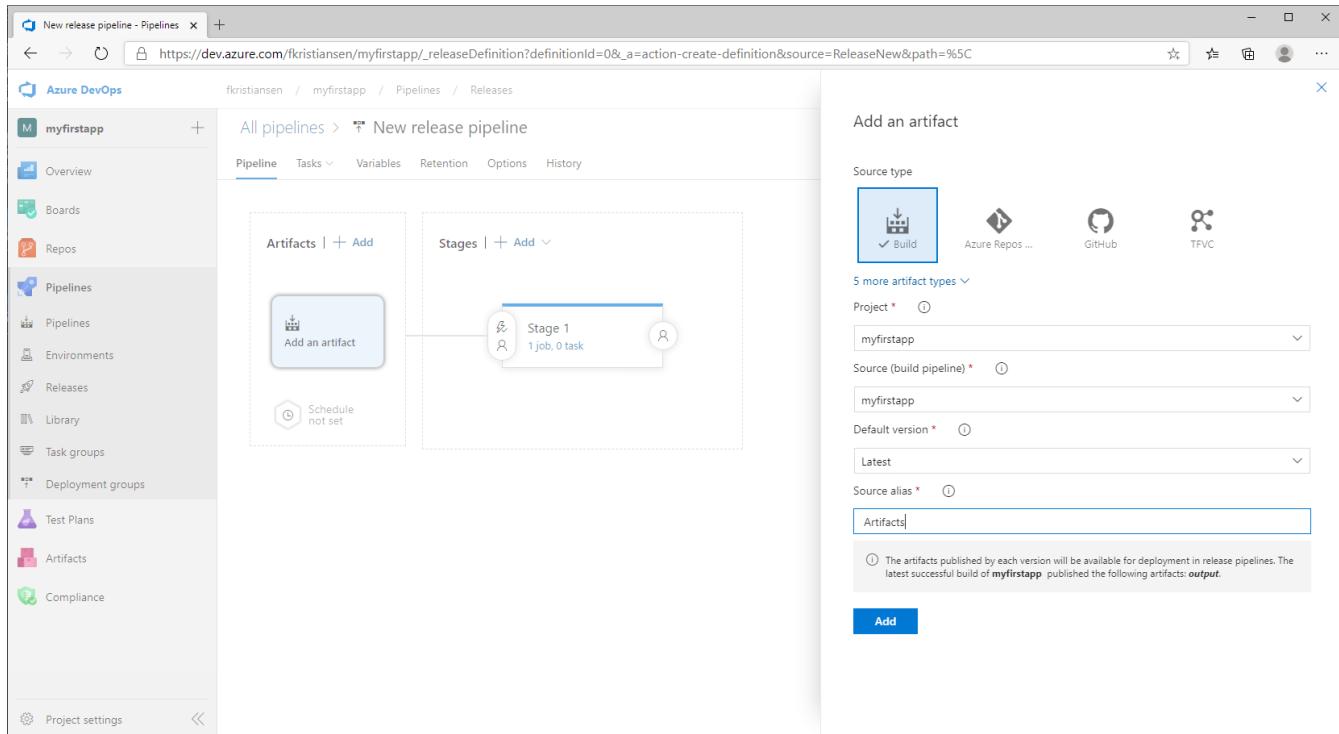
In Azure DevOps, navigate to Pipelines -> Library, modify build variables and add **PublisherAppClientId** and **PublisherAppClientSecret** to get access to these values for the pipelines and **Save**.



The screenshot shows the Azure DevOps Library interface for the project 'myfirstapp'. In the 'Variables' section, several variables are listed with their values and expiration dates. The variables include 'InsiderSasToken', 'Licensefile', 'Password', 'StorageConne...', 'PublisherApp...', and 'PublisherApp...'. The 'PublisherApp...' variables are explicitly mentioned as being added for access to the pipeline values.

Variable	Value	Status	Expiration date
InsiderSasToken		Enabled	3/15/2021 12:00:00 AM
Licensefile		Enabled	Never
Password		Enabled	Never
StorageConne...		Enabled	Never
PublisherApp...		Enabled	Never
PublisherApp...		Enabled	Never

In **Releases**, click **+ New**, add a new **Release Pipeline**, click **Add an artifact**, select the source and set the source alias to **Artifacts**, click **Add**.



New release pipeline - Pipelines

myfirstapp / myfirstapp / Pipelines / Releases

All pipelines > New release pipeline

Artifacts | + Add Stages | + Add

Add an artifact

Schedule not set

Stage 1 1 job, 0 task

Add an artifact

Source type

Build Azure Repos ... GitHub TFVC

5 more artifact types

Project * myfirstapp

Source (build pipeline) * myfirstapp

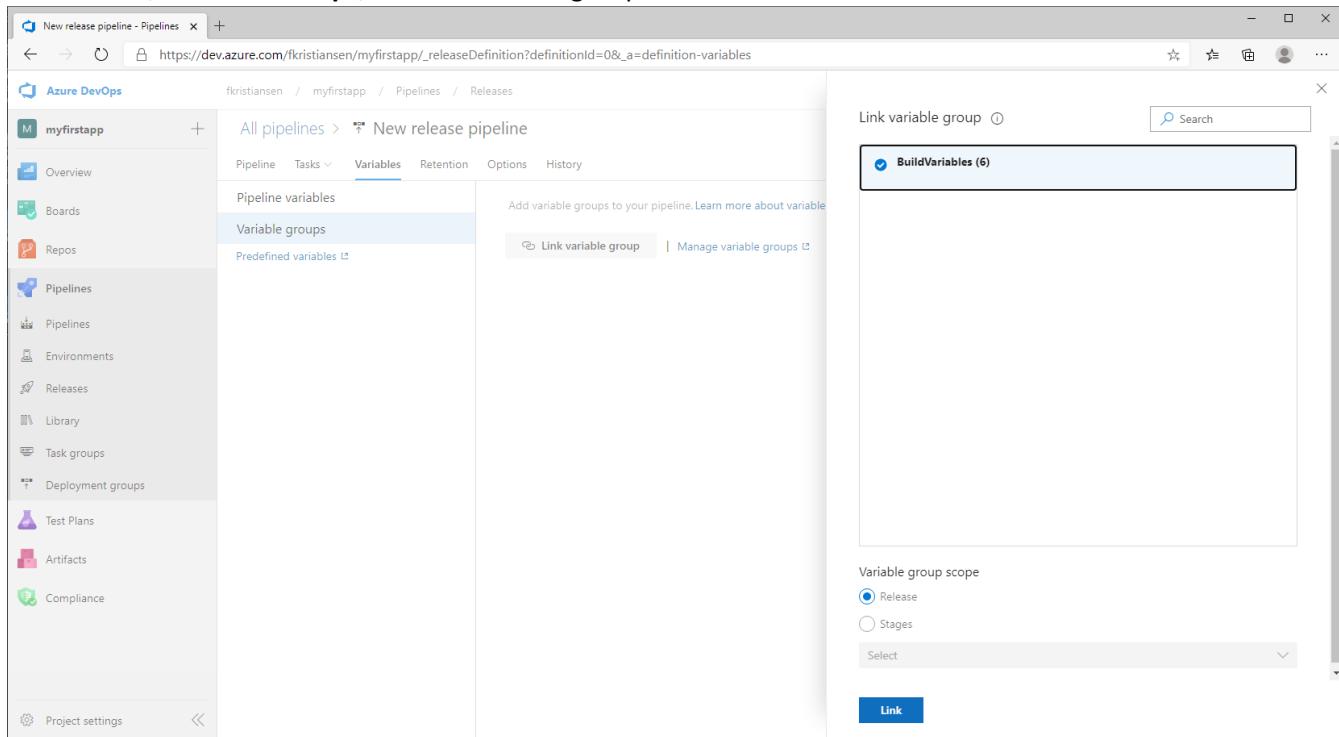
Default version * Latest

Source alias * Artifacts

The artifacts published by each version will be available for deployment in release pipelines. The latest successful build of myfirstapp published the following artifacts: output.

