알짜배Git

Pragmatic Git Seminar

Apple - Young Seok Tony Kim

Version Control

보고서.최종본.docx

보고서.초안.docx

보고서.15.06.30.zip

보고서.최종본(이게진짜).docx

보고서.Final(진짜진짜진짜).pdf

Git?

- Distributed Version Control System
- 분산 버전 관리 시스탬
- SVN과 함께 가장 많이 쓰임
- Linus Torvalds가 만듬
- 빠름
- 그냥 쓰면됨

Git commands

- config
- help
- init
- clone
- add
- status
- diff
- commit
- reset
- rm
- mv
- branch
- checkout
- merge
- mergetool
- log
- stash
- tag
- fetch
- pull
- push
- rebase
- revert
- …등등 많음

config

```
$ git config --global user.name "YOUR NAME"

$ git config --global user.email "YOUR EMAIL"
```

init

```
$ git init
```

// 주로 로컬에서 git repository를 새로 처음부터 만들때 씁니다.

clone

```
$ git clone https://github.com/sparcs-kaist/araplus.git

// 주로 github이나 다른 서버에서 repository를 복사해올때 씀
```

add

```
$ git add foo.txt
```

// foo.txt의 변경사항들을 stage에 올릴때 쓰임. (아직 commit 된 것이 아님)

status

```
$ git status
```

// 현재 파일들에 변화가 있는지, 변화들이 commit이 잘 되었는지, 변화된 파일이 있지만 add가 안되었는지 등등을 확인할 때 쓰임.

commit

```
$ git commit
```

```
$ git commit -m “어쩌구변경사항이 있음”
```

```
// 커밋!
```

branch

```
$ git branch

// 어떠한 branch에 있는지 어떠한 브랜치가 있는지 list해줌.

// Default는 master branch

$ git branch helloworld

// helloworld라는 branch를 만들어줌
```

checkout

```
$ git checkout hotfix
// hotfix라는 branch로 갈아탐.
```

merge

- git merge

log

```
$ git log
```

```
$ git log --all --oneline --decorate –graph
```

```
// .gitconfig에 alias로 등록해두면 좋음! (꿀팁)
```

fetch

```
$ git fetch

// remote 저장소에 추가된 object들을 받아와서 local에 저장
```

pull

```
$ git pull

// remote 저장소에 추가된 object들을 local로 받아오면
서 merge까지 수행
```

push

```
$ git push

// remote 저장소에 작업한 내용을 보냄.
```