

Web

Table of contents

Setting	3
PhpStorm	3
CodeIgniter	3
Git	4
branch	5
Github	10
tag	11
SocialLogin	12
kakao.html	14
google.html	15
facebook.html	16
login.php	17
google.php	18
reCAPTCHA	19
v2_checkbox	19
v2_invisible	21
v3	22
map	23
Daum & Kakao	23
Google	24
email	25

Setting

Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

PhpStorm

- PhpStorm education 계정 등록
PhpStorm 홈페이지에서
- Bitnami root directory 변경
Bitnami 폴더 apache2/conf/bitnami
bitnami.conf 파일에서 Document root, Directory 경로 변경
- PhpStorm php파일에서 html 인식
Setting -> editor -> file types
Php 선택하고 *.html 등 추가
- PhpStorm run으로 실행하기
단축키 Alt + F2
Settings -> Languages & Frame~ -> PHP 에서 CLI Interpreter
Php 설치하고 php.exe 등록
이때, php 다운받은 폴더에서 php.ini-development 를 php.ini로 변경

Created with the Personal Edition of HelpNDoc: [Free EPub producer](#)

CodeIgniter

1. <https://codeigniter.com/> 최신 버전 다운로드
2. 압축 풀고 서버에 폴더 올리기
3. application/controller 폴더에 원하는 php 파일 생성
code 추가 *php 파일 이름 맨 처음은 반드시 대문자

```
<?php
defined('BASEPATH') OR exit('No direct script access allowed');
class /*php 파일 이름*/ extends CI_Controller {
    public function index()
    {
        $this->load->view('index');
    }
}
?>
```

application/views/index.php 파일 생성 html 작성

4. Apache 설정 변경
conf 폴더 httpd.conf 파일
#LoadModule rewrite_module modules/mod_rewrite.so 주석 제거(# 제거)
AllowOverride None 을 AllowOverride All 로 변경

5. 서버의 codeigniter 폴더의 application/config 폴더 config.php 파일
`$config['base_url'] = 'php 파일 이름.php';`
`$config['index_page'] = ' '; // 빈 값으로 설정`

6. 서버의 codeigniter 폴더의 index.php 가 존재하는 폴더에 .htaccess 파일 추가

```
<IfModule mod_rewrite.c>
    RewriteEngine On
    RewriteCond %{REQUEST_FILENAME} !-f
    RewriteCond %{REQUEST_FILENAME} !-d
    RewriteRule ^(.*)$ index.php/$1 [L]
</IfModule>
```

7. 서버의 codeigniter 폴더의 application/config 폴더 routes.php
`$route['default_controller'] = 'php 파일 이름';`

=> localhost/codeigniter 폴더 이름

Created with the Personal Edition of HelpNDoc: [Generate Kindle eBooks with ease](#)

Git

<https://git-scm.com/>

Download for Windows // Git 설치

Git Bash 열기

git 이라고 쳤을 때 이렇게 나오면 정상적으로 잘 설치 된 것

```
$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
      [--exec-path<=path>] [--html-path] [--man-path] [--info-path]
      [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
      [--git-dir<=path>] [--work-tree<=path>] [--namespace<=name>]
      <command> [<args>]

These are common Git commands used in various situations:


start a working area (see also: git help tutorial)
  clone      Clone a repository into a new directory
  init       Create an empty Git repository or reinitialize an existing one


work on the current change (see also: git help everyday)
  add        Add file contents to the index
  mv         Move or rename a file, a directory, or a symlink
  reset      Reset current HEAD to the specified state
  rm         Remove files from the working tree and from the index


examine the history and state (see also: git help revisions)
  bisect     Use binary search to find the commit that introduced a bug
  grep       Print lines matching a pattern
  log        Show commit logs
  show       Show various types of objects
  status     Show the working tree status


grow, mark and tweak your common history
  branch     List, create, or delete branches
  checkout   Switch branches or restore working tree files
  commit     Record changes to the repository
  diff       Show changes between commits, commit and working tree, etc
  merge      Join two or more development histories together
  rebase     Reapply commits on top of another base tip
  tag        Create, list, delete or verify a tag object signed with GPG


collaborate (see also: git help workflows)
  fetch      Download objects and refs from another repository
  pull       Fetch from and integrate with another repository or a local branch
  push       Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
```

자주 쓰는 명령어

`pwd` 현재 자신의 위치 확인

`cd` 이동 할 위치

`mkdir` 새로운 폴더 만들기

`git init` 현재 디렉토리를 git의 (버전) 저장소로 만들

`ls -al` 현재 디렉토리의 파일 목록

`vim` 파일 생성

`i` 삽입

`esc` 삽입 끄기

`:wq` 저장하고 끄기

```

git add git이 파일을 추적하도록 명령
git status 프로젝트 폴더의 상태 확인
cat 파일의 내용 확인
git commit commit 하기
*git은 commit 전에 add를 반드시 해야
소스 코드를 수정하다 보면 commit을 놓칠 때가 많음
많은 작업을 담은 거대한 버전 하나를 만들 때 git은 add라는 과정을 통해
commit 하고자하는 파일만 commit 할 수 있음
git log commit 한 기록
stage area commit 대기 하는 파일들
stage - repository commit이 된 결과 저장소
git log -p commit 과 commit 사이의 소스상의 차이
git diff 버전id..버전id 소스 코드 사이의 차이점 확인
git diff add 하기 전과 add 한 후의 내용 비교
*작업한 내용을 마지막으로 검토 할 수 있는 기회
git reset 버전id --hard 버전id로 돌아감
git commit -a 수정하거나 삭제된 파일을 자동으로 add
git commit --help 메뉴얼
git commit -am "코멘트" git commit 할 때의 설명을 코멘트 안에 바로 적어서 올릴 수 있음

