[Automated publishing for continuous integration and delivery - Azure Data Factory | Microsoft Learn](https://learn.microsoft.com/en-us/azure/data-factory/continuous-integration-delivery-improvements)

**CI/CD Preparation:**

* As we need hosted parallelism to run pipeline in step 2, this can be requested for free by filling out the form at below link, then submit, and await for approval. Usually it takes up to 24 hours to obtain the approval.
* [**https://aka.ms/azpipelines-parallelism-request**](https://aka.ms/azpipelines-parallelism-request)**.**

**A screenshot of a computer

Description automatically generated**

**Step1:**

Create 3 Data Factory located within 3 different resource group:

1/ For Development

2/ For Test

3/ For Production

**Step 2:**

Data Factory Git Configuration using Dev Ops Portal

A screenshot of a computer

Description automatically generated

Select branch under Repo, then click on the 3 dots sign to create branch policy

A screenshot of a computer

Description automatically generated

Go to Development Data Factory to set up the Git depository

A screenshot of a computer

Description automatically generated

**A screenshot of a computer

Description automatically generated**

We cant change the name of the publish branch, it’s always be **adf\_publish**.

**A screenshot of a computer

Description automatically generated**

*No worry to see this error, it’s meant we’re not allowed to work directly on the main branch, we’re required to work on the feature branch.*

There’re 2 options in deploying CI/CD as below:

**Option1:**

A diagram of a software flow

Description automatically generated

1. Each user makes changes in their private branches.
2. Push to master isn't allowed. Users must create a pull request to make changes.
3. Users must load the Data Factory UI and select **Publish** to deploy changes to Data Factory and generate the ARM templates in the publish branch.
4. The DevOps Release pipeline is configured to create a new release and deploy the ARM template each time a new change is pushed to the publish branch.

**Option 2:**

A diagram of a software flow

Description automatically generated

1. Each user makes changes in their private branches.
2. Push to master isn't allowed. Users must create a pull request to make changes.
3. The Azure DevOps pipeline build is triggered every time a new commit is made to master. It validates the resources and generates an ARM template as an artifact if validation succeeds.
4. The DevOps Release pipeline is configured to create a new release and deploy the ARM template each time a new build is available.

So far I only succeeded with deploying option 1. For option 2, I failed at the release pipeline step.

Below is the proof that the pipeline ran successfully.

A screenshot of a computer

Description automatically generated