**Overview**

This PowerShell script automates the migration of repositories from **TFS TFVC (Team Foundation Version Control)** to **GitHub**. It reads project and repository details from a JSON file, creates corresponding repositories on GitHub, clones the TFVC repositories, and pushes them to GitHub.

**Prerequisites**

1. **Personal Access Tokens (PATs):**
   * A TFS PAT with sufficient permissions to access and clone repositories.
   * A GitHub PAT with permissions to create repositories and push code.
2. **Git-TFS Tool:**
   * Ensure git-tfs is installed and available in the system's PATH.
3. **JSON File:**
   * A JSON file containing project and repository details (e.g., projectRepoDetailsTFS2017.json).
4. **Git Installation:**
   * Git must be installed and available in the system's PATH.

**Script Workflow**

**1. Configuration**

* **TFS and GitHub PATs:**
  + Replace the placeholders for $tfsPat and $githubPat with your actual PATs.
  + Good practice to pull them from a GitHub secrets
* **Paths:**
  + Update $ProjectListJsonPath with the correct path to the JSON file containing project and repository details.
  + To get the latest projects list and repository details please run the below script **list-projects.ps1 ([tfs-migration/list-projects.ps1 at main · cigna-group-infrastructure-services/tfs-migration](https://github.com/cigna-group-infrastructure-services/tfs-migration/blob/main/list-projects.ps1))**
  + Update $downloadedReposFolder with the desired directory for cloning repositories.

**2. Read and Parse JSON**

* The script reads the JSON file specified in $ProjectListJsonPath and parses it into $projectData.

**3. Environment Setup**

* Sets the TFS PAT as an environment variable (GIT\_TFS\_PAT) for use by git-tfs.
* Ensures the $downloadedReposFolder exists, creating it if necessary.

**4. Process Projects and Repositories**

* Iterates through each project in the JSON file:
  + Skips invalid or non-TFVC repositories.
  + Extracts the repository name and prepares a sanitized name for GitHub.

**5. Create GitHub Repository**

* Sends a POST request to the GitHub API to create a repository for each TFVC repository.
* Logs the success or failure of the repository creation.

**6. Clone TFVC Repository**

* Uses git-tfs to clone the TFVC repository into the local directory.
* Logs any errors encountered during the cloning process.

**7. Push to GitHub**

* Configures the GitHub repository as the remote origin.
* Pushes all branches and tags to GitHub.
* Cleans up the local repository using git tfs cleanup.