azure dev ops repos

Repos

> 📘 The application code that is developed is a major part of the lifecycle.

Azure Repos are used to host that code. Azure repos are a `source control management system` like Github, and has similar features.

- Code Review

- Pull Requests

- Branch Policies

It uses `git`, and the corresponding workflow. The repository branches and pull requests can be linked back to the tasks they correspond to.

Creating and Initializing a repository

Open the **Repos** page in your project. From the repo drop-down, you can select `Create a New Repository`.

Verify that Git is the repo type, and name for the repository. After this is done, select `Create`. The repository is now created.

- If you created an empty repo with no README or .gitignore files, you'll see instructions on how to clone the repo to your computer. You'll also see instructions on how to push code in an existing repo into the newly created one.

- You can choose to initialize the repo main branch with a README or .gitignore files. You'll now see an overview of the files in your repo. You can clone the repo using the **Clone** link on the upper right of the page to get working with a local copy of the repo immediately.

Cloning a local copy to your computer

To work with a Git repo, you clone it to your computer. Cloning a repo creates a complete local copy of the repo for you to work with. Cloning also downloads all commits and branches in the repo and sets up a named relationship with the repo on the server. Use this relationship to interact with the existing repo, pushing and pulling changes to share code with your team.

1. From your web browser, open the team project for your organization in Azure DevOps and select `Repos > Files`

2. Select `Clone` in the upper-right corner of the `Files` window and copy the clone URL.

3. You will also need to store the credentials that are generated for you. They are needed to commit changes.

4. Open the Git command window (Git Bash on Git for Windows) and browse to the folder where you want the code from the repo stored on your computer. Run the `git clone` command followed by the path copied from the **Clone URL** in the previous section.

   ```
   git clone https://F22-Team21-Sprenkle-Zhao-Sanders-Kissane-Towery@dev.azure.com/F22-Team21-Sprenkle-Zhao-Sanders-Kissane-Towery/F22-Team21-Sprenkle.Zhao.Sanders.Kissane.Towery/_git/F22-Team21-Sprenkle.Zhao.Sanders.Kissane.Towery
   ```

5. Switch your directory to the repository that you cloned.

Working with the code

> 👤 Depending on the IDE you use (e.g Visual Studio Code) the latter half of these steps are done for you.

1. Browse to the folder on your computer where you cloned the repo and open the `README.md` file in your editor of choice.

2. Make some changes, for example add `<yourName> - This is my first edit.` to the file, and save and close the file.

3. In the Git command window, navigate to the cloned directory.

4. Commit your changes by entering the following command in the Git command window:

   a. When using `git commit`, `-a` means to commit all changed files, and `-m` specifies a commit message.

   ```
   git commit -a -m "My first commit"
   ```

5. Push your changes up to the Git repo by entering the following command into the Git command window: `git push`

6. Switch back to the web portal and select **History** from the **Code** view to view your new commit. The new repo has two commits. The first is the commit where the README and .gitignore were added when the repo was created. The second is the commit you just made.