

# Push Changes

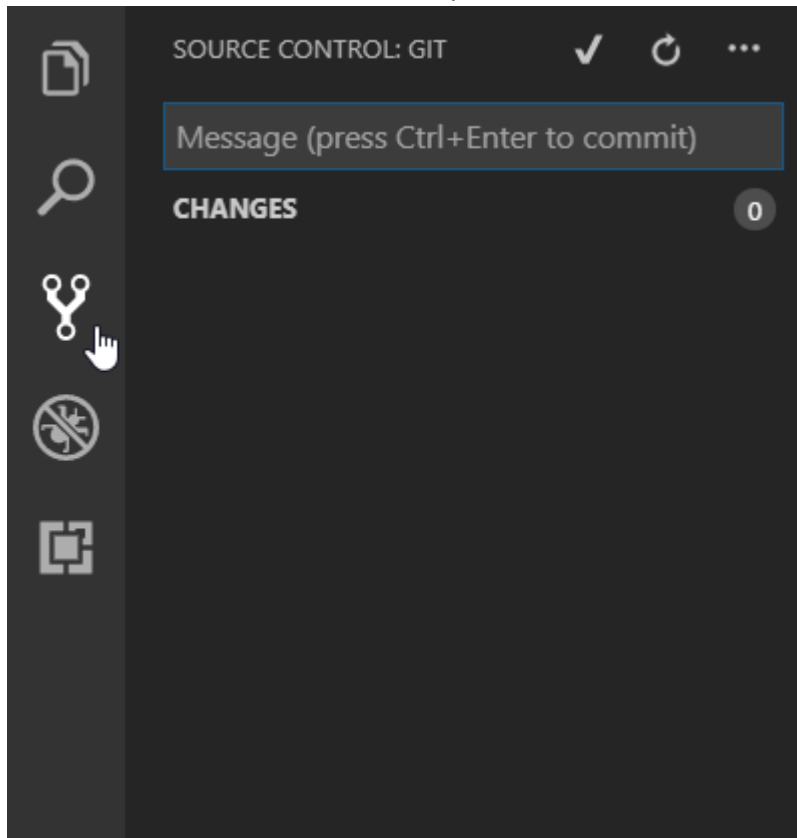
---

When you want to store a change on your remote repository, you will have to **push** those changes up. When you push your changes, they will appear on the server.

## Adding, Committing, and Pushing in Visual Studio Code

If you have Git installed, Visual Studio Code has some built-in tools that make it easy to interact with your repository. First, make sure you have opened the proper root-level folder for your local repository clone.

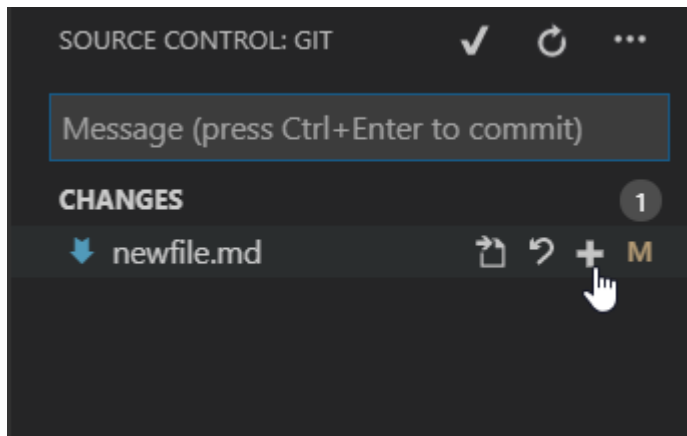
To view the various source control options, click the Source Control icon on the left menu:



Then, if you make a change to one of the files in your local repository, it will appear in the source control pane.

### Stage (add) the changes for commit

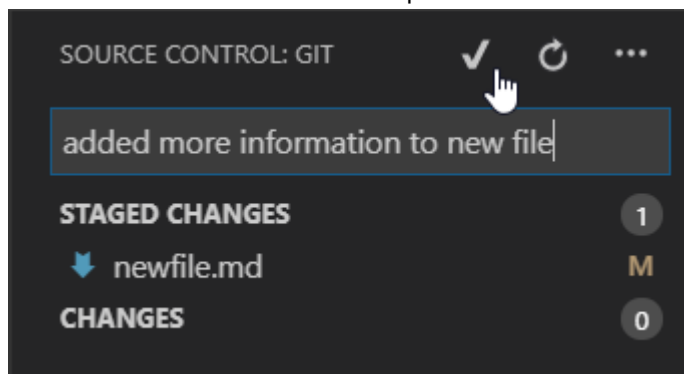
If you have a change and you are ready to commit it, you first have to **add** or **stage** the change for commit. In the source control pane, click the *plus* icon next to the file to add it to the commit:



## Commit the change

After you have added the change, it will be *staged* for commit. From there, you simply have to commit it!

1. Type a **commit message** in the box describing your change
2. Click the *checkmark* icon to complete the commit

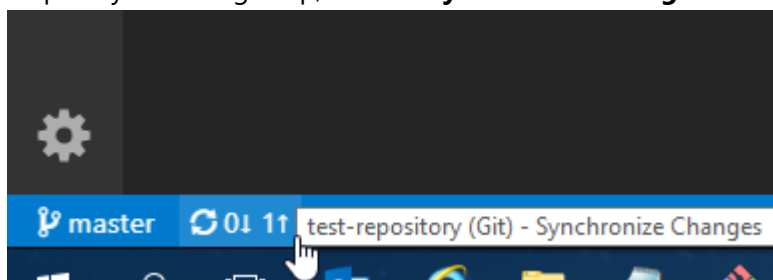


Once your change is committed, your local repository will include the change. The next step is to sync the change with the remote repository, so it's actually in GitHub.