```

* 한 번만 하면 됨

```

git config --global user.name
git config --global user.name 이름
git config --global user.email 이메일

```

Created with the Personal Edition of HelpNDoc: [Full-featured Help generator](#)

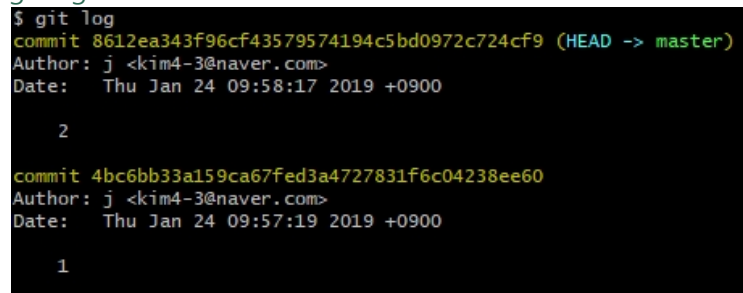
branch

branch 만들기

```

vim 파일 이름 에서 내용 추가 ex) a
git commit -am "1"
vim ↑파일 이름 에서 내용 추가 ex) b
git commit -am "2"
git log

```



```

$ git log
commit 8612ea343f96cf43579574194c5bd0972c724cf9 (HEAD -> master)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 09:58:17 2019 +0900

    2

commit 4bc6bb33a159ca67fed3a4727831f6c04238ee60
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 09:57:19 2019 +0900

    1

```

git branch

```

$ git branch
* master

```

⇒ master

```

git branch 새로운 브랜치 이름 브랜치 생성
git checkout 새로만든 브랜치 이름 브랜치 전환

```

```

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (master)
$ git branch exp

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (master)
$ git branch
  exp
* master

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (master)
$ git checkout exp
Switched to branch 'exp'

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (exp)
$ git branch
* exp
  master

```

⇒ 새로 만든 exp로 전환된 것을 확인할 수 있음

vim ↑파일 이름 에서 내용 추가 ex) c

git commit -am "3"

git log

```

$ git log
commit 209dd872b6793db86176206cbaa75d1caa9f9f26 (HEAD -> exp)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 10:03:39 2019 +0900

    3

commit 8612ea343f96cf43579574194c5bd0972c724cf9 (master)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 09:58:17 2019 +0900

    2

commit 4bc6bb33a159ca67fed3a4727831f6c04238ee60
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 09:57:19 2019 +0900

    1

```

git checkout master master로 브랜치 전환

git log

```

$ git log
commit 8612ea343f96cf43579574194c5bd0972c724cf9 (HEAD -> master)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 09:58:17 2019 +0900

    2

commit 4bc6bb33a159ca67fed3a4727831f6c04238ee60
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 09:57:19 2019 +0900

    1

```

⇒ exp에 추가된 commit 3 은 나오지 않음

cat ↑파일 이름 파일 내용

```

$ cat f1.txt
a
b

```

git checkout 새로만든 브랜치 이름 새로 만든 브랜치로 전환

cat ↑파일 이름 파일 내용

```

$ cat f1.txt
a
b
c

```

branch 정보 확인

git log --branches --graph --decorate --oneline 로그에 모든 브랜치를 표시하고, 그래프로 표현하고, 브랜치 명을 표시하고, 한줄로 표시할 때

```
$ git log --branches --decorate --graph --oneline
* f6c5447 (HEAD -> master) 5
| * c9f63e2 (exp) 4
| * 209dd87 3
|/
* 8612ea3 2
* 4bc6bb3 1
```

⇒ 1, 2 는 공통, 3, 4 는 exp, 5는 master

git log 비교할 브랜치 명1..비교할 브랜치 명2 1브랜치에는 없고 2브랜치에만 있는 것

```
$ git log master..exp
commit c9f63e2685b833fa9986843c0ecf5ae61a9ae88b (exp)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 10:12:42 2019 +0900

    4

commit 209dd872b6793db86176206cbaa75d1caa9f9f26
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 10:03:39 2019 +0900

    3

YouthSoTheCode@DESKTOP-INTG5HD MINGW64 ~/PhpstormProjects/git (master)
$ git log exp..master
commit f6c5447b78d1c75595d2408add2024539495ffbd (HEAD -> master)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 10:47:06 2019 +0900

    5
```

