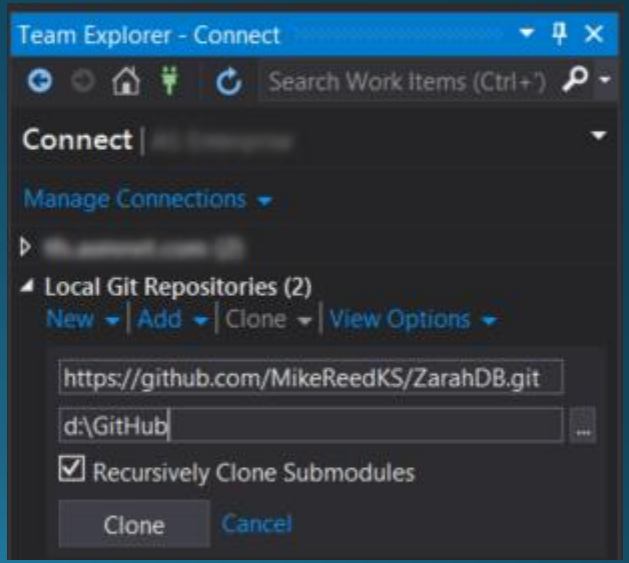
Get, modify and check-in ZarahDB code

GitHub Introduction

Connect to GitHub in Visual Studio

- In Team Explorer, click the plug just right of the house icon to navigate to the “Connect” panel.
- Click “Clone” just to the right of “Add”.
- Fill in http://GitHub.com/MikeReedKS/ZarahDB.git
- Fill in D:\GitHub
  (Or wherever you want the code to live)
- Keep it recursive.
- Click the “Clone” button.
- You now have access to the source code!


Team Explorer - Connect
Search Work Items (Ctrl+')
Connect
Manage Connections
Local Git Repositories (2)
New | Add | Clone | View Options
Clone
Cancel

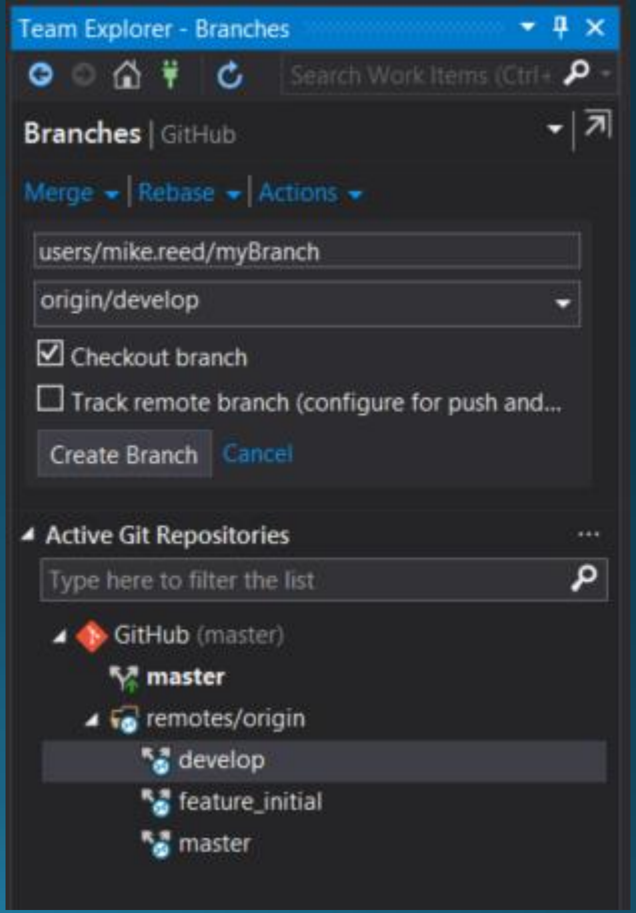
https://github.com/MikeReedKS/ZarahDB.git

d:\GitHub

☑ Recursively Clone Submodules

Create your own working branch

- Navigate to the "Branches" panel using the down arrow to the right of the heading "Connect". This is the easiest way to change panels.
- Open the "remotes/origin" node so that you can see the "develop" branch.
- Right click the "develop" branch and choose "New Local Branch From"
- Enter "users/your.name/NameOfYourBranch" (Make sure the "Track" checkbox is unchecked)
- Click the "Create Branch" button.
- You now have a copy of the develop branch locally.



