

Getting Started with Git

Jeremy Clark

www.jeremybytes.com

What is Git?

- Distributed Source Control
- Works Off-Line
 - Full functionality on the local machine
- Supports Remote Repositories
 - GitHub is the most popular
- Full Source Control Features
 - Change Tracking
 - Branching / Merging
 - History
 - And more...

Centralized vs. Distributed

Centralized Source Control

- TFS, Visual Source Safe, others
- Central server is required
- Server has the “true” copy
 - Checkout / locked files
- Server has full history
- Limited off-line features

Distributed Source Control

- Git, Mercurial, others (Git has won)
- Server is optional
- Each user has a “true” copy
 - No locked files
- Each user has full history
- Full features available off-line

Interacting with Git

Command Line

```
C:\Windows\system32\cmd.exe
C:\Development\Sessions\Git\RunThrough\LambdaExpressions>git init
Initialized empty Git repository in C:/Development/Sessions/Git/RunThrough/LambdaExpressions/.git/

C:\Development\Sessions\Git\RunThrough\LambdaExpressions>git status
On branch master

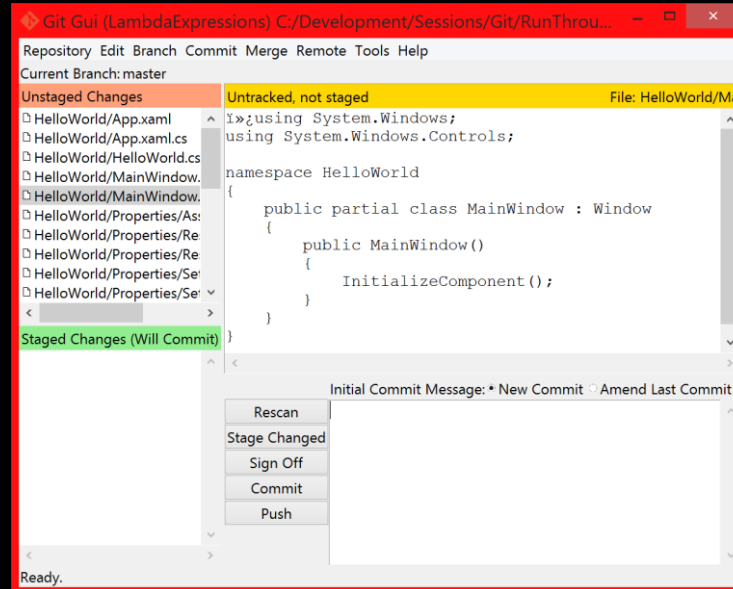
Initial commit

Untracked files:
  (use "git add <file>..." to include in what will be committed)

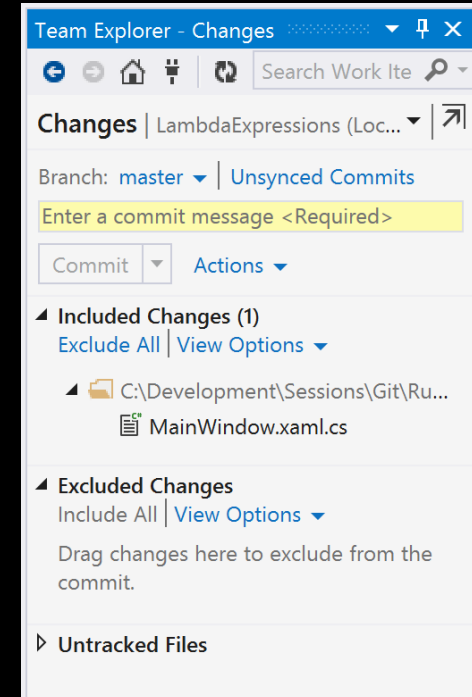
        HelloWorld/
        LambdaExpressions.sln
        LambdaExpressions.v12.suo

nothing added to commit but untracked files present (use "git add" to track)
C:\Development\Sessions\Git\RunThrough\LambdaExpressions>
```