git log -p 비교할 브랜치 명1..비교할 브랜치 명2 소스코드도 보고싶을 때

```
$ git log -p exp..master
commit f6c5447b78d1c75595d2408add2024539495ffbd (HEAD -> master)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 10:47:06 2019 +0900

    5

diff --git a/f3.txt b/f3.txt
new file mode 100644
index 0000000..7898192
--- /dev/null
+++ b/f3.txt
@@ -0,0 +1 @@
+a
```

git diff 비교할 브랜치 명1..비교할 브랜치 명2 브랜치 간의 코드 비교

```
$ git diff master..exp
diff --git a/f1.txt b/f1.txt
index 422c2b7..708caeb 100644
--- a/f1.txt
+++ b/f1.txt
@@ -1,2 +1,4 @@
    a
    b
+ c
+
diff --git a/f2.txt b/f2.txt
index 6747a1f..f5eea67 100644
--- a/f2.txt
+++ b/f2.txt
@@ -1 +1 @@
-source : 2
+a
diff --git a/f3.txt b/f3.txt
deleted file mode 100644
index 7898192..0000000
--- a/f3.txt
+++ /dev/null
@@ -1 +0,0 @@
-a
```

branch 병합

A branch로 B 브랜치를 병합할 때

git checkout A

git merge B

```
$ git merge exp
Merge made by the 'recursive' strategy.
 f1.txt | 2 ++
 f2.txt | 2 +-
 2 files changed, 3 insertions(+), 1 deletion(-)
```

```
$ git log --branches --graph --decorate --oneline
* a847388 (HEAD -> master) Merge branch 'exp'
| \
| * c9f63e2 (exp) 4
| * 209dd87 3
* | f6c5447 5
|/
* 8612ea3 2
* 4bc6bb3 1
```

```
$ git checkout exp
Switched to branch 'exp'

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (exp)
$ git merge master
Updating c9f63e2..a847388
Fast-forward
 f3.txt | 1 +
 1 file changed, 1 insertion(+)
 create mode 100644 f3.txt

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (exp)
$ git log --branches --graph --decorate --oneline
* a847388 (HEAD -> exp, master) Merge branch 'exp'
| \
| * c9f63e2 4
| * 209dd87 3
* | f6c5447 5
|/
* 8612ea3 2
* 4bc6bb3 1
```

git branch -d 삭제할 브랜치 이름

```
$ git branch -d exp
Deleted branch exp (was a847388).

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (master)
$ git log --branches --graph --decorate --oneline
* a847388 (HEAD -> master) Merge branch 'exp'
| \
| * c9f63e2 4
| * 209dd87 3
* | f6c5447 5
|/
* 8612ea3 2
* 4bc6bb3 1
```

⇒ exp가 삭제되고 master로 병합

branch 병합 시 충돌 해결

예를 들어, master에 common.txt 파일이

```
function b(){
}
```

```
function a(){
}
```

exp에 commont.txt 파일이

```
function a(){
}
```

```
function c(){
```



```

}
로 작성되어 있을 때,
git checkout master
git merge exp
cat common.txt

```

```

$ cat common.txt
function b(){
}
function a(){
}
function c(){
}

```

하지만, master에 common.txt 파일이

```

function b(){
}
function a(master){
}

```

exp에 common.txt 파일이

```

function a(exp){
}
function c(){
}

```

로 작성되어 있을 때,

```

git checkout master

```

```

git merge exp

```

충돌이 일어납니다.

```

$ git merge exp
Auto-merging common.txt
CONFLICT (content): Merge conflict in common.txt
Automatic merge failed; fix conflicts and then commit the result.

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (master|MERGING)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 15 commits.
  (use "git push" to publish your local commits)

You have unmerged paths.
  (fix conflicts and run "git commit")
  (use "git merge --abort" to abort the merge)

Unmerged paths:
  (use "git add <file>..." to mark resolution)

        both modified:   common.txt

```

이때, vim common.txt 를 해서

내용을 원하는 대로 수정 후

git add common.txt 하면 merge 할 수 있어요

```

$ git add common.txt

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git (master|MERGING)
$ git status
On branch master
Your branch is ahead of 'origin/master' by 15 commits.
  (use "git push" to publish your local commits)

All conflicts fixed but you are still merging.
  (use "git commit" to conclude merge)

Changes to be committed:
  modified:   common.txt

```

stash

파일을 commit 하기 전 브랜치를 전환하고 싶을 때

```
$ vim f1.txt

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git/git2 (exp)
$ git checkout master
Switched to branch 'master'
M       f1.txt

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git/git2 (master)
$ 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:   f1.txt
```

⇒ modified : f1.txt

git stash

```
$ git stash save
warning: LF will be replaced by CRLF in f1.txt.
The file will have its original line endings in your working directory
Saved working directory and index state WIP on exp: ebe3b1a 1

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git/git2 (exp)
$ git status
On branch exp
nothing to commit, working tree clean
```

git stash list stash 한 리스트
 git stash apply stash 한 파일 복원(리스트는 삭제되지 않음)
 git stash drop stash 리스트 삭제
 git stash pop apply + drop

Created with the Personal Edition of HelpNDoc: [News and information about help authoring tools and software](#)

Github

<https://github.com/>

원격 저장소 만들기
 회원가입 후 <https://github.com/new>

...or push an existing repository from the command line

```
git remote add origin https://github.com/kim43/git.git
git push -u origin master
```

저장소를 만든 후 첫번째 줄을 복사하여 GitBash에 붙여넣기

git push -u origin master 원격 저장소에 연결되어 push 됨
 연결 시켜놓았기 때문에 그 이후에 push 할 때는 git push 만.