Team Explorer - Branches

Search Work Items (Ctrl+

Branches | GitHub

Merge | Rebase | Actions

users/mike.reed/myBranch

origin/develop

☑ Checkout branch

□ Track remote branch (configure for push and...

Create Branch Cancel

Active Git Repositories

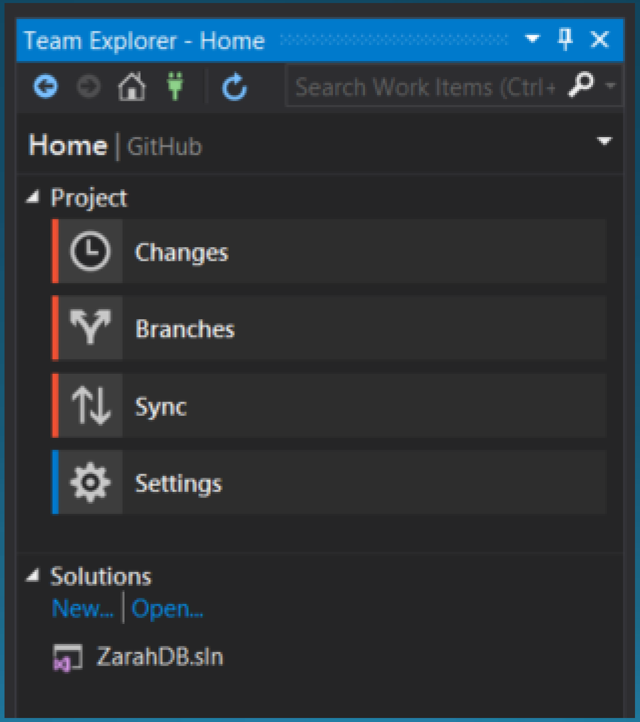
Type here to filter the list

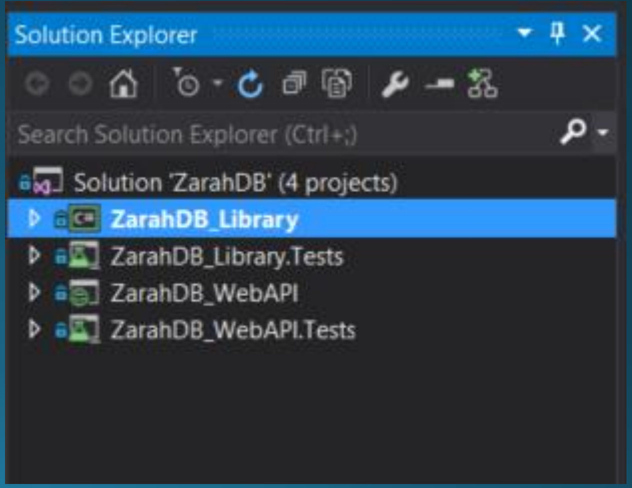
- GitHub (master)
  - master
  - remotes/origin
    - develop
    - feature_initial
    - master

Open the Project

- Navigate to the “Home” panel.
- Double-Click “ZarahDB.sln” to open the solution.
- The solution should contain four projects.

