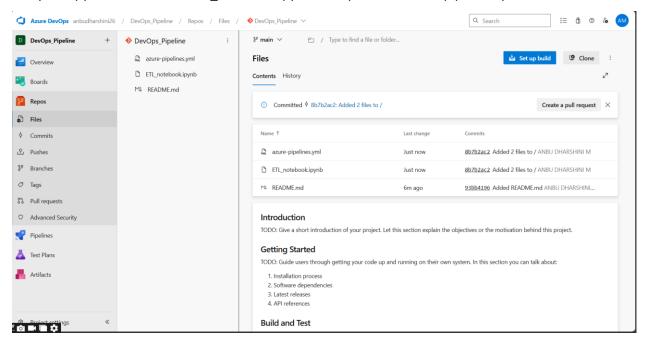# Automation via Azure DevOps Workflow
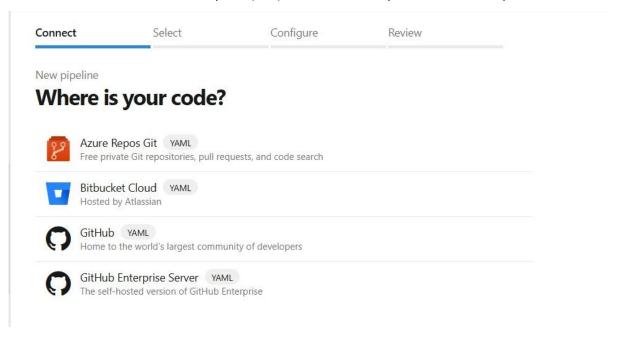
Step-1:

Create New Project -> SupplyChainAutomation -> Click on to the Three dots on the top right corner and add your .ipynb notebook -> "ETL_notebook.ipynb" and yaml file "azure-pipelines.yml" file.

Step-2:

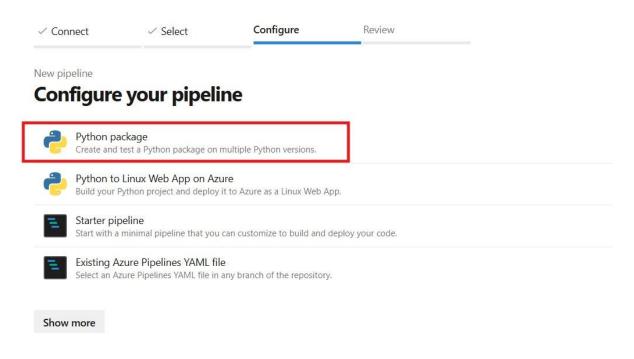Select the Version control system(VCS) to continue. In my case it is Azure Repo Git.

Step-3:

Select the respective repo at which the code, datasets are present. In my case it is present in hexacicd-local.

Step-4:

Then select python package to configure the pipeline.

Step-5:
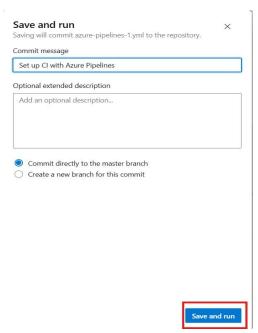
Ensure the YAML file configured properly

trigger	branch to run
pool	agent to run the tasks
vmImage	virtual machine image
steps	sequence of tasks to perform
task	a predefined tasks from azure
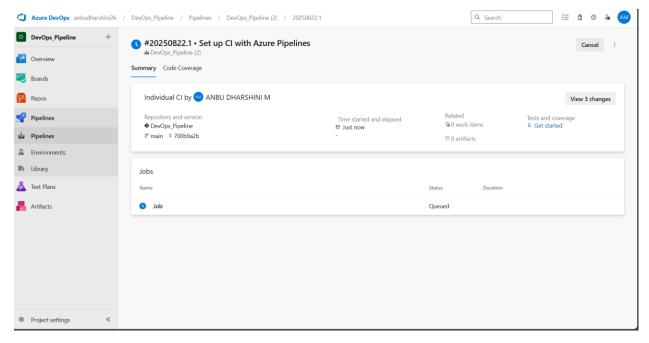
Step-6:

Once all configuration are set then click save and run

Step-7:

And then we can see the summary of the pipeline and it is scheduled to run with the configured agent.

Capstone Tasks

1. Set up a DevOps pipeline to automate weekly processing

Entire workflow defines how to setup a devops pipeline and automate it.

2. Schedule the pipeline to run every Monday

During the YAML configuration in **step-5** we can define the schedule using cron expression

```
1  schedules:
2  - cron: "0 0 * * 1"
3    displayName: Monday Run
4    branches:
5      include:
6      - main
7    always: true
```

3. Output a report with top 5 absentees or lowest performing departments

The ETL_databricks_notebook.ipynb has the operations to report the 1. top 5
abscentees

```
1  abscenteesCount = dfEmp.groupBy("EmpID") \
2      .agg(
3      F.sum("IsAbscent").alias("AbscentCount")
4      ) \
5      .sort("AbscentCount", ascending=False) \
6      .limit(5)
7
8  abscenteesCount.write.mode("overwrite").csv("abs_emp_count")
```

2.   lowest performing departments

```
1  deptMetrics = dfJoined.groupBy("department") \
2      .agg(
3          F.sum("isLate").alias("LateCount"),
4          F.sum("isAbscent").alias("AbscentCount"),
5          F.round(F.mean("hoursWorked"), 2).alias("AverageWorkHours"),
6          F.round(F.mean("productivityScore"), 2).alias("AverageProductivityScore"),
7          F.sum("tasksCompeleted").alias("TotalTasksCompleted")
8      ) \
9      .sort("AverageProductivityScore", ascending=True) \
10      .limit(2)
11
12  deptMetrics.write.mode("overwrite").csv("deptMetrics")
```

Deliverables

YAML file and report file of latest attendance metrics is present in /Deliverables folder pushed into
github.