



제품소프트웨어 패키징
(Git과 GitHub를 활용한 소스코드 관리)

Git 개념 및 설치



학습내용

- Git이란?
- Git 설치하기



학습목표

- Git이 개발된 역사와 특징을 설명할 수 있다.
- Git을 사용하기 위해 PC에 설치하고 환경 설정을 할 수 있다.



Git이란?

1. Git의 탄생

- 리눅스 토발즈가 리눅스 개발에 이용하기 위해 처음 개발한 소스코드 관리 시스템
- Git의 역사
 - 리눅스를 처음 개발할 당시 : 단순히 압축 파일 형태로만 소스코드를 관리함
 - 2002년 : BitKeeper 라고 불리는 상용 DVCS를 리눅스 개발에 사용함
 - 2005년 : BitKeeper의 무료 사용이 제한되면서 자체적인 버전 관리 도구 제작
- 개발 목적
 - 빠른속도
 - 단순한 구조
 - 완벽한 분산
 - 대형 프로젝트에서의 유용성



Git이란?

2. Git의 특징

- 델타가 아니라 스냅샷
 - 변경 사항을 등록하는 Commit 동작이 발생할 때마다 이전 상태와 비교하여 변경 점만 저장
 - 기존 VCS시스템보다 빠르고 저장공간을 적게 차지
- 거의 모든 명령을 로컬에서 실행
 - 서버의 연결 상태와 상관없이 로컬에 변경사항 반영
 - 최종 소스와의 동기화는 서버와 연결이 된 상태에서 수행
- Git의 무결성
 - 데이터를 저장하기 전에 해시값을 구해서 그것으로 데이터를 관리
- 데이터 추가 동작만 수행
 - 사용자 인터페이스상으로는 되돌리거나 수정하는 것 같은 느낌을 줄 수 있으나, 모두 추가되어 관리되므로 그 히스토리를 모두 추적할 수 있음
- Git의 데이터 상태 관리
 - Modified : 수정한 파일을 아직 로컬 데이터베이스에 적용하지 않은 상태
 - Staged : 현재 수정한 파일을 곧 적용할 것이라고 표시한 상태
 - Committed : 로컬 데이터베이스에 안전하게 저장 완료된 상태



Git이란?

3. Git의 기본 용어

- Repository
 - 저장소
 - 소스코드의 모든 내용을 저장
 - 각 소스코드의 변경 사항을 관리
- Working Tree
 - 원격저장소에 있는 모든 소스코드와 변경 정보를 복사한 내 컴퓨터의 작업 디렉토리의 현재 상태
- Directory
 - 원격저장소에 저장된 소스코드와 그 변경사항을 관리하기 위해 내 컴퓨터에 변경점에 대한 정보만을 따로 관리
- Branch
 - 소프트웨어 개발에서 다양한 버전이나 특정 기능들을 따로 구현해야 할 때에 기본 개발부분을 건드리지 않고 분기
- Merge
 - Branch를 생성해서 작업하던 내용이 Master Branch에 통합되어 반영되어야 할 때 기능 수행
 - 합치기 기능