Add

In **Variables**, **Variable Groups**, link the variable group **BuildVariables**



New release pipeline - Pipelines

myfirstapp / myfirstapp / Pipelines / Releases

All pipelines > New release pipeline

Variables

Pipeline variables

Variable groups

Predefined variables

Link variable group | Manage variable groups

Link variable group

Search

BuildVariables (6)

Variable group scope

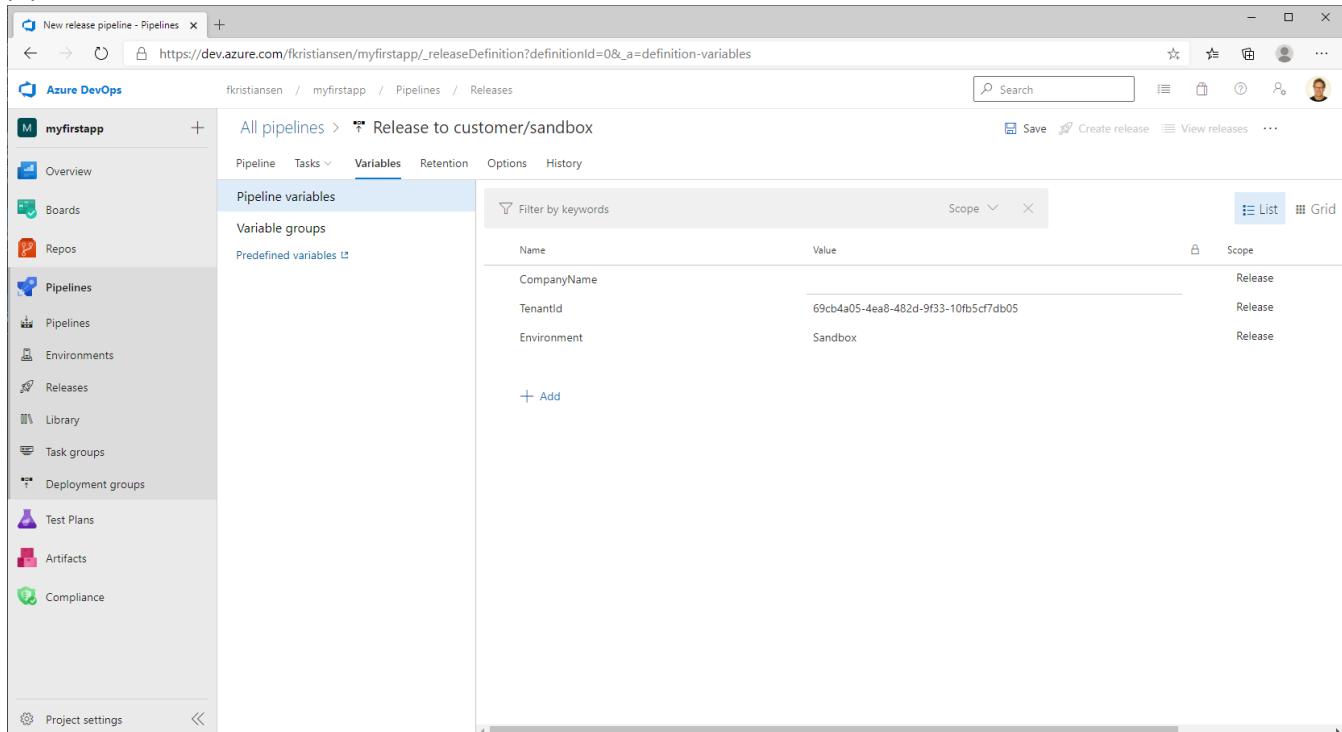
Release (selected)

Stages

Select

Link

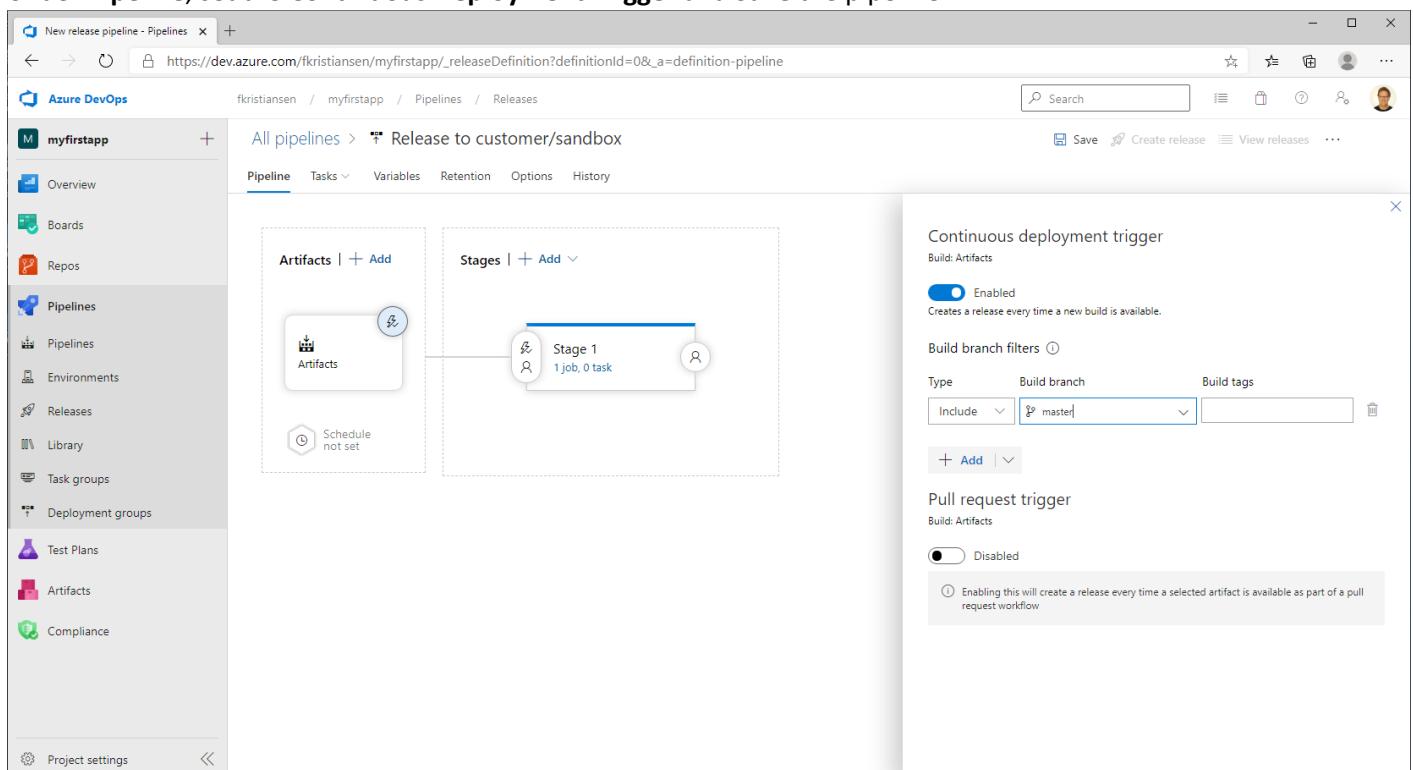
Under pipeline variables, define 3 variables: **CompanyName**, **TenantId** and **Environment**. Set the name of the pipeline to **Release to customer Sandbox**.