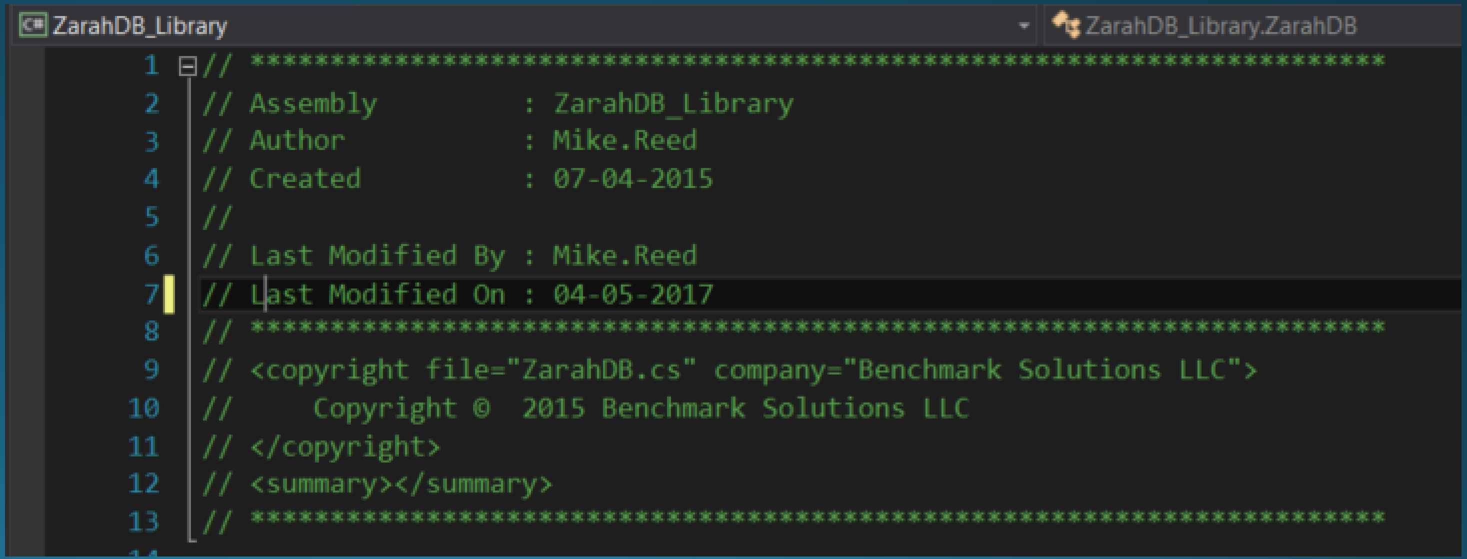
1. The Library (50K DLL)
2. Library Tests
3. The WebAPI
4. WebAPI tests.





Make Changes – Please!

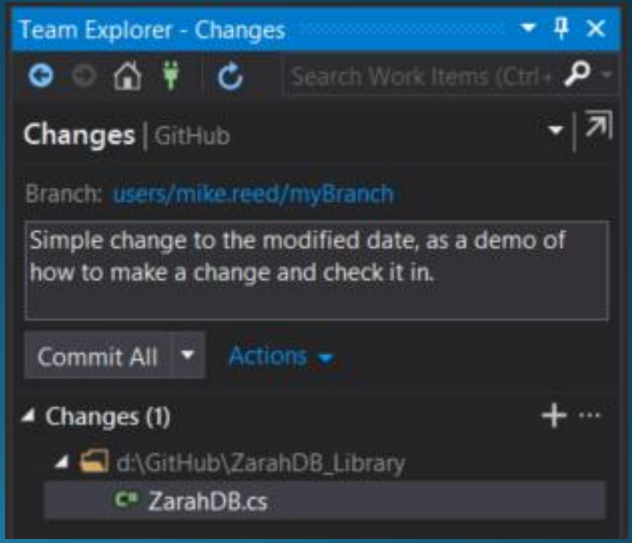
- If you are a C# developer, we invite you to improve the project.
- Edit, add, delete files. Change anything that makes the project better. If your changes will be substantial, you might want to chat with Mike Reed or other contributors to assure that the changes will have a high likelihood of making it into the final product.
- Tests that increase code coverage are always welcomed.
- Make sure to include tests covering any new code you contribute.



```
// ***********************************************************************
// Assembly         : ZarahDB_Library
// Author           : Mike.Reed
// Created          : 07-04-2015
//
// Last Modified By : Mike.Reed
// Last Modified On : 04-05-2017
// ***********************************************************************
// <copyright file="ZarahDB.cs" company="Benchmark Solutions LLC">
//     Copyright ©  2015 Benchmark Solutions LLC
// </copyright>
// <summary></summary>
// ***********************************************************************
```

