# Quick Installation Notes for Windows 10

**Version: 1.0**

# Introduction

This document provides a quick listing of the tools needed and basic install instructions for each -- which is used throughout this course. Before you get started installing all the tools and software for this course, there are a few basic requirements. After that, I provide the general instructions for each tool used. Since this page is designed to aide the "get to the point" crowd, I keep my instructions as brief as possible.

In order to support the most recent version of Windows available, these instructions were tested using **Windows 10**. However, with some modification, these instructions will generally work for older versions of Windows.

# Getting Started and Common Tools

## Admin Rights

You need to have Administrator rights to your system. Most modern versions of Windows come with several "flavors" of user accounts -- only Administrators can install software.

## The Right Bits

Windows comes in two flavors: 32-bit and 64-bit. What's even more confusing -- you might have a 32-bit version on hardware able to run 64-bit software.

The fastest way to find out if you have 32 or 64-bits installed:

- Right-click on the *Start Menu*, this will display a pop-up menu
- Click on the *System* item
- Once the *System* window appears, look for the **System Type** entry under the *System* section. This should tell you if you have 32 or 64-bit version of Windows.

Make a note of this -- you'll want to install the 32-bit or 64-bit version of any software in order to best match your operating system and to have the best performance possible, when given the choice.

## Google Chrome

**Optional.**

I use Google Chrome for most of my courses. A few years ago, I would have strongly recommended or border-lined required the use of Chrome. However, most modern versions of all common browsers are adequate -- although the software engineer in me still prefers Chrome. For those wanting to follow along as closely as possible, install and use Chrome during this course. However, this is an *optional* step now, but I include it for completeness.

### Install for Windows

- Go to the Google Chrome Desktop page at https://www.google.com/chrome/browser/desktop
- Click on the **Download Chrome** button
- Accept the *Terms of Service* agreement (after reading, of course)
- Follow the instructions through the install process

# Git for Windows

**Required.**

Git is the source control tool used in this course. While Jenkins supports many other control control tools, Git is the most popular these days.

## Install on Windows 10

- Download Git for Windows directly from https://github.com/git-for-windows/git/releases/latest.
- Run the installer program, follow the defaults (recommend other choices in the video, but defaults are ok too)
- Open the **Git Bash** program, which is a Bash Shell terminal designed specifically for Git on Windows

### Configure Git

Git requires your name and email address before any real work can be done. It is best to just configure Git from the start.

```
git config --global user.name "Your Name"
git config --global user.email "your.email@your-place.com"
```

## Notepad++

**Optional.**

Windows comes with a text editor called **Notepad**, but it doesn't do much beyond allow you to edit text and many IT professionals prefer something more. I use a free and open-source program called **Notepad++** for most of my Windows based courses. If you are happy with Notepad, then this step is *optional*.

### Install

- Download Notepad++ from https://notepad-plus-plus.org/download
- Since there are many adverts on the page, ensure you select the **Notepad++ Installer** and not an AD by mistake.
- Once the installer has finished downloading, run the installer.
- Follow all the defaults through the install process with the following exceptions:
  - Check *Create Shortcut on Desktop* (personal choice)

### Notepad++ System-Wide

If you plan to use Notepad++ a lot, I highly recommend adding Notepad++ to your system's PATH environment variable. You can confirm weather or not this is needed by opening a **command prompt** or **Git Bash** and type `notepade++` and press the `enter` key. If Notepad++ launches, then no additional work is needed. If you get a `Command Not Found` or similar error, then add the Notepad++ install folder to the system **Path** variable.

### Bash Configuration

Open or create the **~/.bash_profile** file and add the following line:

```
alias npp='notepad++ -multiInst -nosession'
```

### Git Integration

Open Git Bash and issue the command:

```
git config core.editor "notepad++ -multiInst -nosession"
```

Then test it out by:

```
git config --global -e
```

## P4Merge on Windows

*P4Merge for Windows* which is a visual comparsion and merge resolution tool that integrates well with Git.

### Install

- Download P4Merge: Visual Merge Client from https://www.perforce.com/downloads/helix#clients
- Once the installer has finished downloading, run the installer.
- Follow all the defaults through the install process with the following exceptions:
  - During install, take care to only install the **P4Merge Visual Merge** client

## Configuration

Now, let's integrate P4Merge with Git. You'll need to know where P4Merge is installed on your system -- which is normally under *Program Files*. With the next series of commands, you may need to modify them slightly to fit your system.

Configure P4Merge as **Diff Tool** in Git:

```
git config --global diff.tool p4merge
git config --global difftool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"
git config --global difftool.prompt false
```

Configure P4Merge as **Merge Tool** in Git:

```
git config --global merge.tool p4merge
git config --global mergetool.p4merge.path "C:/Program Files/Perforce/p4merge.exe"
git config --global mergetool.prompt false
```

The above commands should work, but some systems may require converting the paths to Unix friendly versions where C: is replaced with /c/.