Git 설치하기

1. Git 다운로드

1) 홈페이지 접속



2) 다운로드 진행

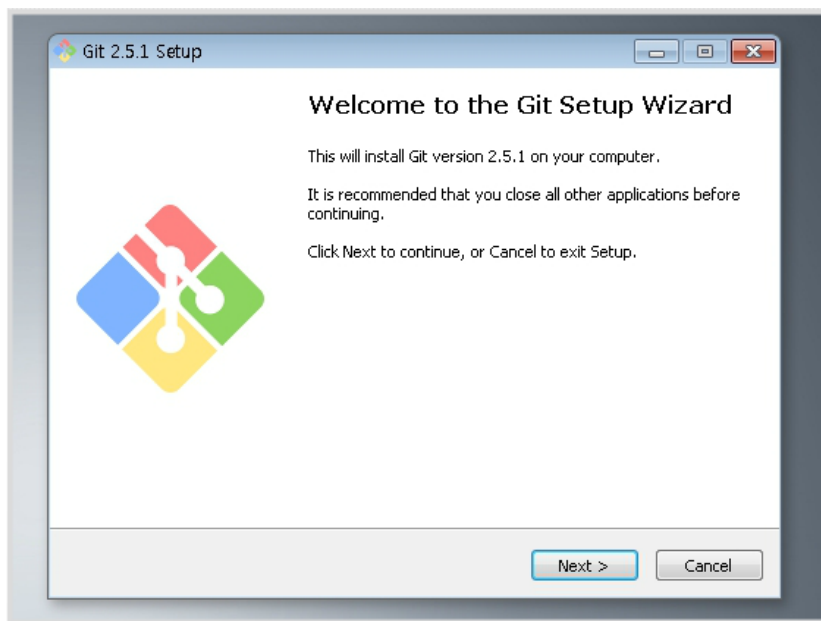




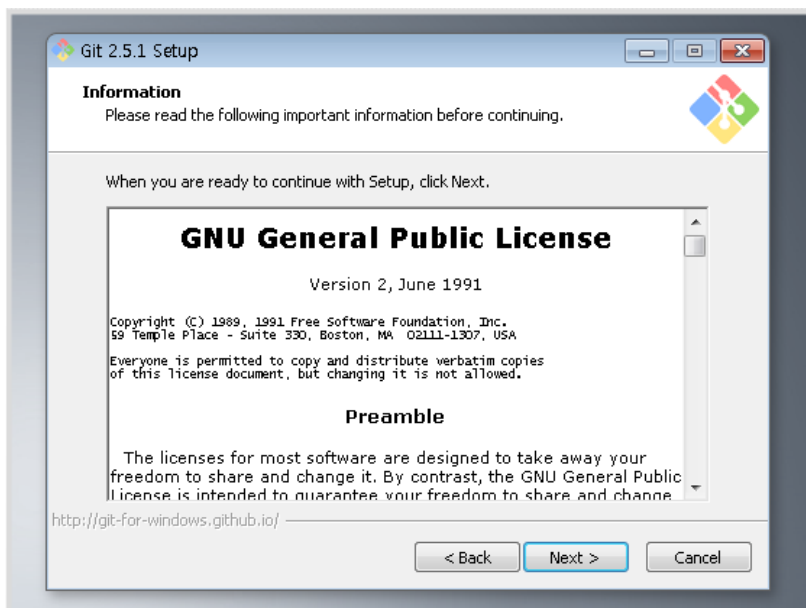
Git 설치하기

2. Git 설치하기

1) 실행파일 클릭



2) GNU 오픈소스 라이선스 확인

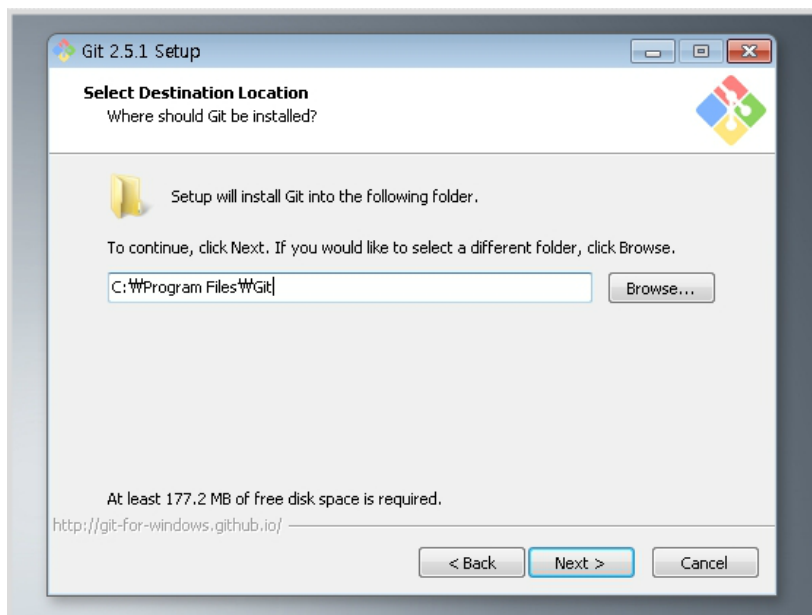




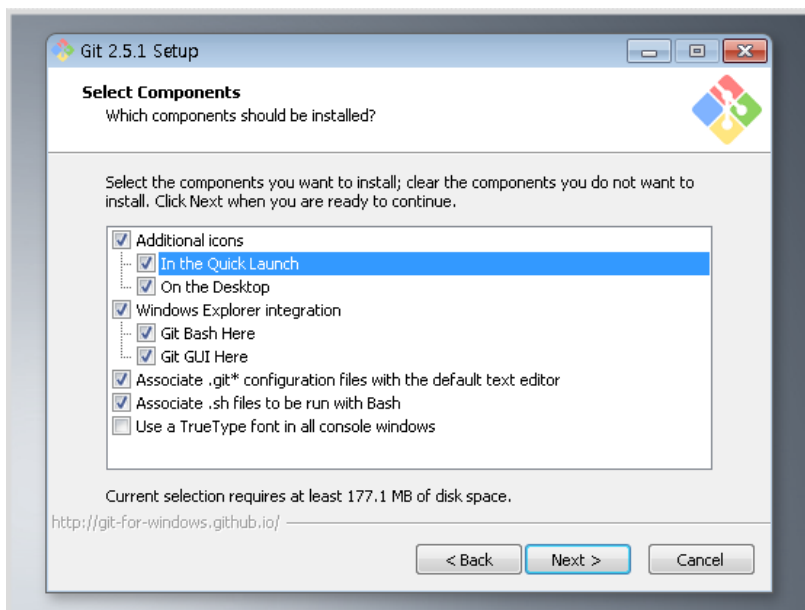
Git 설치하기

2. Git 설치하기