The screenshot shows the 'Pipeline variables' section of the Azure DevOps interface. It lists three variables:

Name	Value	Scope
CompanyName	Release	
TenantId	69cb4a05-4ea8-482d-9f33-10fb5cf7db05	Release
Environment	Sandbox	Release

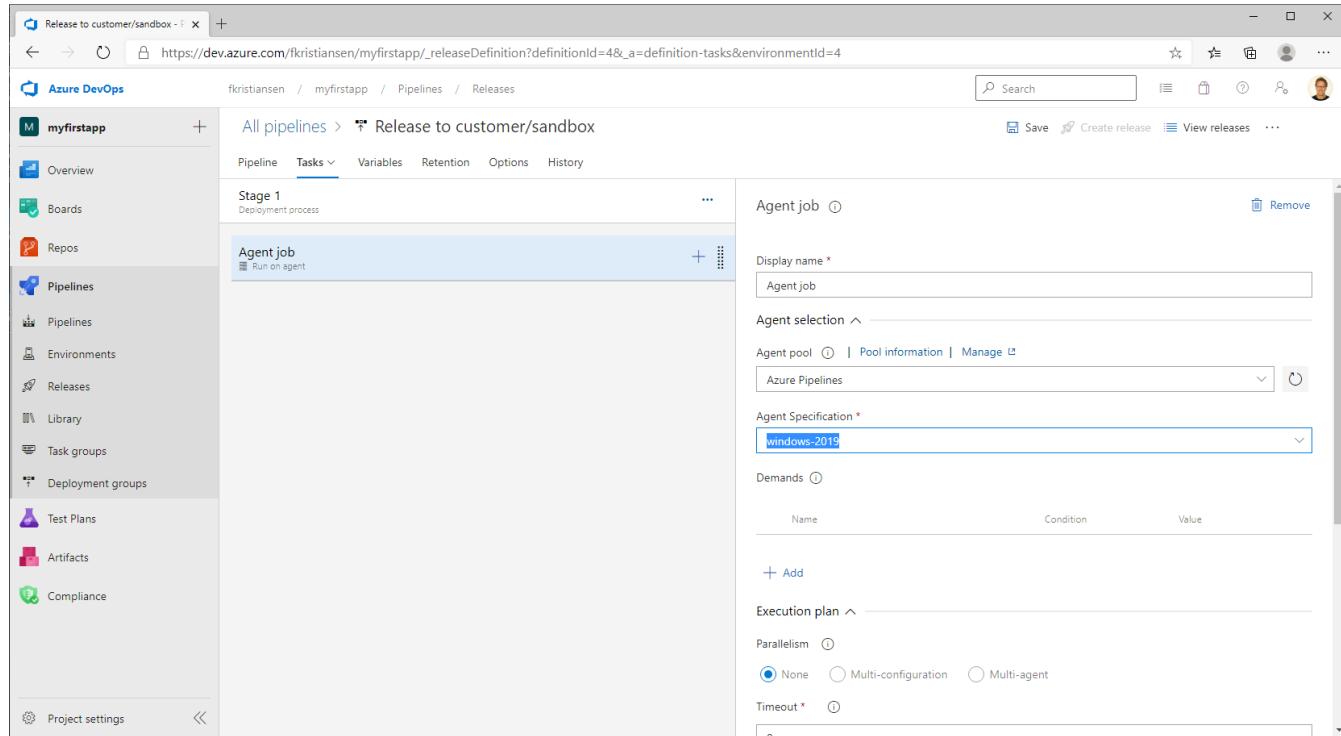
Under **Pipeline**, set the **Continuous Deployment Trigger** and **Save** the pipeline.



The screenshot shows the 'Pipeline' editor for the 'Release to customer/sandbox' pipeline. On the right, the 'Continuous deployment trigger' settings are displayed:

- Build: Artifacts**: Enabled. Creates a release every time a new build is available.
- Build branch filters**: Type: Include, Build branch: master.
- Pull request trigger**: Build: Artifacts, Disabled. Enabling this will create a release every time a selected artifact is available as part of a pull request workflow.

Select Tasks, click Agent Job and select Windows-2019 as Agent Specification.

