To connect Azure Databricks to Azure DevOps for version control, you'll need to set up a Git repository in Azure DevOps and then configure your Azure Databricks workspace to connect to it. Here's a step-by-step guide on how to do this:

---

\*\*Connecting Azure Databricks to Azure DevOps for Version Control\*\*

\*\*Step 1: Set up a Git Repository in Azure DevOps\*\*

1. Sign in to your Azure DevOps account.

2. Create a new project or select an existing one where you want to host your Databricks notebooks.

3. Navigate to Repositories and create a new Git repository. Give it a name and configure any other settings as needed.

\*\*Step 2: Generate a Personal Access Token (PAT)\*\*

1. Go to your Azure DevOps account settings.

2. Under "Personal access tokens," generate a new token with the necessary permissions (typically, you'll need at least "Code - Read & write").

3. Save the generated token securely, as you'll need it later.

\*\*Step 3: Configure Azure Databricks Workspace\*\*

1. Open your Azure Databricks workspace in the Azure portal.

2. Go to the "Workspace" tab and select the workspace you want to connect to Azure DevOps.

3. In the workspace settings, navigate to the "Git Integrations" section.

\*\*Step 4: Connect Azure Databricks to Azure DevOps\*\*

1. Click on "Connect Git Repository."

2. Choose "Azure DevOps" as your Git provider.

3. Enter the repository URL (should be in the format `https://dev.azure.com/<organization>/<project>/\_git/<repository>`).

4. Paste the Personal Access Token you generated in Step 2 when prompted.

5. Configure other settings like the branch to sync notebooks from (typically `main` or `master`).

6. Click "Connect" to finalize the connection.

\*\*Step 5: Sync Notebooks\*\*

1. Once connected, you can sync notebooks between Azure Databricks and Azure DevOps.

2. In Databricks, select the notebooks you want to version control.

3. Use the "Version Control" option to commit changes to the connected Git repository.

\*\*Step 6: Collaborate and Manage Versions\*\*

1. With the connection established, you can collaborate with your team on notebooks using Azure DevOps.

2. Use the version control features of Git to manage changes, branches, and merge requests.

3. Monitor and track changes directly from Azure DevOps.

By following these steps, you've successfully connected your Azure Databricks workspace to Azure DevOps for version control, enabling seamless collaboration and efficient management of your Databricks notebooks.

---

Save this content in a Word document, and you'll have a guide ready to connect Azure Databricks to Azure DevOps for version control.