원격 저장소의 내용을 지역 저장소로 복제하는 방법

Clone or download 버튼 누르고 주소 복사
 git clone 복사 된 주소 .

하나의 원격 저장소를 매개로 해서 두개의 지역 저장소의 작업을 동기화 하는 방법
 clone 후 git push / git pull

ssh를 이용해서 로그인 없이 원격 저장소 사용하기

Clone or download 에서 use SSH 복사
 GitBash
 ssh-keygen (enter 4번) *경로 잘 보기

```

$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/YouthSoTheCode/.ssh/id_rsa):
Created directory '/c/Users/YouthSoTheCode/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/YouthSoTheCode/.ssh/id_rsa.
Your public key has been saved in /c/Users/YouthSoTheCode/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:h97qKbocFJpZoaRyhIpojdXqVcK98dzUEzr9rZisp9M YouthSoTheCode@DESKTOP-IMTG5HD
D
The key's randomart image is:
+---[RSA 2048]---+
| .o + .      ... |
| .+ o = +    .oo |
| * * + o = oo .. |
| =+ B o . + .. . |
| . = o   S .   .. |
|   o   . o . o . |
|   .   . . . + . |
|   . . . o . E   |
|   +o o+ o+     |
+---[SHA256]---+

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~
$ cd ~/.ssh

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/.ssh
$ ls -al
total 17
drwxr-xr-x 1 YouthSoTheCode 197121  0 1월 23 16:47 ./
drwxr-xr-x 1 YouthSoTheCode 197121  0 1월 23 16:47 ../
-rw-r--r-- 1 YouthSoTheCode 197121 1843 1월 23 16:47 id_rsa
-rw-r--r-- 1 YouthSoTheCode 197121  412 1월 23 16:47 id_rsa.pub

```

id_rsa - private key

id_rsa.pub - public key

cat id_rsa.pub

나오는 코드를 복사하여 github 홈페이지에서 Settings

SSH and GPG keys

new SSH keys 등록

GitBash

git clone 복사한 ssh

Created with the Personal Edition of HelpNDoc: [What is a Help Authoring tool?](#)

tag

git tag 태그 이름 (태그가 가르킬 버전의 커밋 아이디) 태그 생성(light weight tag)

```

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git/tag (master)
$ git tag 1.0.0 master

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git/tag (master)
$ git tag
1.0.0

YouthSoTheCode@DESKTOP-IMTG5HD MINGW64 ~/PhpstormProjects/git/tag (master)
$ git log --decorate
commit 21faa5313a347458585bc3ce097820f2ae39f6ae (HEAD -> master, tag: 1.0.0)
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 16:41:54 2019 +0900

    2

commit 54cd0bacecebd415e86aca6d05d8e90bd62d17ce
Author: j <kim4-3@naver.com>
Date: Thu Jan 24 16:41:31 2019 +0900

    1

```

git tag 태그 목록 보기

git tag -d 삭제할 태그명 태그 삭제

```

YouthSoTheCode@DESKTOP-IMTG5HD MINGW
$ git tag 1.1.1

YouthSoTheCode@DESKTOP-IMTG5HD MINGW
$ git tag
1.0.0
1.1.0
1.1.1

YouthSoTheCode@DESKTOP-IMTG5HD MINGW
$ git tag -d 1.1.1
Deleted tag '1.1.1' (was fef23cd)

```

`git tag -a` 태그 이름 -m 태그에 대한 설명 (태그가 가르킬 버전의 커밋 아이디) 태그 생성(annotated tag)

```

$ git tag -a 1.1.0 -m "bug fix"

$ git tag -v 1.1.0
object 1cfa27a11360fd1a1f51ae6e34ca817e12911852
type commit
tag 1.1.0
tagger j <kim4-3@naver.com> 1548316019 +0900

bug fix
error: no signature found

```

`git push --tags` 태그도 원격 저장소로 업로드

Created with the Personal Edition of HelpNDoc: [Produce electronic books easily](#)

SocialLogin

· 카카오 로그인 * javascript key

<https://developers.kakao.com/>

앱 개발 시작하기 → 앱 만들기 (key 발급)

→ 설정(일반) → 플랫폼 추가 (웹) / 설정(사용자관리) → 접근 권한 관리항목 설정

· 구글 로그인 * content ID

<https://developers.google.com/>

Products → Google+ Platform for Web → GET STARTED → Google API Console

→ 라이브러리 만들기 → 사용자 인증 정보 만들기

→ OAuth 클라이언트 ID (OAuth 동의 화면 에서 제품 이름과 사진 저장)