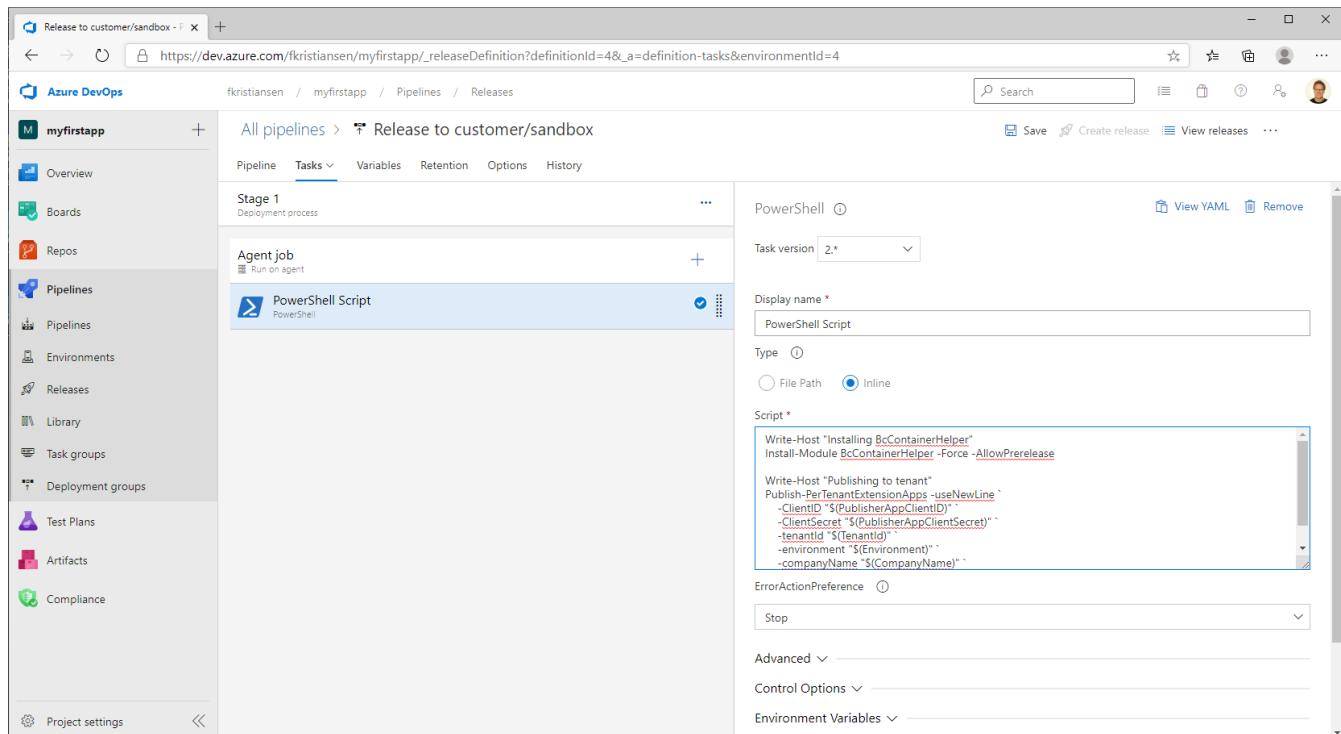


The screenshot shows the Azure DevOps Pipeline configuration for a release definition. The 'Tasks' tab is selected. An 'Agent job' is defined with a display name 'Agent job'. The 'Agent Specification' is set to 'windows-2019'. The 'Demands' section is empty. The 'Execution plan' section is collapsed. The 'Parallelism' section shows 'None' selected. The 'Timeout' section is collapsed.

Click the + in Agent job to add a job, search for PowerShell and add a PowerShell Task. Click the task line and set the type to be inline and copy/paste the following script:

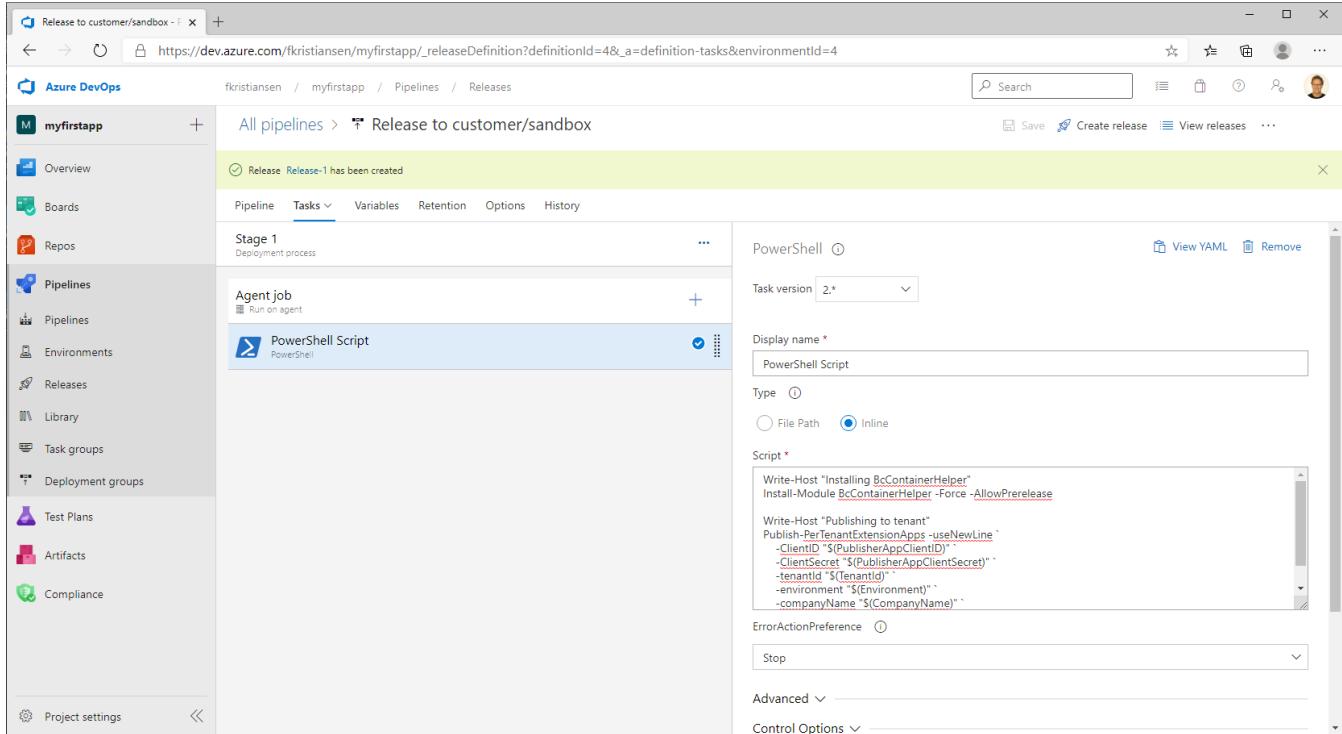
```
Write-Host "Installing BcContainerHelper"
Install-Module BcContainerHelper -Force -AllowPrerelease

Write-Host "Publishing to tenant"
Publish-PerTenantExtensionApps -useNewLine
-ClientID "$(_PublisherAppClientID)"
-ClientSecret "$(_PublisherAppClientSecret)"
-tenantId "$(_TenantId)"
-environment "$(_Environment)"
-companyName "$(_CompanyName)"
-appFiles @((Get-Item "Artifacts/output/Apps/*.app" | % { $_.FullName }))
```



The screenshot shows the Azure DevOps Pipeline configuration for a release definition. A 'PowerShell' task is added to the 'Agent job'. The 'Type' is set to 'Inline'. The 'Script' field contains the PowerShell script provided in the previous step. The 'ErrorActionPreference' is set to 'Stop'. The 'Advanced' and 'Control Options' sections are collapsed. The 'Environment Variables' section is collapsed.

Save the pipeline and press **Create Release** to publish the latest version of your app to the **Sandbox** environment.