Commit Changes Locally

- Git requires that changes made locally be committed before they can be moved out of your private source repository.
- Navigate to the “Changes” panel in Team Explorer. You will see a list of each file you changed.
- Add a comment about the changes.
- Click the “Commit All” button. This locks in your changes locally as ones you have approved.
- You can make as many commits to your local branch as needed to finish all the changes you wish to make.

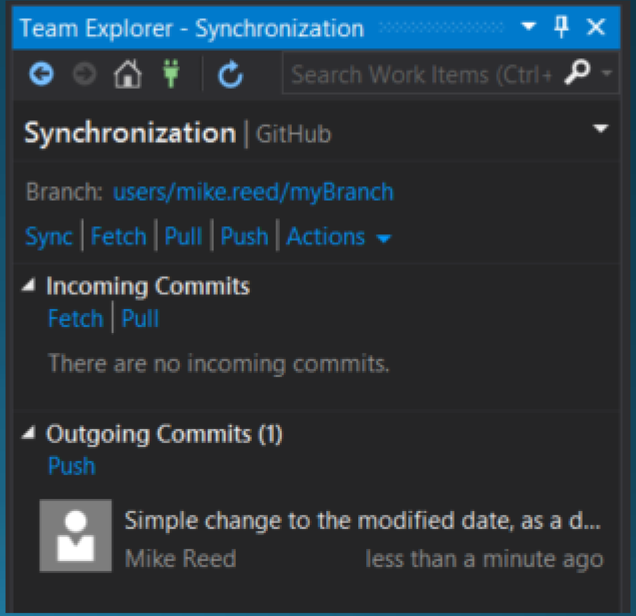


Branch: users/mike.reed/myBranch

Simple change to the modified date, as a demo of how to make a change and check it in.

Publish Local Changes to Remote

- Share your changes with the community by pushing them to the remote repository at GitHub.
- Navigate to the “Synchronization” panel.
- You should see one outgoing commit for each commit you created.
- Click “Publish” or “Push” just under the “Outgoing Commits” heading.
  “Publish” means you have never pushed to this branch before, while “Push” means to update the remote branch with even more changes.



Team Explorer - Synchronization

Synchronization | GitHub

Branch: users/mike.reed/myBranch

Sync | Fetch | Pull | Push | Actions

Incoming Commits

Fetch | Pull

There are no incoming commits.

Outgoing Commits (1)