→ 웹 애플리케이션 생성

{ Google+ API 사용설정

· 페이스북 로그인 * app ID

<https://developers.facebook.com/>

내 앱 새로 만들기 → 제품 추가 (Facebook 로그인) → Facebook 로그인(설정)

→ 유효한 OAuth 리디렉션 URI 추가

카카오 로그인

<개인 정보>

프로필 정보(이름, 아이디, 프로필 사진, 썸네일 사진) : 연결 시 필수

* 연결 시 선택 | 이용 중 사용 | 사용 안함

카카오 서비스 내 친구목록

카카오 계정 (이메일) [소유여부, 값, 인증여부, 유효여부]

성별 [소유여부, 값]

연령대 [소유여부, 값]

생일 [소유여부, 값]

구글 로그인

<개인 정보>

Google+ API 라이브러리

아이디

이름(full, given, family)

이메일

프로필 사진

Google People API 라이브러리

상세 주소

이메일 주소

전화번호

생년월일

페이스북 로그인

<개인 정보>

이름

프로필 사진

이메일

추가 가능한 정보 (앱 검수 → 권한 및 기능)

연령대

생일

성별

출신지

현재 거주지

사용자의 게시물, 좋아요, 프로필 url, 태그된 장소 등

Created with the Personal Edition of HelpNDoc: [Easily create iPhone documentation](#)

kakao.html

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8"/>
  <meta http-equiv="X-UA-Compatible" content="IE=edge"/>
  <meta name="viewport" content="user-scalable=no, initial-scale=1.0, maximum-scale=1.0,
minimum-scale=1.0, width=device-width"/>
  <title>카카오톡</title>
  <script src="//developers.kakao.com/sdk/js/kakao.min.js"></script>
  <script src="//code.jquery.com/jquery.min.js"></script>
</head>
<body>
<a id="kakao-login-btn"></a>
<a href="http://developers.kakao.com/logout"></a>
<script type='text/javascript'>
  //<![CDATA[
  Kakao.init('76afc7d22ca69f2e0527c1300528c2a6'); // 사용할 앱의 JavaScript 키를 설정해 주세요.
  // 카카오 로그인 버튼을 생성합니다.
  function loginWithKakao() {
    Kakao.Auth.createLoginButton({
      container: '#kakao-login-btn',
      //lang : 'en',
      //size : 'large',
      //persistAccessToken : false,
      success: function(authObj) {
        alert(JSON.stringify(authObj));
        getKakaotalkUserProfile();
        //Kakao.Auth.cleanup(); // 카카오 로그인 버튼 삭제
      },
      fail: function(err) {
        alert(JSON.stringify(err));
      }
    });
  }

  function getKakaotalkUserProfile() {
    Kakao.API.request({
      url:'/v2/user/me',
      success:function (res) {
        $.ajax({
          type:'POST',
          url:'login.php', // 정보 받을 url
          data: { // 정보
            //"accesstoken":Kakao.Auth.getAccessToken(),
            "user_Id":res.id,
            "user_Name":res.properties.nickname,
            "user_Email":res.kakao_account.email
          }
        });
      }
    });
  }
  ]]>
</script>
</body>
</html>
```

```

    },
    }).done(function (JsonData) {
        alert(JsonData);
    });
    },
    fail:function (error) {
        alert(JSON.stringify(error));
    }
    });
}

```

```
loginWithKakao();
```

```
//]]>
```

```

</script>
<div id="kakao-logged-group"></div>
<div id="kakao-profile"></div>
</body>
</html>

```

Created with the Personal Edition of HelpNDoc: [Full-featured EBook editor](#)

google.html

```

<html>
<head>
    <meta name="google-signin-scope" content="profile email">
    <meta name="google-signin-client_id" content="849486600358-
8mtu3ei6lka1226n0aatjtkm94bi4eh4.apps.googleusercontent.com"> <!--content = "client_id"-->
    <script src="https://apis.google.com/js/platform.js" async defer></script>
</head>
<body>
<div
    class="g-signin2"
    data-onsuccess="onSignIn"
    data-onfailure="onSignInFailure">
</div>
<script>
    function onSignIn(googleUser) {
        var profile = googleUser.getBasicProfile();
        console.log("ID: " + profile.getId()); // Don't send this directly to your server!
        console.log("Full Name: " + profile.getName());
        console.log("Email: " + profile.getEmail());

        // The ID token you need to pass to your backend:
        var id_token = googleUser.getAuthResponse().id_token;
        console.log("ID Token: " + id_token);

        //id_token 을 받아서 verifyIdToken 으로 분해
        var xhr = new XMLHttpRequest();
        xhr.open('POST', 'google_login.php', true);
        xhr.setRequestHeader('Content-Type', 'application/x-www-form-urlencoded');
        xhr.onload = function() {
            console.log("Signed in as: " + xhr.responseText);
        };
        xhr.send('id_token=' + googleUser.getAuthResponse().id_token);
    }

```

```

}
</script>
</body>
</html>

```

Created with the Personal Edition of HelpNDoc: [Easily create Web Help sites](#)

facebook.html