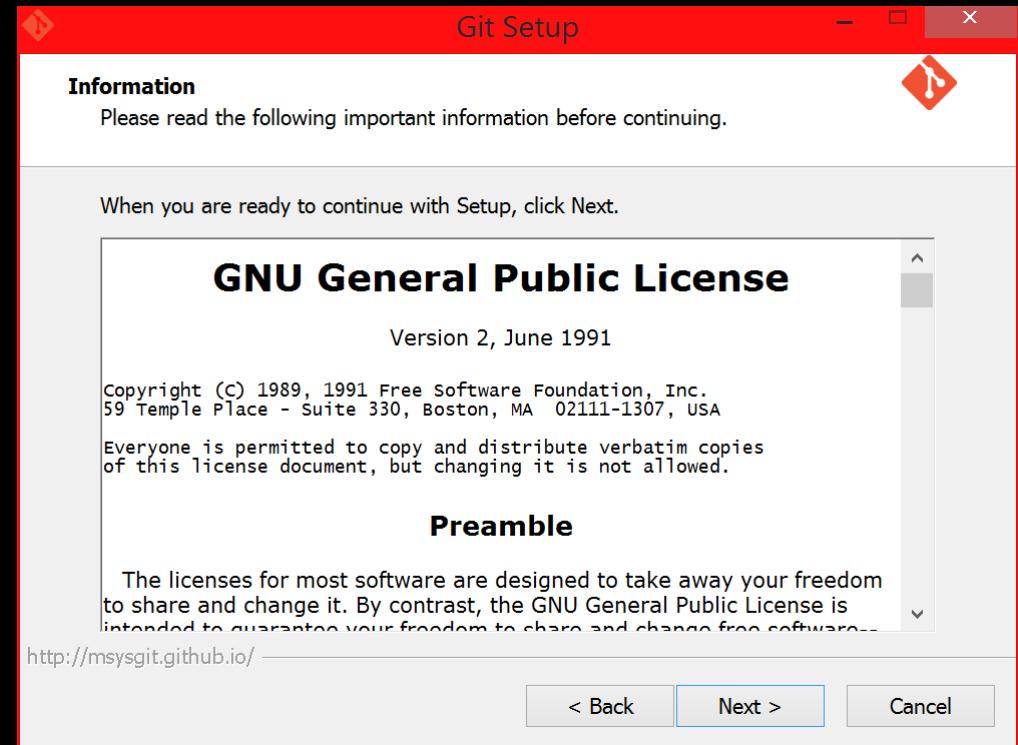
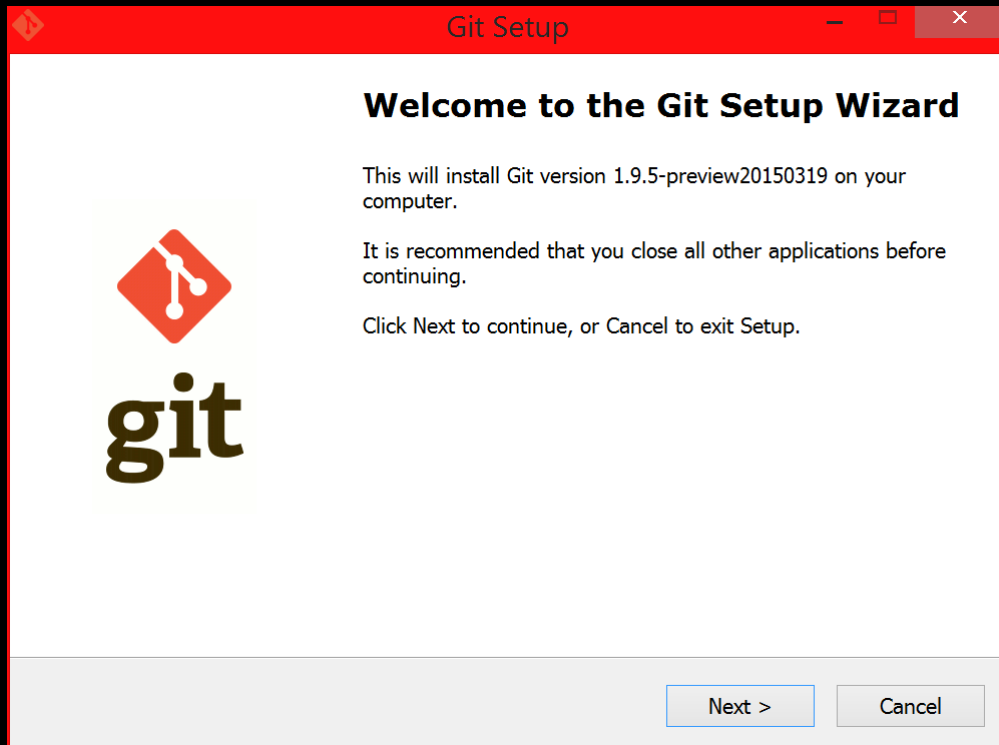
Git GUI