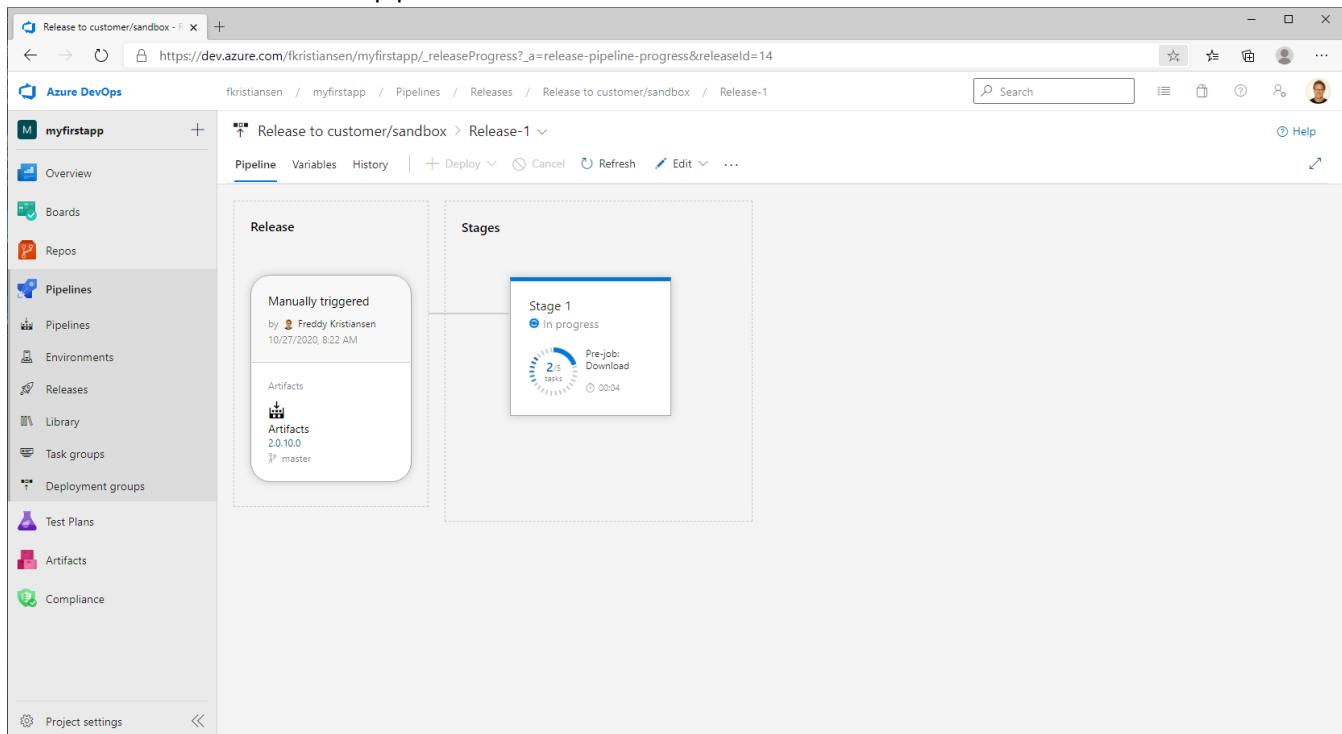
The screenshot shows the Azure DevOps Pipelines interface for a project named 'myfirstapp'. The pipeline 'Release to customer/sandbox' is displayed, with a 'Release-1' release being created. The 'Tasks' tab is selected, showing a single 'PowerShell Script' task. The task configuration includes:

- Task version:** 2.*
- Display name:** PowerShell Script
- Type:** Inline
- Script:**

```
Write-Host "Installing BcContainerHelper"
Install-Module BcContainerHelper -Force -AllowPrerelease

Write-Host "Publishing to tenant"
Publish-PerTenantExtensionApps -useNewLine
-ClientID "$PublisherAppClientId"
-ClientSecret "$PublisherAppClientSecret"
-tenantId "$tenantId"
-environment "$Environment"
-companyName "$CompanyName"
```
- ErrorActionPreference:** Stop

Click **Release-1** to monitor the pipeline

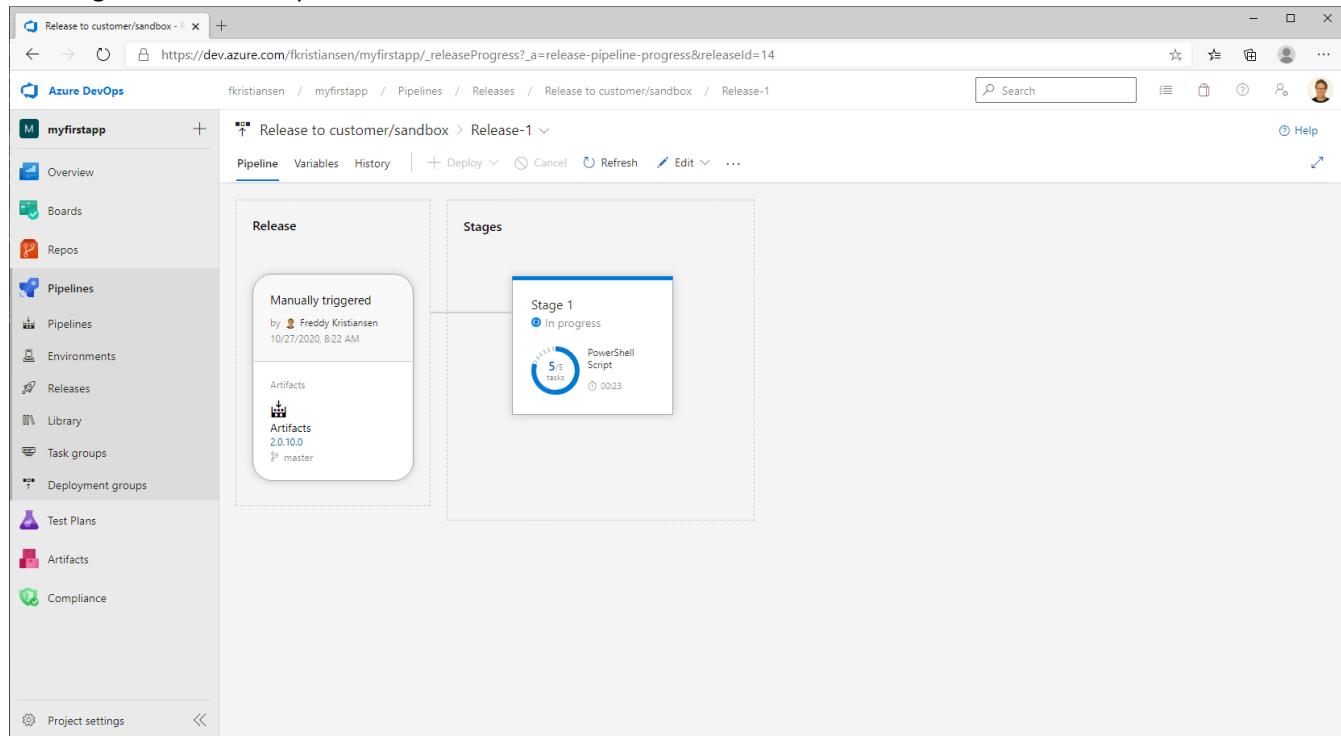


The screenshot shows the Azure DevOps Releases interface for the 'Release to customer/sandbox' pipeline. A release named 'Release-1' is shown, which was manually triggered on 10/27/2020 at 8:22 AM. The release consists of a single stage named 'Stage 1', which is currently 'In progress'. The stage details show:

- Pre-job:** Download
- 2/5 tasks:** 00:04

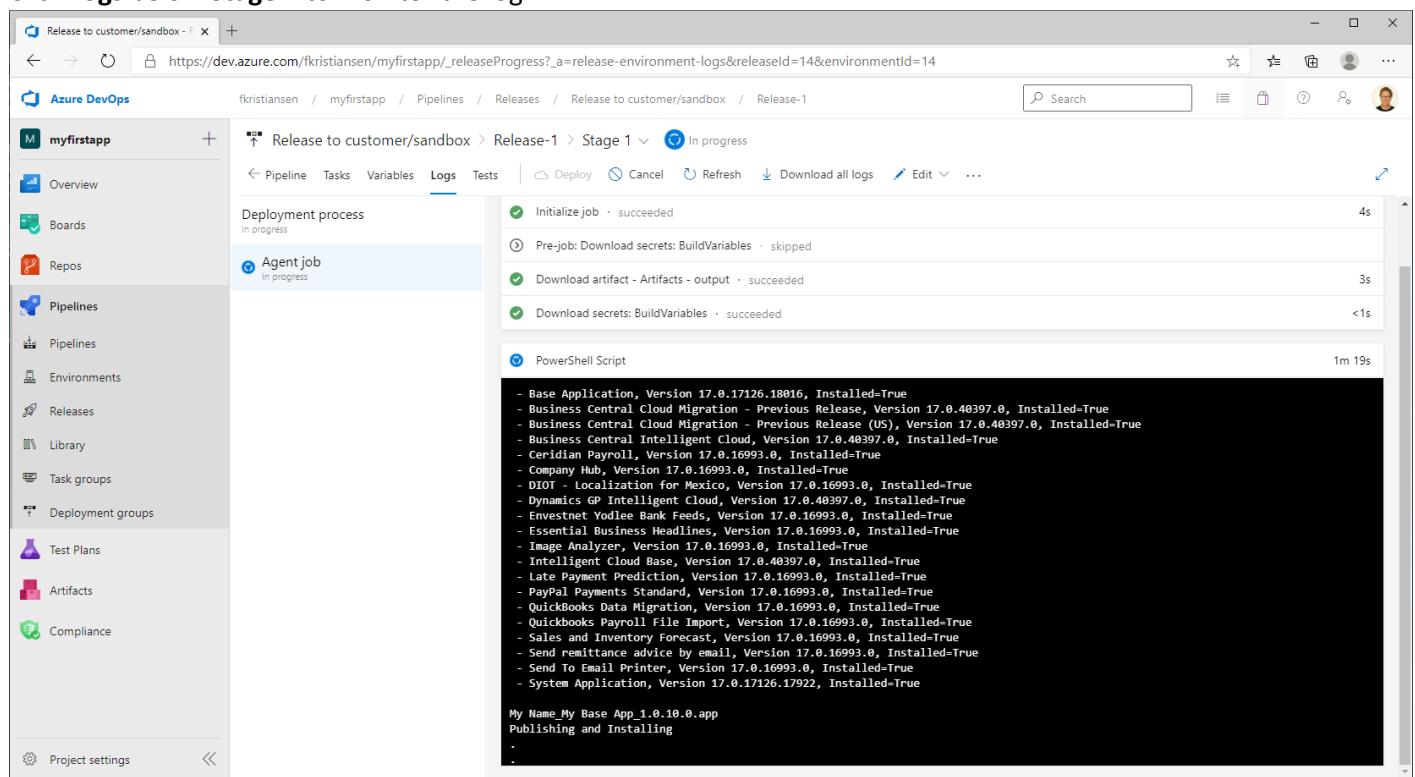
The pipeline configuration on the left shows the 'PowerShell Script' task with the same PowerShell script as in the previous screenshot.

Running PowerShell Script



The screenshot shows the Azure DevOps Pipeline interface for a release named 'Release to customer/sandbox' under 'Release-1'. The 'Release' section shows it was manually triggered by 'Freddy Kristiansen' on 10/27/2020, 8:22 AM. The 'Stages' section shows 'Stage 1' is in progress, with a 'PowerShell Script' task. The task has 5/5 tasks completed and took 00:23. Artifacts for version 2.0.10.0 from the 'master' branch are listed.