```

<!DOCTYPE html>
<html>
<head>
  <meta charset = "UTF-8">
  <script src="//code.jquery.com/jquery.min.js"></script>
</head>
<body>
<script>
  function statusChangeCallback(response) {
    console.log('statusChangeCallback');
    console.log(response);

    if (response.status === 'connected') {
      // Logged into your app and Facebook.
      testAPI();
    } else {
      // The person is not logged into your app or we are unable to tell.
      document.getElementById('status').innerHTML = 'Please log ' +
        'into this app.';
    }
  }

  function checkLoginState() {
    FB.getLoginStatus(function(response) {
      statusChangeCallback(response);
    });
  }

  window.fbAsyncInit = function() {
    FB.init({
      appId: '392287181507266', // 앱 ID
      cookie: true,
      xfbml: true,
      version: 'v3.2' // 현재 버전
    });

    FB.getLoginStatus(function (response) {
      statusChangeCallback(response);
    });
  };

  (function(d, s, id){
    var js, fjs = d.getElementsByTagName(s)[0];
    if (d.getElementById(id)) {return;}
    js = d.createElement(s); js.id = id;
    js.src = "https://connect.facebook.net/en_US/sdk.js";
    fjs.parentNode.insertBefore(js, fjs);
  })(document, 'script', 'facebook-jssdk');

```



```
function testAPI() {
    console.log('Welcome! Fetching your information.... ');
    FB.api('/me', {fields: 'email, name'}, function(response) {
        console.log('Successful login for: ' + response.name);
        console.log('id : ' + response.id);
        //console.log(JSON.stringify(response));

        $.ajax({
            type:'POST',
            url:'login.php',
            data: {
                "user_Id":response.id,
                "user_Name":response.name,
                "user_Email":response.email
            },
        }).done(function (JsonData) {
            alert(JsonData);
        });

        document.getElementById('status').innerHTML =
            'Thanks for logging in, ' + response.name + '!';
    });
}
```

```
</script>
<fb:login-button scope="public_profile,email" onlogin="checkLoginState();">
    로그인
</fb:login-button>

<div id="status">
</div>
</body>
</html>
```

login.php

```
<?php
$user_id = $_POST['user_Id'];
$user_name = $_POST['user_Name'];
$user_email = $_POST['user_Email'];

echo "$user_name 님이 로그인 하셨습니다.";

try{
    $pdo = new PDO("mysql:."host=localhost;".dbname=test", 'root', '123456',
array(PDO::MYSQL_ATTR_INIT_COMMAND=>"SET NAMES utf8"));
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    $duplicate = $pdo->query("SELECT id FROM login WHERE id='$user_id'");
    $count = $duplicate->rowCount();
```

```

    if($count == 1)
    {
    }
    else
    {
        $db = $pdo->prepare("INSERT INTO login (id, name, account_email) VALUES
(:user_id, :user_name, :user_email)");

        $db->bindParam(':user_id', $user_id);
        $db->bindParam(':user_name', $user_name);
        $db->bindParam(':user_email', $user_email);

        $db->execute();
        $pdo = null;
    }
}
catch(Exception $e){
    echo $e->getMessage(); //에러 출력
}

?>

```

Created with the Personal Edition of HelpNDoc: [Produce online help for Qt applications](#)

google.php

```
<?php
```

```
require_once 'vendor/autoload.php';
```

```
// Get $id_token via HTTPS POST.
```

```
$id_token = $_POST['id_token'];
```

```
$client = new Google_Client(); // Specify the CLIENT_ID of the app that accesses the backend
```

```
$client->setApplicationName("google_login");
```

```
$client->setDeveloperKey("849486600358-8mtu3ei6lka1226n0aatjtkm94bi4eh4.apps.googleusercontent.com");
```

```
$payload = $client->verifyIdToken($id_token);
```

```
if ($payload) {
```

```
    $user_id = $payload['sub'];
```

```
    $user_name = $payload['name'];
```

```
    $user_email = $payload['email'];
```

```
    try{
```

```
        $pdo = new PDO("mysql:host=localhost;dbname=test", 'root', '123456',
```

```
array(PDO::MYSQL_ATTR_INIT_COMMAND=>"SET NAMES utf8"));
```

```
        $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
```

```
        $duplicate = $pdo->query("SELECT id FROM login WHERE id='$user_id'");
```

```
        $count = $duplicate->rowCount();
```

```
        if($count == 1)
```

```
{
```

```

    }
    else
    {
        $db = $pdo->prepare("INSERT INTO login (id, name, account_email) VALUES
(:user_id, :user_name, :user_email)");

        $db->bindParam(':user_id', $user_id);
        $db->bindParam(':user_name', $user_name);
        $db->bindParam(':user_email', $user_email);

        $db->execute();
        $pdo = null;
    }
}
catch(Exception $e){
    echo $e->getMessage(); //에러 출력
}