3) Git 설치 위치 설정



4) 컴포넌트 설정

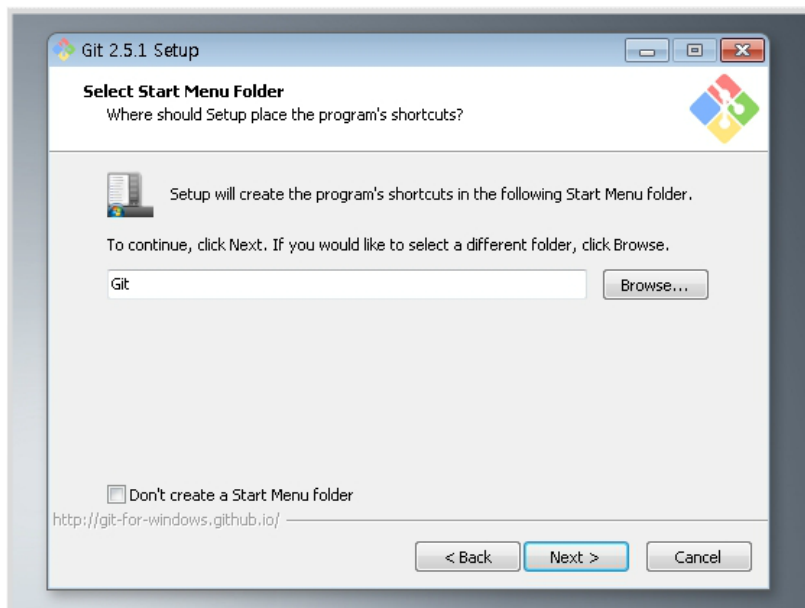




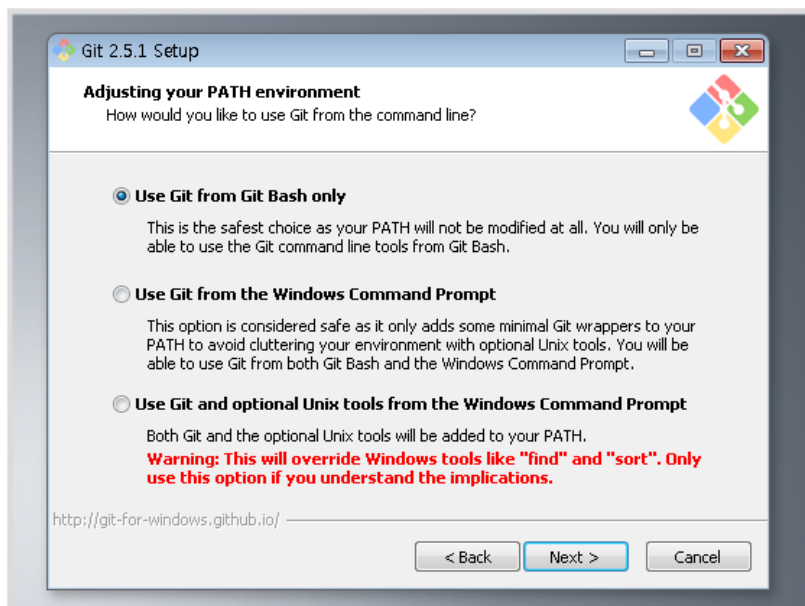
Git 설치하기

2. Git 설치하기

5) Start Menu Folder 위치 지정



6) 커맨드 창 스타일 설정

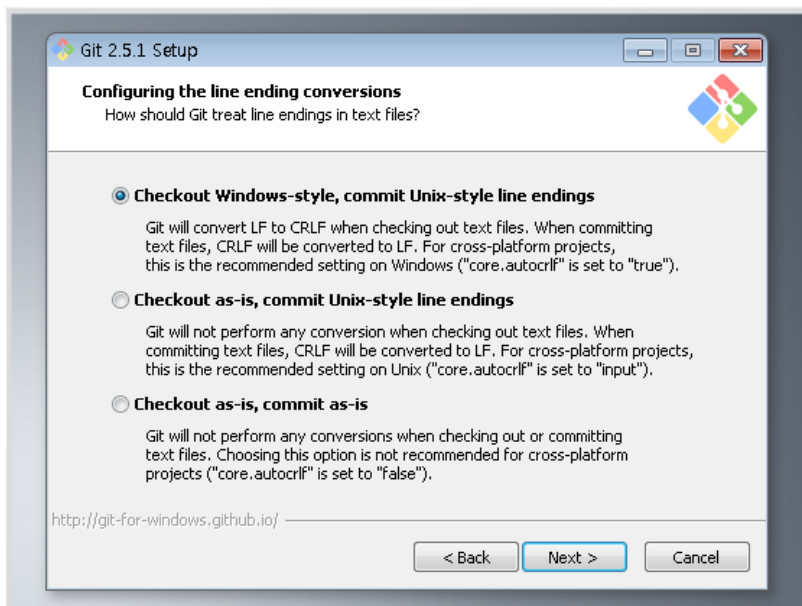




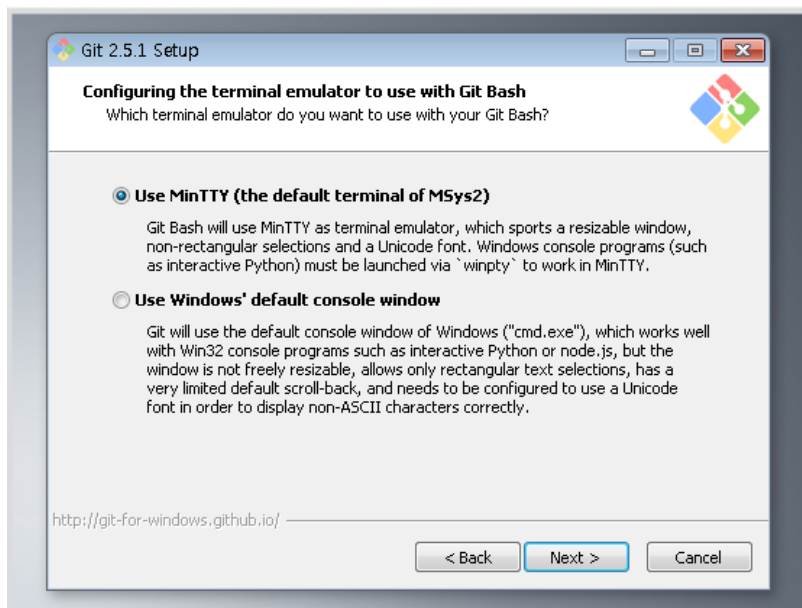
Git 설치하기

2. Git 설치하기