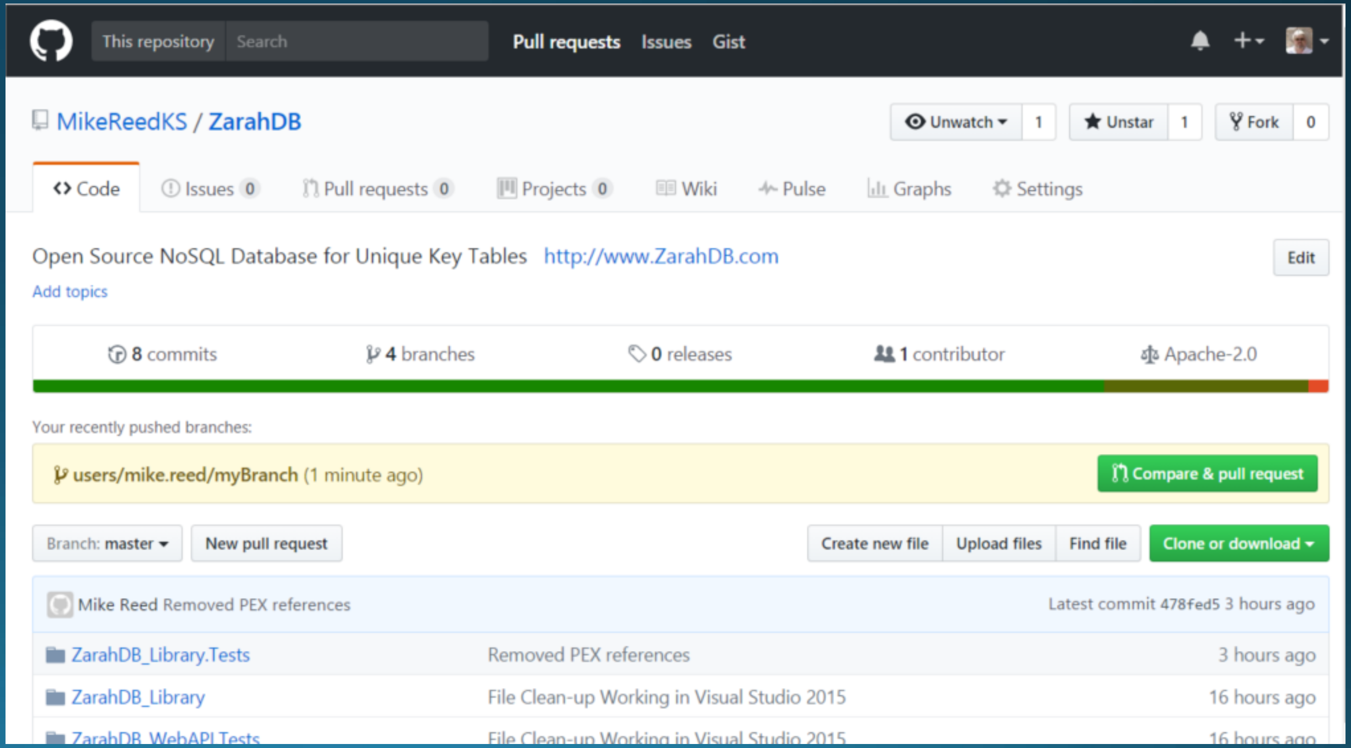
Push

Simple change to the modified date, as a d...

Mike Reed less than a minute ago

Initiate a Pull Request

- Go to https://github.com/MikeReedKS/ZarahDB and you should see that you have a recently pushed branch awaiting a pull request.
- Click the “Compare & pull request” button.
- A pull request is how you ask to have your changes merged into the develop branch on the remote repository.



Open Source NoSQL Database for Unique Key Tables http://www.ZarahDB.com

Your recently pushed branches:

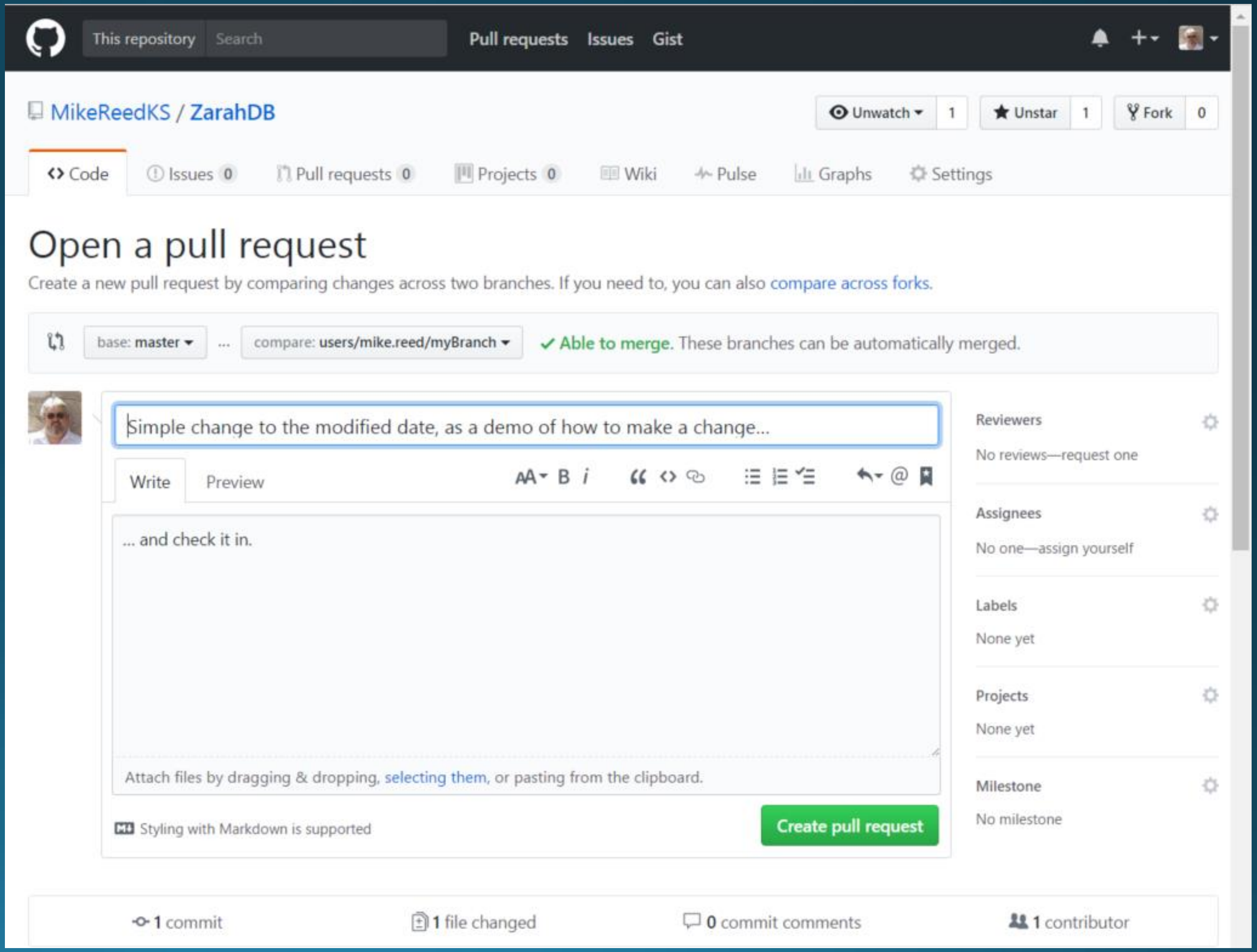
users/mike.reed/myBranch (1 minute ago)

Mike Reed Removed PEX references — Latest commit 478fed5 3 hours ago

ZarahDB_Library.Tests	Removed PEX references	3 hours ago
ZarahDB_Library	File Clean-up Working in Visual Studio 2015	16 hours ago
ZarahDB_WebAPI.Tests	File Clean-up Working in Visual Studio 2015	16 hours ago

Create a Pull Request

- Add any additional comments to help those reviewing your pull request to know exactly what you changed, added, or deleted.
- Make sure your base is set to “develop”.
- Click the “Create pull request” button.



Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also compare across forks.

base: master ... compare: users/mike.reed/myBranch ✓ Able to merge. These branches can be automatically merged.

Simple change to the modified date, as a demo of how to make a change...

... and check it in.

Attach files by dragging & dropping, selecting them, or pasting from the clipboard.

Styling with Markdown is supported

Reviewers: No reviews—request one

Assignees: No one—assign yourself

Labels: None yet

Projects: None yet

Milestone: No milestone

1 commit | 1 file changed | 0 commit comments | 1 contributor

Code Review and Merge

- Administrator(s) will review your pull request and if approved it will be merged in to the “develop” branch.
- When a new release is ready, the “develop” branch, with your changes among others, will be merged into the “master” branch.
- The master branch is then used to create a new NuGet package and the NuGet.org package is then updated.
- This is all done by those approved to help at this higher level, you don’t have to worry about these steps, but it helps to be aware of them.

Thank you in advance!!!