Click Logs below Stage 1 to monitor the log

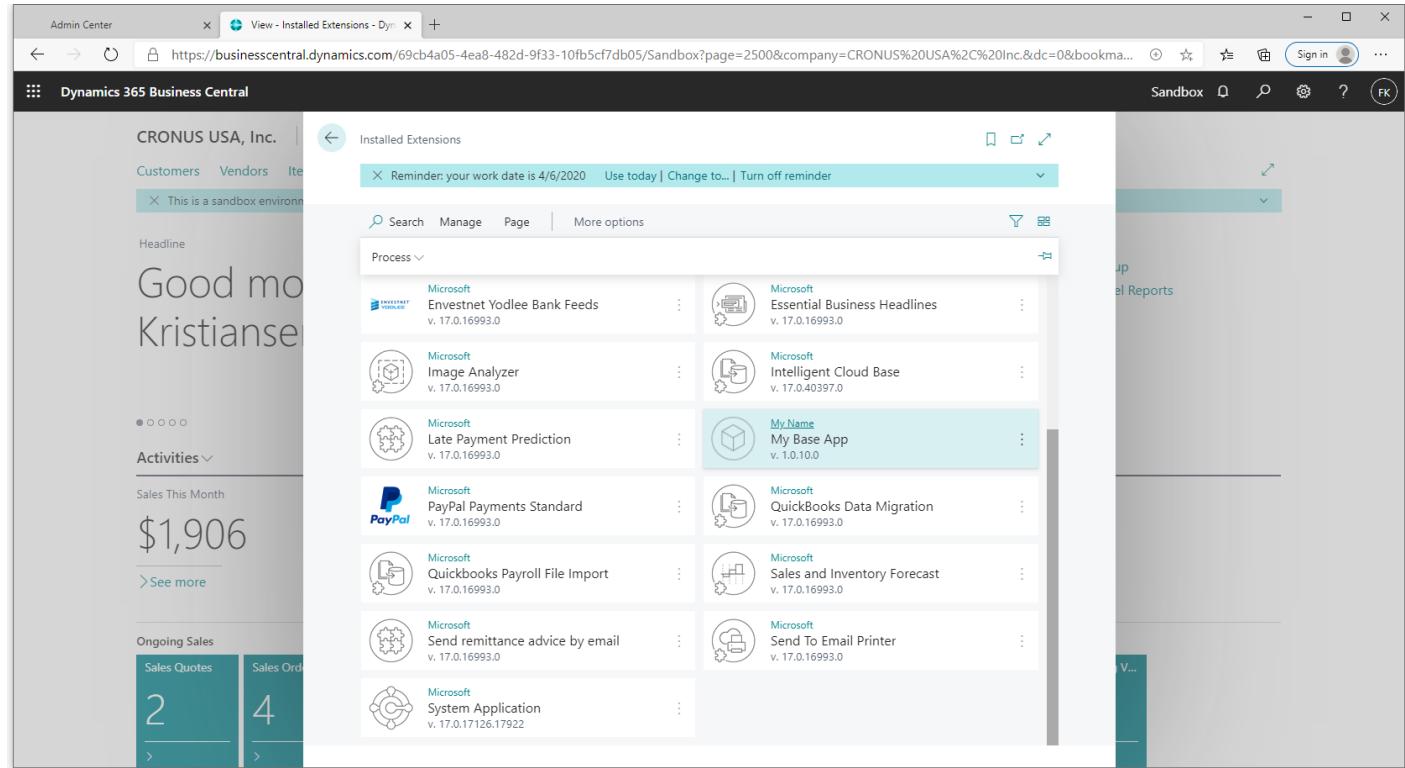


The screenshot shows the 'Logs' tab for Stage 1 of the release. The deployment process is in progress. The 'Agent job' task is also in progress. The log output for the PowerShell Script task shows the following content:

```
- Base Application, Version 17.0.17126.18816, Installed=True
- Business Central Cloud Migration - Previous Release, Version 17.0.40397.0, Installed=True
- Business Central Cloud Migration - Previous Release (US), Version 17.0.40397.0, Installed=True
- Business Central Intelligent Cloud, Version 17.0.40397.0, Installed=True
- Business Central Payroll, Version 17.0.16993.0, Installed=True
- Company Hub, Version 17.0.16993.0, Installed=True
- DIOT - Localization for Mexico, Version 17.0.16993.0, Installed=True
- Dynamics GP Intelligent Cloud, Version 17.0.40397.0, Installed=True
- Envestnet Yodlee Bank Feeds, Version 17.0.16993.0, Installed=True
- Essential Business Headlines, Version 17.0.16993.0, Installed=True
- Image Analyzer, Version 17.0.16993.0, Installed=True
- Intelligent Cloud Base, Version 17.0.40397.0, Installed=True
- Late Payment Prediction, Version 17.0.16993.0, Installed=True
- PayPal Payments Standard, Version 17.0.16993.0, Installed=True
- QuickBooks Data Migration, Version 17.0.16993.0, Installed=True
- Quickbooks Payroll File Import, Version 17.0.16993.0, Installed=True
- Sales and Inventory Forecast, Version 17.0.16993.0, Installed=True
- Send Remittance advice by email, Version 17.0.16993.0, Installed=True
- Send To Email Printer, Version 17.0.16993.0, Installed=True
- System Application, Version 17.0.17126.17922, Installed=True

My Name_My Base App_1.0.10.0.app
Publishing and Installing
.
```

You can also login to your **sandbox environment** and see that the apps are getting installed



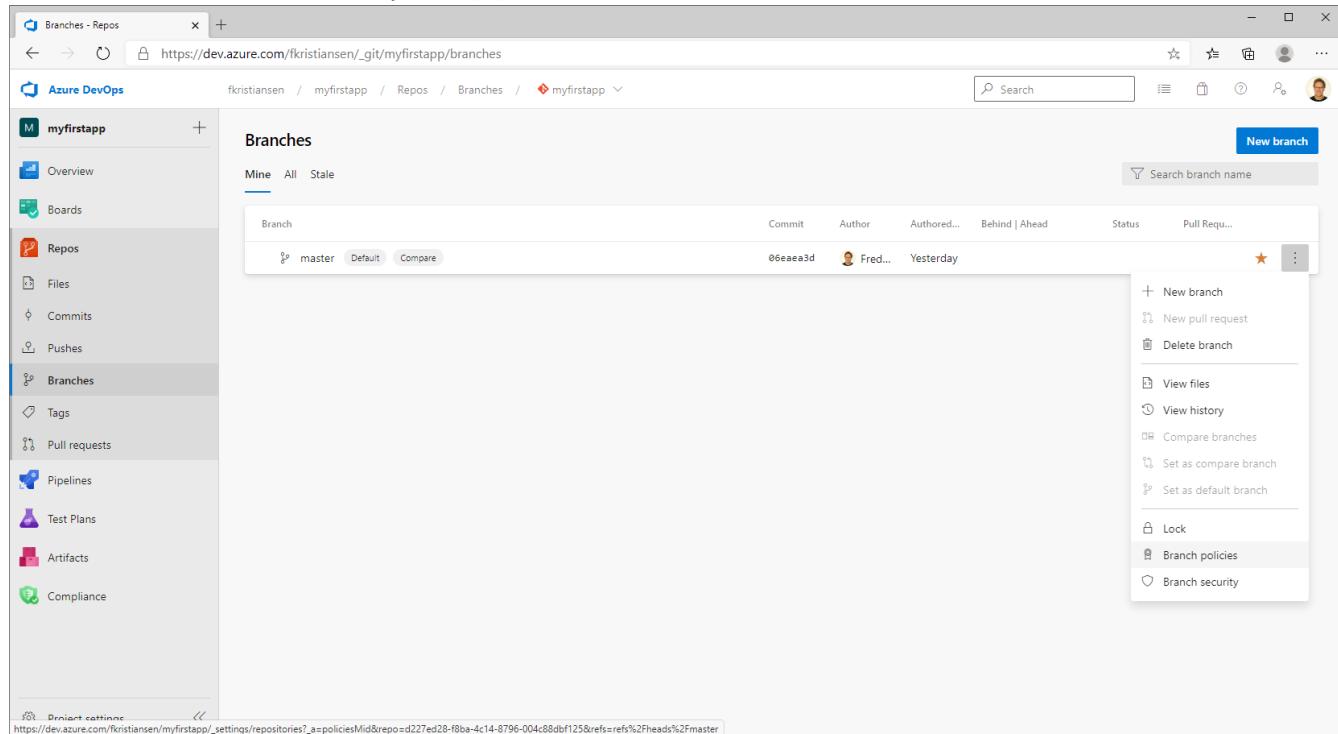
You can repeat the process for the production tenant, without the continuous deployment trigger and deploy the app on demand.

Congratulations, you have successfully published your app to an online tenant.

Branch Policies

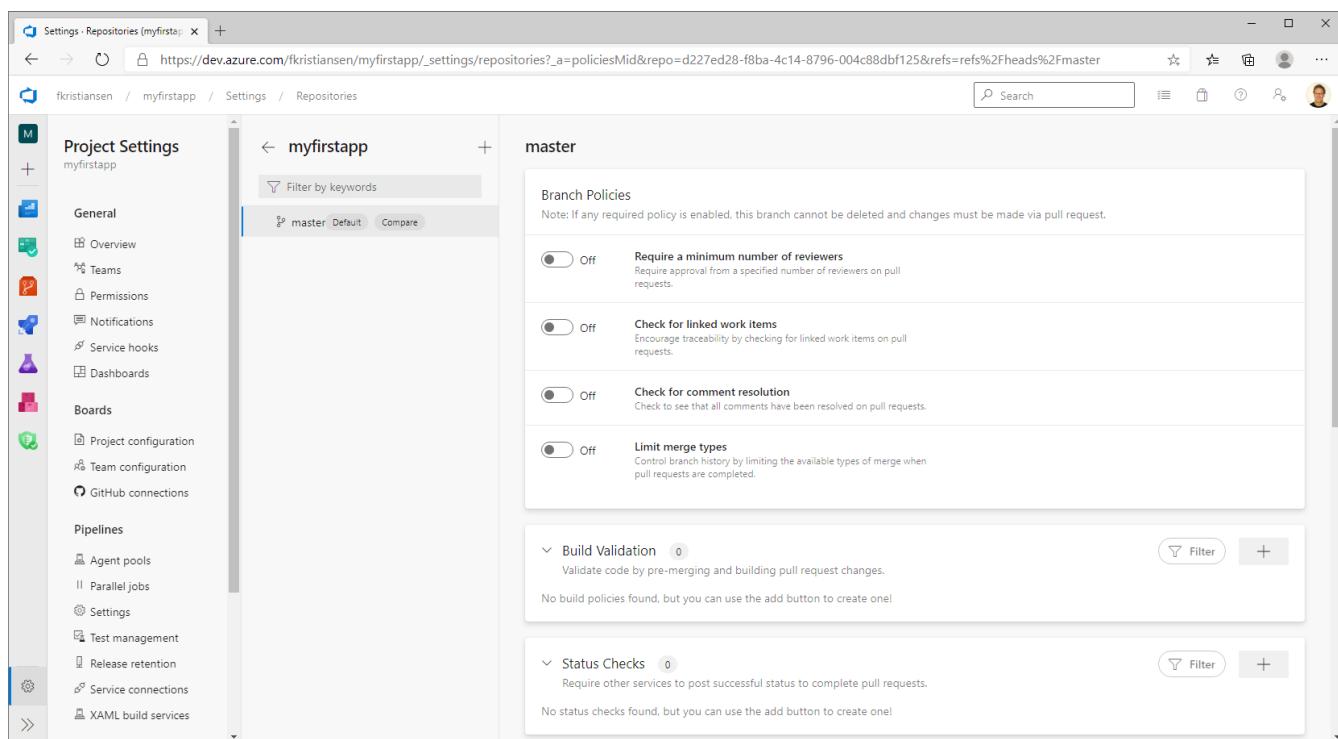
At this time, we have an Azure DevOps project with a build pipeline, which we can kick off manually. We also have a release pipeline, which helps us release the product, but in order to raise the quality bar even more, we should setup branch policies to demand code reviews and successful builds before a fix makes it into the repository.

