git add

git add 명령은 작업 디렉토리의 변경 사항을 스테이징 영역에 추가합니다. Git에게 다음 커밋에서 특정 파일에 대한 업데이트를 포함하길 원한다는 것을 알려줍니다. 그러나 git add는 저장소에 중요한 영향을 미치지 않습니다. 변경 사항은 git commit을 실행할 때까지 실제로 기록되지 않습니다.

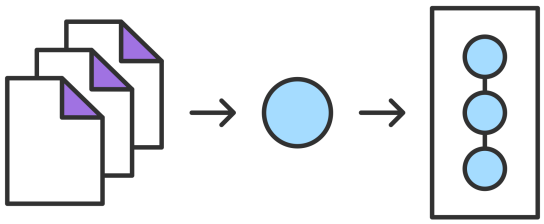
이 명령들과 함께 작업 디렉토리와 스테이징 영역의 상태를 보려면 git 상태가 필요합니다.

How it works

git add와 git commit 명령은 근본적인 Git 작업 흐름을 구성합니다. 이 명령은 팀의 공동 작업 모델에 관계없이 모든 Git 사용자가 이해해야하는 두 가지 명령입니다. 그것들은 저장소의 히스토리에 프로젝트 버전을 기록하는 수단이다.

프로젝트를 개발하는 것은 기본적인 편집 / 단계 / 커밋 패턴을 중심으로 이루어집니다. 먼저 작업 디렉토리에서 파일을 편집합니다. 프로젝트의 현재 상태 사본을 저장할 준비가되면 git add 명령으로 변경 사항을 저장합니다. 준비된 스냅 샷에 만족하면 git commit을 사용하여 프로젝트 히스토리에 커밋합니다. git reset 명령은 커밋 또는 준비된 스냅 샷을 실행 취소하는 데 사용됩니다.

git add 및 git commit 외에도 세 번째 명령 인 git push는 완전한 협업 작업 흐름을 위해 필수적입니다. git push는 협업을 위해 커밋 된 변경 사항을 원격 저장소에 보내는 데 사용됩니다. 이렇게하면 다른 팀 구성원이 저장된 변경 사항 집합에 액세스 할 수 있습니다.



git add 명령을 저장소에 파일을 추가하는 svn add와 혼동해서는 안됩니다. 대신, git add는 더 추상적 인 변경 레벨에서 작동합니다. 즉, 파일을 변경할 때마다 git add를 호출해야하지만 svn add는 각 파일에 대해 한 번만 호출하면됩니다. 중복되는 것처럼 들릴 수도 있지만,이 워크 플로우는 프로젝트를 체계적으로 정리하는 것을 훨씬 쉽게 만듭니다.

The staging area

git add 명령의 주요 기능은 작업 디렉토리에서 보류중인 변경 사항을 git staging 영역으로 승격시키는 것입니다. 준비 영역은 Git의 고유 한 기능 중 하나이며, SVN (또는 심지어 Mercurial) 배경에서 오는 경우에는 머리를 감싸는 데 약간의 시간이 걸릴 수 있습니다. 작업 디렉토리와 프로젝트 기록 사이의 버퍼로 생각하는 것이 도움이 됩니다. 준비 영역은 Git의 "세 나무"중 하나, 작업 디렉토리 및 커밋 기록 중 하나로 간주됩니다.

마지막 커밋 이후에 변경 한 사항을 모두 커밋하는 대신 스테이지에서 관련 변경 사항을 포커스가있는 스냅 샷으로 그룹화하여 실제로 프로젝트 기록에 커밋 할 수 있습니다. 즉, 무관 한 파일에 대한 모든 종류의 편집 작업을 수행 한 다음 관련 변경 사항을 스테이지에 추가하고이를 논리적 커밋으로 분리하여 개별적으로 커밋 할 수 있습니다. 모든 개정 관리 시스템에서와 마찬가지로 원자 커밋을 만들어 버그를 추적하고 변경 사항을 프로젝트의 나머지 부분에 미치는 영향을 최소화하면서 쉽게 되돌릴 수 있도록하는 것이 중요합니다.

공통 옵션

```
git add <file>
```

다음 커밋을 위해 <file>의 모든 변경 사항을 준비하십시오.

```
git add <directory>
```

다음 커밋을 위해 <directory>의 모든 변경 사항을 스테이지하십시오.

```
git add -p
```

다음 커밋에 추가 할 파일의 부분을 선택할 수있는 대화 형 스테이징 세션을 시작하십시오. 이렇게하면 변경 사항이 표시되고 명령을 묻는 메시지가 나타납니다. 청크를 스테이지하려면 y를, 청크를 무시하려면, 작은 청크로 분할하고, 청크를 수동으로 편집하고, q를 종료하십시오.

Examples

새 프로젝트를 시작할 때 git add는 svn import와 동일한 기능을 제공합니다. 현재 디렉토리의 초기 확약을 작성하려면 다음 두 명령을 사용하십시오.

```
git add .
git commit
```

프로젝트를 시작하고 실행하면 git add에 경로를 전달하여 새 파일을 추가 할 수 있습니다.

```
git add hello.py
git commit
```

위의 명령을 사용하여 기존 파일의 변경 사항을 기록 할 수도 있습니다. 다시 한번 말하지만, Git은 새로운 파일의 준비 변경과 이미 저장소에 추가 된 파일의 변경을 구분하지 않습니다.