7) Checkout&Commit 스타일 설정



8) Git Bash창 설정

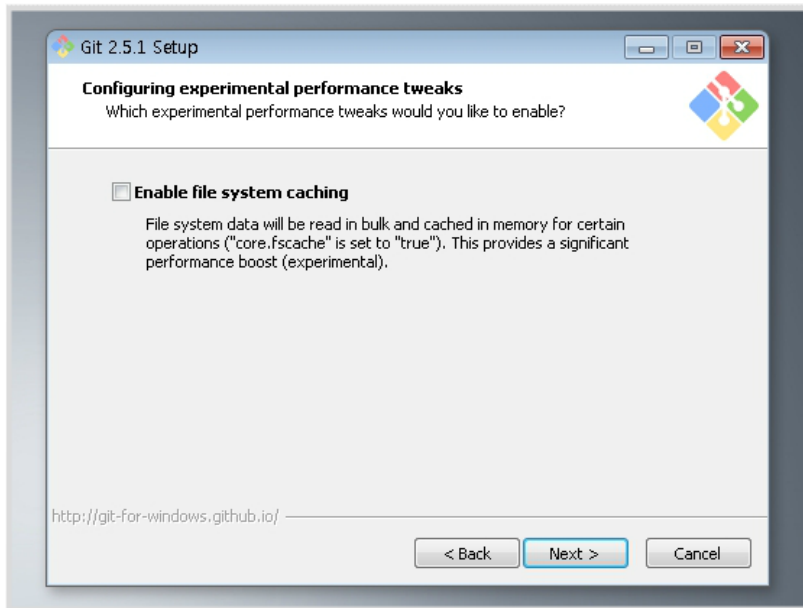




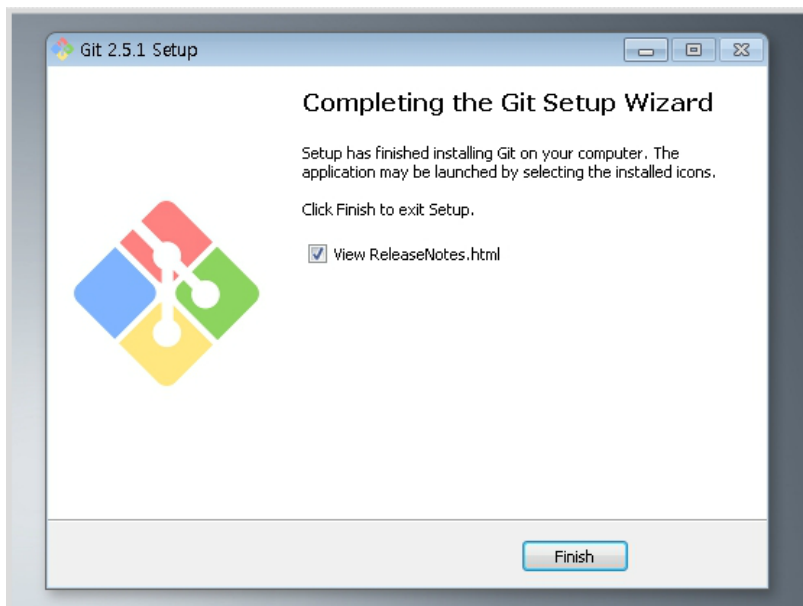
Git 설치하기

2. Git 설치하기

9) 파일 시스템 캐싱 사용 여부



10) 설치완료

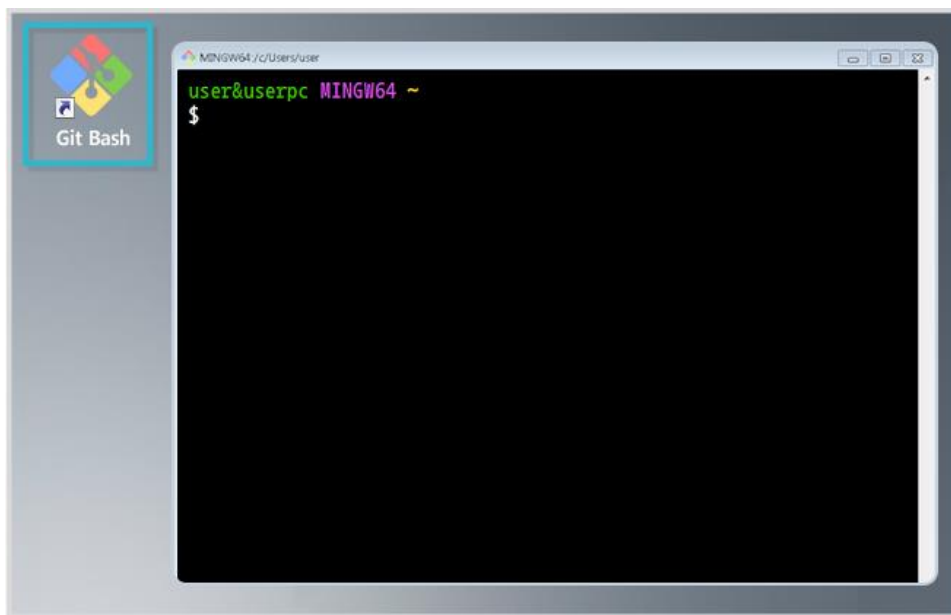




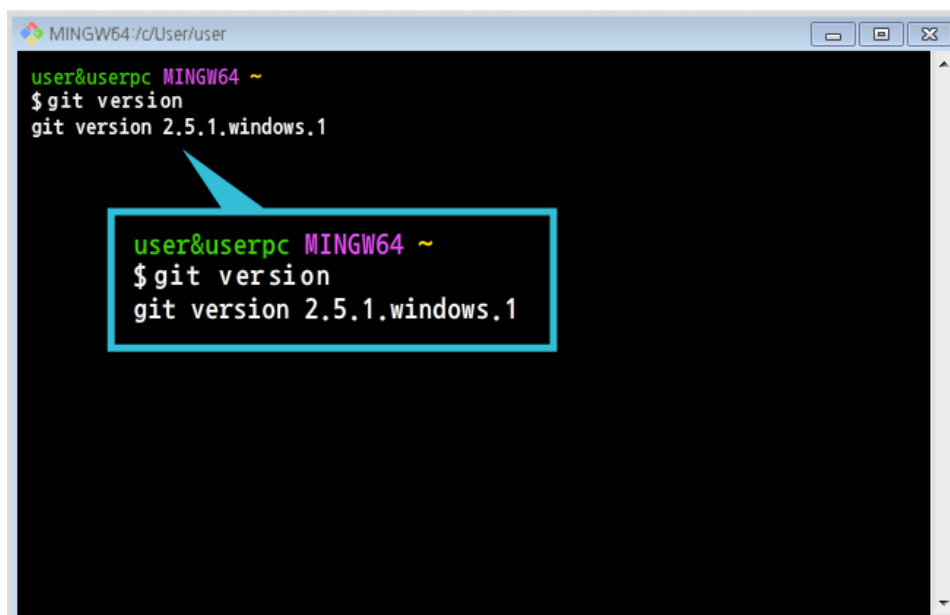
Git 설치하기

3. Git 환경설정