Open your Azure DevOps organization (eg. <https://dev.azure.com/>) and select **Repos** -> **Branches**. Select the **master** branch, click the **more actions** symbol (...) and choose **Branch Policies**.



The screenshot shows the 'Branches' page in Azure DevOps. The 'Branches' table lists the 'master' branch, which is the default branch. A context menu is open over the 'master' branch, with the 'Branch policies' option highlighted. Other options in the menu include 'New branch', 'New pull request', 'Delete branch', 'View files', 'View history', 'Compare branches', 'Set as compare branch', 'Set as default branch', 'Lock', and 'Branch security'. The URL in the browser is https://dev.azure.com/fkristiansen/myfirstapp/_settings/repositories?_a=policiesMid&repo=d227ed28-f8ba-4c14-8796-004c88dbf125&refs=refs%2Fheads%2Fmaster.

You should now be able to setup branch policies for your branch



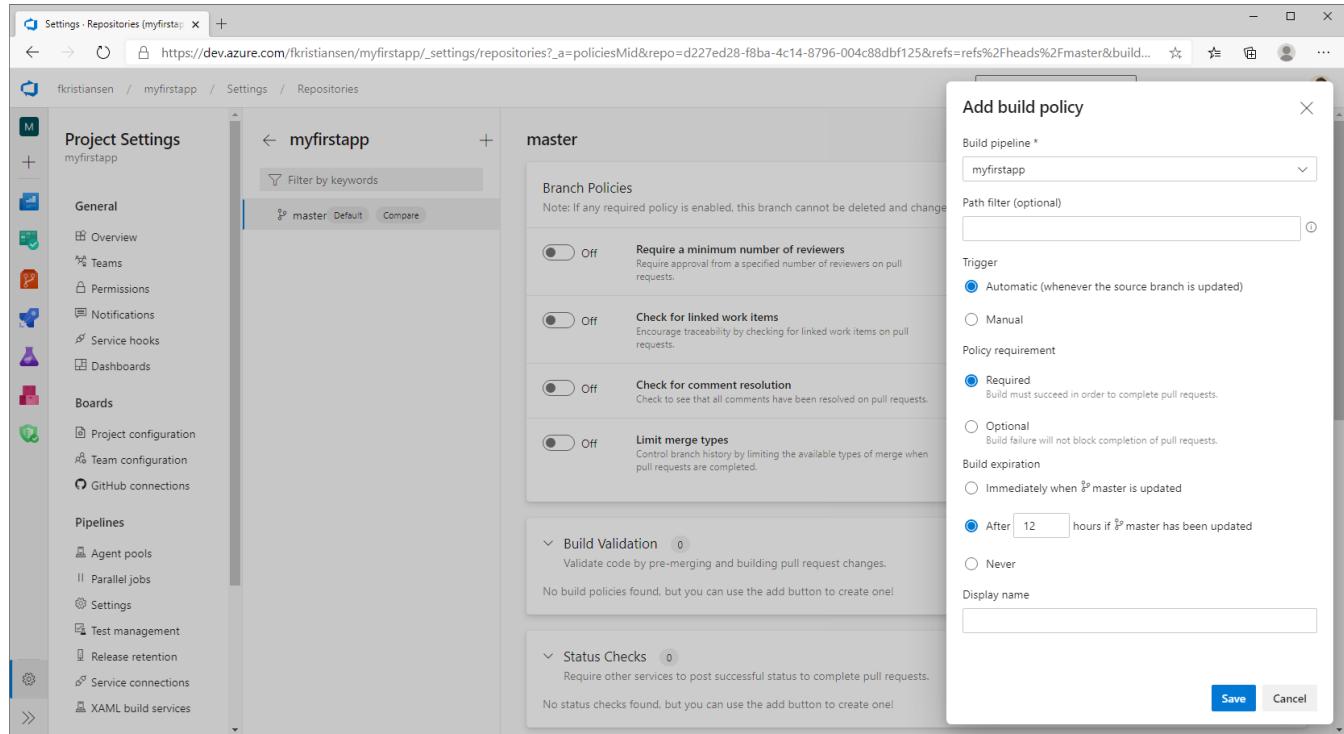
The screenshot shows the 'Settings - Repositories (myfirstapp)' page. The 'master' branch settings are displayed, showing the 'Branch Policies' section. The section notes that if any required policy is enabled, the branch cannot be deleted and changes must be made via pull request. There are four policy options: 'Require a minimum number of reviewers' (off), 'Check for linked work items' (off), 'Check for comment resolution' (off), and 'Limit merge types' (off). Below these are sections for 'Build Validation' and 'Status Checks', both of which currently have no policies defined.

The first thing you will notice is, that setting **any Required policy** will enforce the use of **pull requests** and will **disallow direct checkins** to the **master** branch. It will also prevent the branch from unintended or evil deletion.

You can read much more about the policies here: <https://docs.microsoft.com/en-us/azure/devops/repos/git/branch-policies>.

What I want, is to ensure that my CI build pipeline runs whenever somebody checks something in and that the checkin cannot be completed if the build isn't successful. This is called **Build validation** and by adding a build policy

with our CI pipeline, we should be ready to go.



The screenshot shows the Azure DevOps interface for managing repository settings. On the left, the 'Project Settings' sidebar is visible with options like General, Boards, Pipelines, and Test management. The main area shows the 'myfirstapp' repository with the 'master' branch selected. A modal window titled 'Add build policy' is open on the right, prompting the user to define a policy for the 'myfirstapp' pipeline. The 'Build pipeline' dropdown is set to 'myfirstapp'. The 'Trigger' section is set to 'Automatic (whenever the source branch is updated)'. The 'Policy requirement' section is set to 'Required'. The 'Build expiration' section is set to 'After 12 hours if the master has been updated'. The 'Display name' field is empty. The 'Save' and 'Cancel' buttons are at the bottom right of the modal.

Congratulations

**I know you think you are just getting started here,
but with this, you have successfully completed this
Hands On Lab.**