## Push the change

Visual Studio Code has an interesting tool for pushing changes up to the remote repository. Instead of simply pushing changes, it actually *synchronizes* the local and remote repositories. This means that it pushes any commit changes up, and also pulls down any changes from the server that may exist. For now, you'll only use the tool to push, but it works for pulling too.

To push your changes up, use the **Synchronize Changes** button on the blue bar at the bottom of the window:



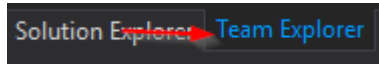
The button should have a recycle/refresh symbol, a number and a down arrow, and another number and an up arrow. The down number corresponds with how many new commits exist on the server, and the up number corresponds with how many new commits exist in the local repository.

Once you click that button, you will have to enter your credentials. After that, your changes should exist in the server repository! You can go to your repository on GitHub and make sure the changes are there.

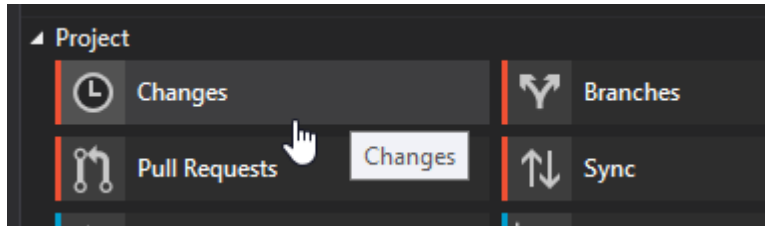
## Adding, Committing, and Pushing in Visual Studio

If you have Git installed, Visual Studio has some built-in tools that make it easy to interact with your repository.

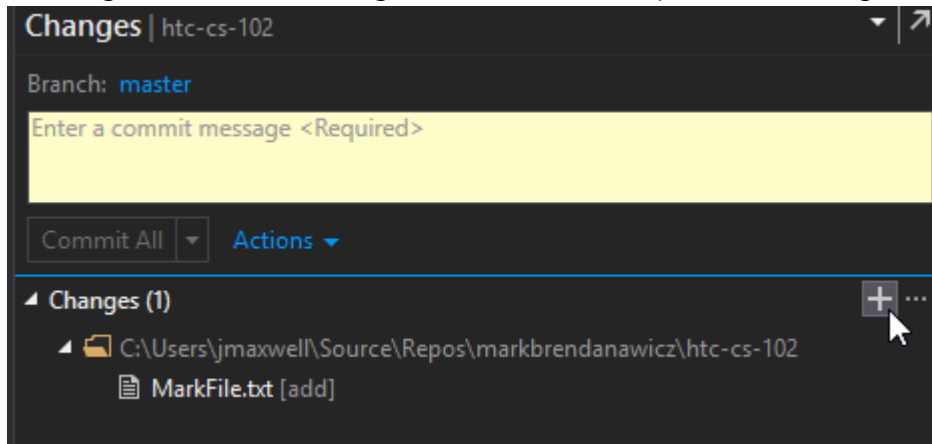
1. Open the **Team Explorer** pane



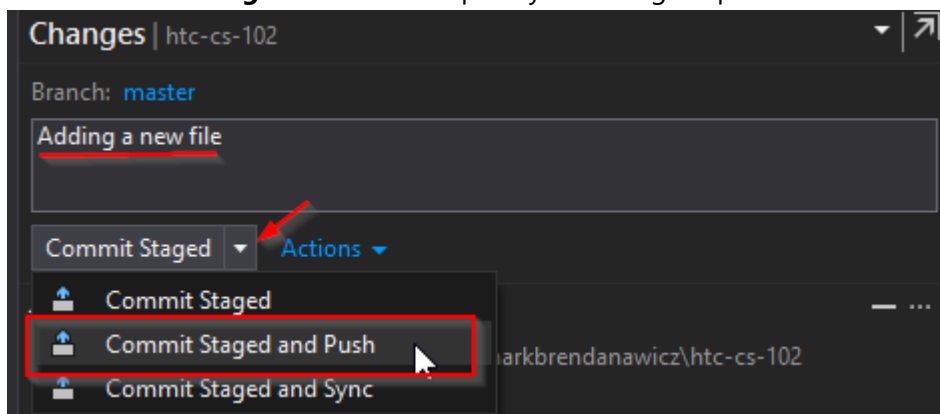
2. Under "Project", click **Changes**



3. In Changes, next to the "Changes" section, click the "plus" icon to Stage All changes



4. In the commit message box, enter a message
5. On the **Commit Staged** button, click the dropdown
6. Select **Commit Staged and Push** to push your changes up to the server



Once you click that button, your changes should exist in the server repository! You can go to your repository on GitHub and make sure the changes are there.

## Adding, Committing, and Pushing via the Command Line

First, make sure your present working directory is your repository's folder. Then, use the commands below to interact with git.

## Status

To see the current status of your local repository, you can use the `git status` command:

```
git status
```

## Add all changes in the repository

This command will take any changes you currently have, and stage them for commit:

```
git add .
```

## Commit all changes

Once all of your changes are staged for commit, you can use the `git commit` command to commit them. Replace `COMMIT MESSAGE HERE` with a message regarding your changes:

```
git commit -m "COMMIT MESSAGE HERE"
```

## Push all changes

Once your changes are committed, you can push them up to the server repository. Use the `git push` command to accomplish this:

```
git push
```

Note: you will have to enter your credentials in order to push to the server repository. After that, your changes should exist in the server repository! You can go to your server repository on GitHub and make sure the changes are there.