1) Git Bash 아이콘 실행



2) Git 버전 확인





Git 설치하기

3. Git 환경설정

3) 환경설정 데이터 확인

```
MINGW64:/c/User/user
user&userpc MINGW64 ~
$ git config --list
core.symlinks=false
core.autocrlf=true
color.diff=auto
color.status=auto
color.branch=auto
color.interactive=true
pack.packsizelimit=2g
help.format=html
http.sslcainfo=c:/program Files/Git/mingw64/ss1/certs/ca-bund1e.crt
diff.astextplain.textconv=astextplain

user&userpc MINGW64 ~
$ git config --global user.name "girrrMaster"

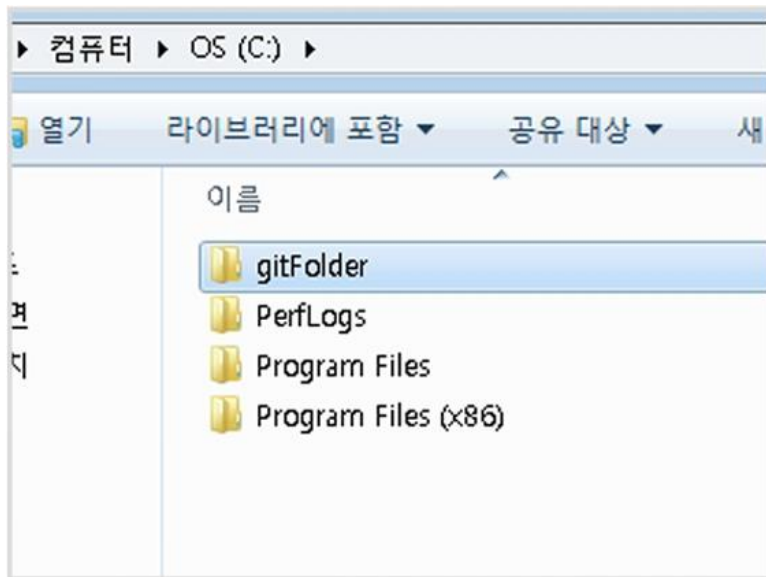
user&userpc MINGW64 ~
$ git config --global user.email "girrrr.master@gmail.com"
```