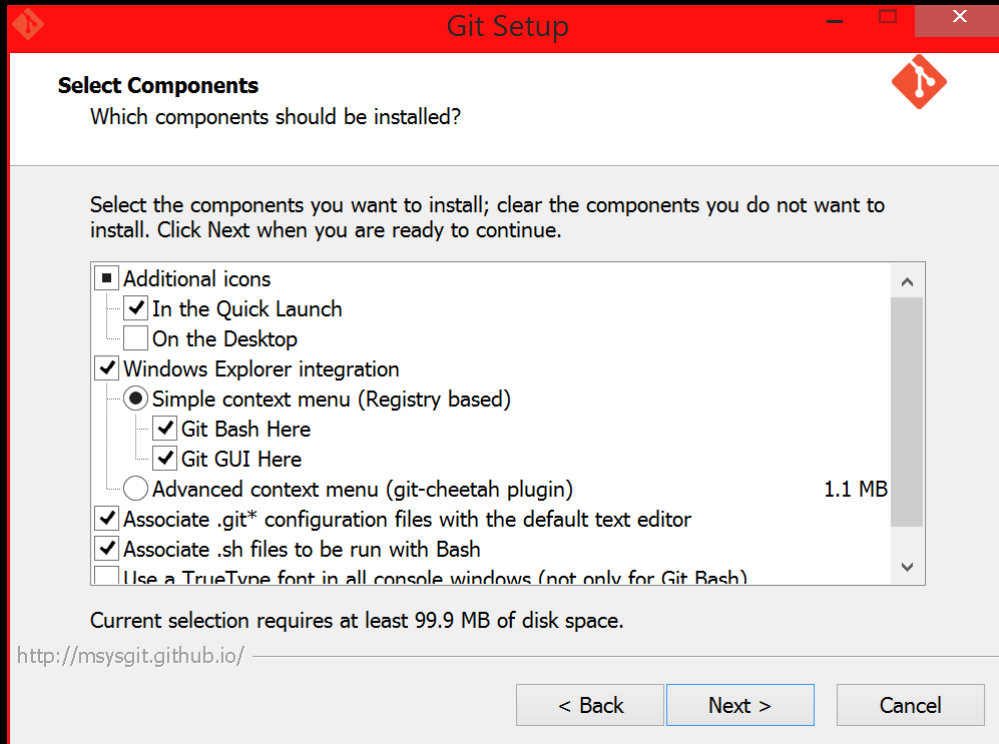
Visual Studio 2013



Installation

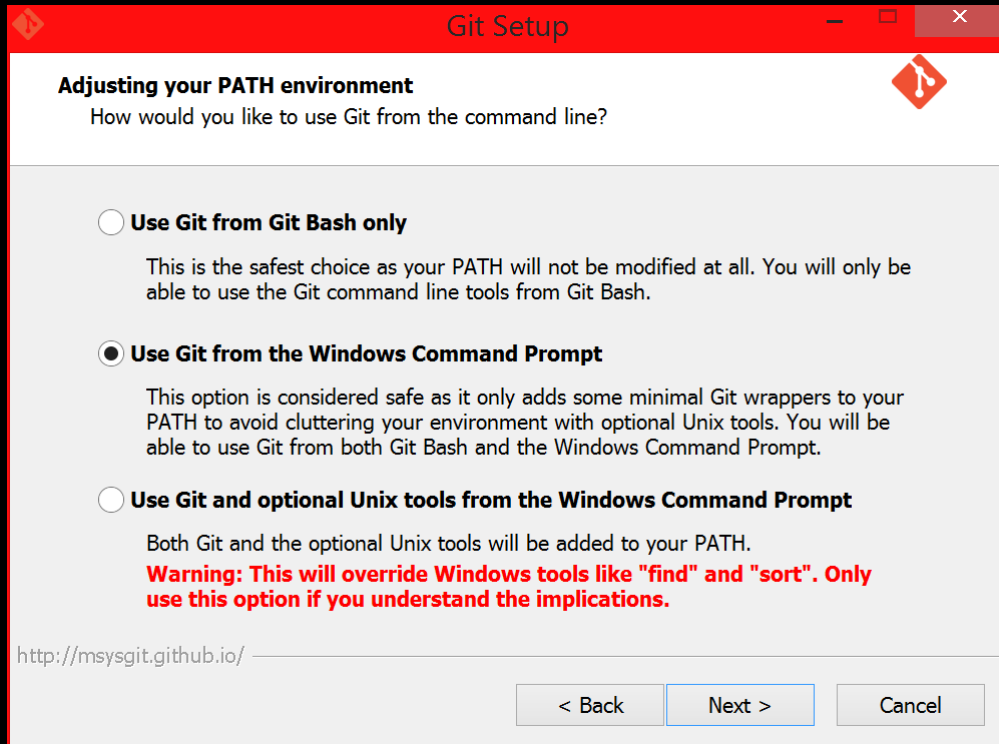


Installation



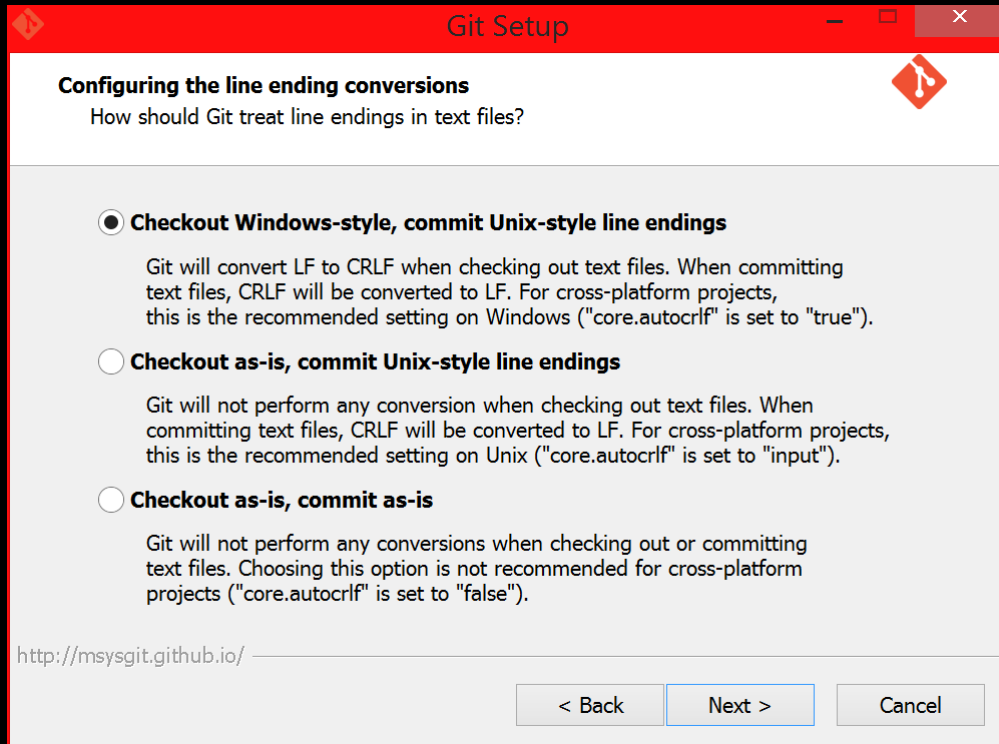
- Git Bash Here
 - Adds right-click menu to open Git command environment
 - bash is a Unix shell
 - *I don't use this
- Git GUI Here
 - Adds right-click menu to open Git graphical environment
 - *I don't use this