// If request specified a G Suite domain:
//$domain = $payload['hd'];

echo "로그인 성공";
} else {
    // Invalid ID token
}

?>

```

Created with the Personal Edition of HelpNDoc: [Free CHM Help documentation generator](#)

reCAPTCHA

key 발급 방법

<https://developers.google.com/recaptcha/> → GET STARTED → sign up for and API key pair

Choose the type of reCAPTCHA 항목에서 원하는 버전 선택

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

v2_checkbox

reCAPTCHA_v2_Checkbox

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8"/>
    <script src="https://www.google.com/recaptcha/api.js" async defer></script>
</head>
<body>
<form action="~~.php" method = "post">

    <p>아이디&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input type = "text" name = "id" size = "20"></p>

```

```

<p>비밀번호        <input type = "password" name = "password" size =
"20"></p>

<div class="g-recaptcha" data-sitekey=""></div>
<br>
<input type="submit" value="Submit">

</form>
</body>
</html>

```

서버 측 실행 code

```
<?php
```

```

header("Content-Type: text/html; charset=UTF-8");
/*
$id = $_POST['id'];
$password = $_POST['password'];
*/
if(isset($_POST['g-recaptcha-response']))
{
    $captcha=$_POST['g-recaptcha-response'];
}
//$secretKey = ""; //v2_Checkbox
$ip = $_SERVER['REMOTE_ADDR'];
$response = file_get_contents("https://www.google.com/recaptcha/api/siteverify?
secret=".$secretKey."&response=".$captcha."&remoteip=".$ip);
$responseKeys = json_decode($response,true);

if(intval($responseKeys["success"]) !== 1)
{
    echo '로봇';
}
else
{
    echo '사람';
}
/*
    try{
        $pdo = new PDO("mysql:host=localhost;dbname=test",'root', '123456',
array(PDO::MYSQL_ATTR_INIT_COMMAND=>"SET NAMES utf8"));
        $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

        $db = $pdo->prepare("INSERT INTO test (id, password) VALUES (:id, :password)");

        $db->bindParam(':id', $id);
        $db->bindParam(':password', $password);

        $db->execute();
        $pdo = null;
    }
    catch(Exception $e){
        echo $e->getMessage(); //에러 출력
    }
*/

```

```
}
?>
```

Created with the Personal Edition of HelpNDoc: [Easy EPub and documentation editor](#)

v2_invisible

reCAPTCHA_v2_Invisible

```
<!DOCTYPE html>
<html>
<head>

  <script src='https://www.google.com/recaptcha/api.js'></script>
  <script>
    function onSubmit(token)
    { document.getElementById("test-form").submit(); }
  </script>
</head>

<body>
  <form id = "test-form" action = "~~.php" method = "post">

    <button
      class="g-recaptcha"
      data-sitekey=""
      data-callback='onSubmit'>
      Submit
    </button>

  </form>
</body>
</html>
```

서버 측 실행 code

```
<?php

header("Content-Type: text/html; charset=UTF-8");

if(isset($_POST['g-recaptcha-response']))
{
  $captcha=$_POST['g-recaptcha-response'];
}
$secretKey = ""; //v2_Invisible

$ip = $_SERVER['REMOTE_ADDR'];
$response = file_get_contents("https://www.google.com/recaptcha/api/siteverify?secret=".$secretKey."&response=".$captcha."&remoteip=".$ip);
$responseKeys = json_decode($response,true);

if(intval($responseKeys["success"]) !== 1)
{
  echo '로봇';
}
```

```

}
else
{
    echo '사람';
}
?>

```

Created with the Personal Edition of HelpNDoc: [iPhone web sites made easy](#)

v3

v3 (웹 사이트에서의 상호작용 기반으로 작동, 직접 인증 X)

```

<!DOCTYPE html>
<html>
<head>
    <script src='https://www.google.com/recaptcha/api.js?render=YOUR_SITE_KEY'></script>
    <script>
        window.onload = function() {
            document.getElementById('button-id').addEventListener("click",function () {
                grecaptcha.ready(function() {
                    grecaptcha.execute('6LdWZoYUAAAAALzVoHa1kygP4qYAVBMs8PDzPQUA',
{action: 'action_name'}).then(function(token) {
                        document.getElementById('form-id').innerHTML = '<input
type="hidden" name="token" value="' + token + '"><input type="hidden" name="action"
value="action_name">';
                        document.getElementById('form-id').submit();
                    });
                });
            });
        }
    </script>
</head>
<body>
<form action="~~.php" method="post" id="form-id">
</form>
<button id="button-id">Submit</button>
</body>
</html>

```

<button id="button-id">Submit</button> </body> </html>

v3 서버 측 실행 code

```

<?php
$recaptcha_url = 'https://www.google.com/recaptcha/api/siteverify';
$post_data["secret"] = '';
$post_data["response"] = $_POST['token'];

$ch = curl_init();
curl_setopt($ch, CURLOPT_URL, $recaptcha_url);
curl_setopt($ch, CURLOPT_POST, 1);
curl_setopt($ch, CURLOPT_POSTFIELDS, $post_data);
curl_setopt ($ch, CURLOPT_RETURNTRANSFER, 1);
$recaptcha = curl_exec ($ch);
$recaptcha = json_decode($recaptcha);

if($recaptcha->score>=0.5) {
    // 리캡차 성공시 수행할 동작
    echo "산랏";
} else {
    // 리캡차 실패시 수행할 동작
    echo "롤받";
}