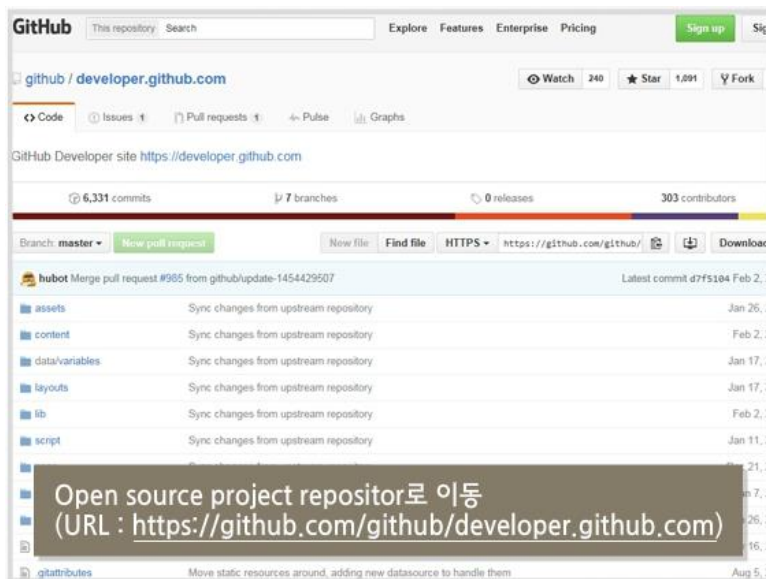
Git 설치하기

4. Git 설치 확인

1) 사용자 컴퓨터에 폴더 생성



2) 오픈소스 저장소 페이지 접속

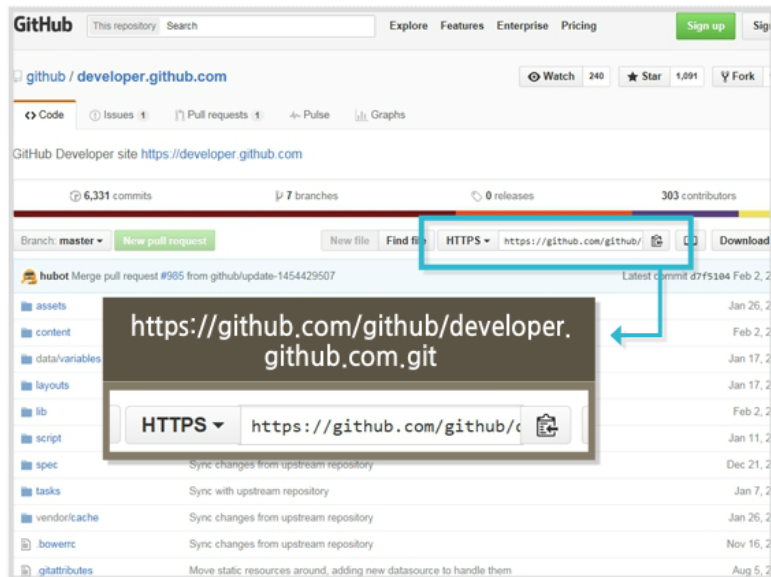




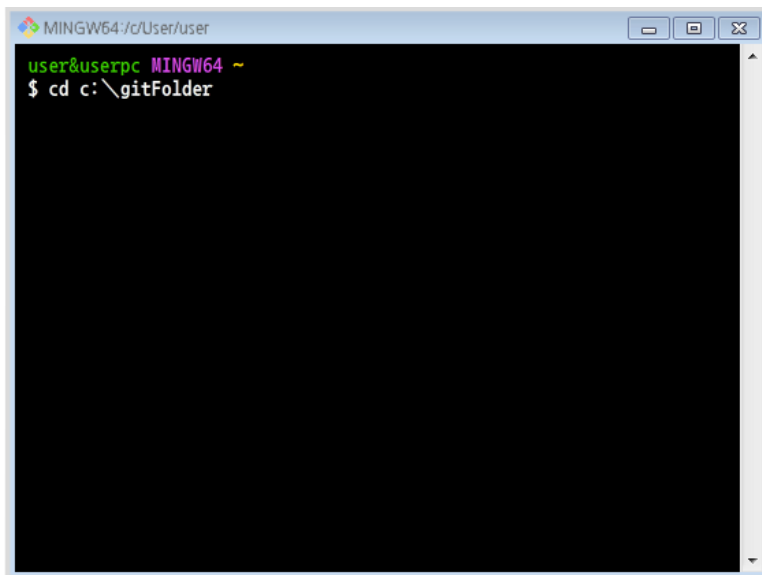
Git 설치하기

4. Git 설치 확인

3) HTTPS 주소 복사



4) 해당 디렉토리로 이동





Git 설치하기

4. Git 설치 확인

5) c:\gitFolder에 Repository를 Clone

```
MINGW64 ~/c:/User/user
user@userpc MINGW64 ~
$ cd c:\gitFolder
```

6) HTTPS 명령어 입력

```
MINGW64 ~/c:/User/user
user@userpc MINGW64 ~
$ cd c:\gitFolder

user@userpc MINGW64 /c/gitFolder
$ git clone http://github.com/github/developer.github.com.git
```