Installation



- Use Git from Windows Command Prompt
 - Add "git" to the path
 - Allows git to be used from the command prompt
 - *I use this

Installation



- Line Endings
 - Windows-style = CRLF
 - Carriage return & line feed
 - Unix-style = LF
 - Line feed
- This will impact code editors that don't understand Unix-style.
- This will impact "diff" tools that do file comparisons.

Key Commands

- `init`
 - Create a repository
- `add / rm`
 - Include / exclude files from the repository (tracking)
- `.gitignore`
 - File types to ignore
- `status`
 - View unsaved changes
- `commit`
 - Save changes

Branching

- branch
 - Create a new branch
- checkout
 - Change to a different branch
- merge
 - Update branch with changes from another branch
- stash
 - Temporary holding area (like “shelve”)
 - Can create a new branch from a stash

Remote Repositories

- remote
 - Configure a remote location
- push
 - Move changes from local repository to remote repository
- pull
 - Move changes from remote repository to local repository

Demo

The image shows a Visual Studio IDE with a red-bordered command prompt window in the foreground. The command prompt displays the output of a `git log` command, showing a commit by Jeremy Clark. The background shows the Visual Studio interface with a C# code file open, displaying a `MainWindow` class that inherits from `Window` and has a `ClickMeButton_Click` event handler that shows a `MessageBox` with the text "Hello World!". The Team Explorer on the right shows the project structure and the current commit.

```
C:\Windows\system32\cmd.exe
C:\Development\Sessions\Git\RunThrough\LambdaExpressions>git log
commit 44c8ec014786191b48f5c4792e501e3dcdda72b2
Author: Jeremy Clark <jeremy@jeremybytes.com>
Date:   Wed Jun 17 10:38:26 2015 -0700

    Initial project

C:\Development\Sessions\Git\RunThrough\LambdaExpressions>git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git checkout -- <file>..." to discard changes in working directory)

        modified:   HelloWorld/MainWindow.xaml.cs

no changes added to commit (use "git add" and/or "git commit -a")
C:\Development\Sessions\Git\RunThrough\LambdaExpressions>
```

```
using System.Windows;
using System.Windows.Controls;

namespace LambdaExpressions
{
    class MainWindow : Window
    {
        MainWindow()
        {
            InitializeComponent();
            ClickMeButton.Click += ClickMeButton_Click;
        }

        private void ClickMeButton_Click(object sender, RoutedEventArgs e)
        {
            MessageBox.Show("Hello World!");
        }
    }
}
```

Links

- Code School
<https://try.github.io>
- Git Documentation
<https://git-scm.com/docs>
- MSDN Visual Studio Documentation
<https://msdn.microsoft.com/en-us/library/hh850437.aspx>
- Git .gitignore
<https://github.com/github/gitignore>



Thank You!

Jeremy Clark

- <http://www.jeremybytes.com>
- jeremy@jeremybytes.com
- @jeremybytes