Git 설치하기

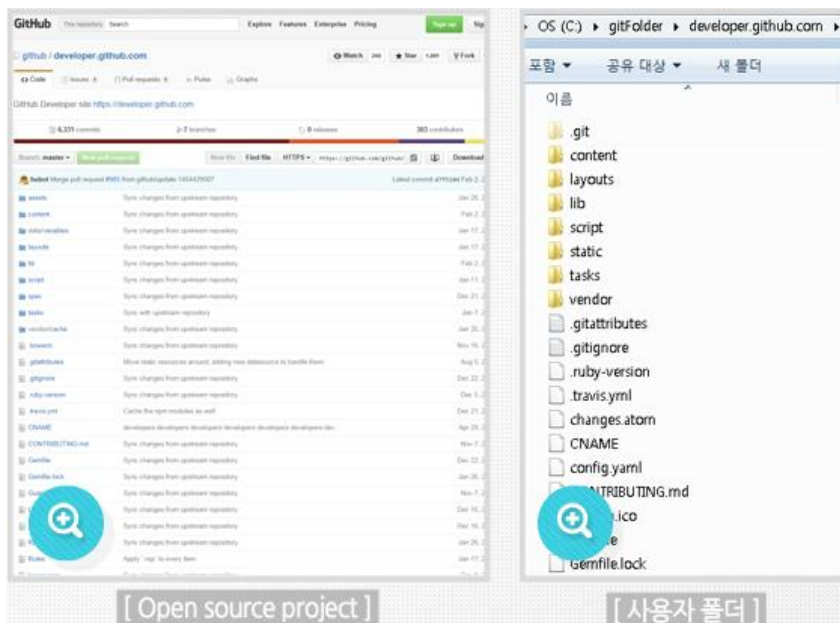
4. Git 설치 확인

7) 설치 완료 확인

```
MINGW64~/c:/User/user
user&userpc MINGW64 ~
$ cd c:\gitFolder

user&userpc MINGW64 /c/gitFolder
$ git clone http://github.com/github/developer.github.com.git
Cloning into 'developer.github.com' . . .
remote: Counting objects: 68322, done.
remote: Compressing objects: 100% (18/18), done.
remote: Total 68322 (delta 3), reused 0 (delta 0), pack-reused 68304
Receiving objects: 100% (68322/68322), 68.28 MiB | 8.10 MiB/s, done.
Resolving deltas: 100% (35086/35086), done.
Checking connectivity. . . done.
```

8) Clone한 Repository 비교





! 핵심정리



Git이란?

1. Git의 탄생

- Git은 리눅스 개발시 소스코드 관리를 목적으로 리눅스 토발즈에 의해 개발

2. Git의 특징

- 델타가 아니라 스냅샷
- 거의 모든 명령을 로컬에서 실행
- Git의 무결성
- Git은 데이터 추가 동작만 수행
- Git의 데이터 상태 관리
 - Modified, Staged, Committed의 세 가지 상태로 관리

3. Git의 기본 용어

- Repository : 소스코드의 모든 내용을 저장하는 곳
- Working Tree : 현재 작업중인 내 컴퓨터의 현재 상태
- Directory : 내 컴퓨터에 변경점에 대한 정보만을 따로 관리
- Branch : 다양한 버전이나 특정 기능 구현 시 기본 개발부분을 건드리지 않고 분기
- Merge : Master Branch에 통합되어 반영된 합치기 기능



! 핵심정리



Git 설치하기

1. 다운로드

- 홈페이지 접속 → 다운로드 진행

2. 설치하기

- 실행파일 클릭 → GNU 오픈소스 라이선스 확인 → Git 설치 위치 설정
→ 컴포넌트 설정 → Start Menu Folder 위치 지정
→ 커맨드 창 스타일 설정 → Checkout&Commit 스타일 설정
→ Git Bash창 설정 → 파일 시스템 캐싱 사용 여부 → 설치완료

3. 환경설정

- Git Bash 아이콘 실행 → Git 버전 확인 → 환경설정 데이터 확인

4. 설치확인

- 사용자 컴퓨터에 폴더 생성 → 오픈소스 저장소 페이지 접속
→ HTTPS 주소 복사 → 해당 디렉토리로 이동
→ c:\wgitFolder에 Repository를 Clone → HTTPS 명령어 입력
→ 설치 완료 확인 → Clone한 Repository 비교