```

```
}
?>
```

Created with the Personal Edition of HelpNDoc: [Write EPub books for the iPad](#)

map

Created with the Personal Edition of HelpNDoc: [Free Web Help generator](#)

Daum & Kakao

다음 & 카카오 지도

<http://apis.map.daum.net/> 참고

1. <https://developers.kakao.com/> 접속
2. 개발자 등록 및 앱 생성
3. 웹 플랫폼 추가 : 앱 선택 [설정] - [일반] - [플랫폼 추가] - 웹 선택 후 추가
4. 사이트 도메인 등록 : [웹] 플랫폼을 선택하고, [사이트 도메인] 등록 (ex: localhost)
5. 페이지 상단의 [JavaScript 키]를 지도 API의 appkey로 사용

[주소로 장소 표시하는 html code]

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8"/>
  <title>카카오 지도</title>
</head>
<body>
<div id="map" style="width:600px,height:350px;"></div>

<script type="text/javascript" src="//dapi.kakao.com/v2/maps/sdk.js?
appkey=&libraries=services"></script> <!--appkey = 발급받은 Javascript key-->
<script>

  var mapContainer = document.getElementById('map'), // 지도를 표시할 div
      mapOption = {
        center: new daum.maps.LatLng(35.813803, 127.087413), // 지도의 중심좌표
        level: 3 // 지도의 확대 레벨
      };

  // 지도를 생성합니다
  var map = new daum.maps.Map(mapContainer, mapOption);

  // 주소-좌표 변환 객체를 생성합니다
  var geocoder = new daum.maps.services.Geocoder();

  // 주소로 좌표를 검색합니다
  geocoder.addressSearch('전라북도 전주시 완산구 천잠로 303', function(result, status) {

    // 정상적으로 검색이 완료됐으면
    if (status === daum.maps.services.Status.OK) {

      var coords = new daum.maps.LatLng(result[0].y, result[0].x);
```

```

// 결과값으로 받은 위치를 마커로 표시합니다
var marker = new daum.maps.Marker({
    map: map,
    position: coords
});

// 인포윈도우로 장소에 대한 설명을 표시합니다
var infowindow = new daum.maps.InfoWindow({
    content: '<div style="width:150px;text-align:center;padding:3px 0;">소덕코드</div>'
});
infowindow.open(map, marker);

// 지도의 중심을 결과값으로 받은 위치로 이동시킵니다
map.setCenter(coords);
}
});
</script>
</body>
</html>

```

Created with the Personal Edition of HelpNDoc: [Free Qt Help documentation generator](#)

Google

apikey 발급

```

<!DOCTYPE html>
<html>
<head>
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no">
    <meta charset="utf-8">
    <title>구글 지도</title>
    <style>
        /* Set the size of the div element that contains the map */
        #map {
            height: 400px; /* The height is 400 pixels */
            width: 100%; /* The width is the width of the web page */
        }
    </style>
</head>
<!--The div element for the map -->
<div id="map"></div>

<script>
    // Initialize and add the map
    function initMap() {
        // The location of Uluru
        var uluru = {lat: 35.813803, lng: 127.087413};
        // The map, centered at Uluru
        var map = new google.maps.Map(
            document.getElementById('map'), {zoom: 17, center: uluru});
        // The marker, positioned at Uluru
        var marker = new google.maps.Marker({position: uluru, map: map});
    }
</script>

```



```
<script async defer
  src="https://maps.googleapis.com/maps/api/js?key=YOUR_API_KEY&callback=initMap">
</script>
</html>
```

Created with the Personal Edition of HelpNDoc: [Full-featured Kindle eBooks generator](#)

email

use phpmailer with out php composer

<https://github.com/PHPMailer/PHPMailer> : 최신 버전의 PHPMailer 다운받기

1. Naver

메일 환경 설정 → POP3/IMAP 설정 → IMAP/SMTP 설정 → 사용함 체크 & SMTP 서버명, 포트 번호 확인

2. Google

Gmail 설정 → 전달 및 POP/IMAP → IMAP 사용 체크 → (자세히 알아보기) 발신 메일 서버 내용 확인
Google 계정 → 보안 → 보안 수준이 낮은 앱의 액세스 → 보안 수준이 낮은 앱 허용 : 사용

서버 안에 꼭 있어야 할 파일

설치된 PHPMailer 폴더의 → PHPMailer.php / SMTP.php / Exception.php

[php code]

```
<?php
use PHPMailer\PHPMailer\PHPMailer;
use PHPMailer\PHPMailer\Exception;

require 'Exception.php';
require 'PHPMailer.php';
require 'SMTP.php';

$toAddress = ""; // 받는 이메일, 주소 무관
$mail = new PHPMailer();
$mail->CharSet = 'UTF-8';
$mail->IsSMTP();
$mail->SMTPAuth = true;
$mail->Host = "smtp.gmail.com";
$mail->Port = 587;
$mail->SMTPOptions = array(
    'ssl' => array(
        'verify_peer' => false,
        'verify_peer_name' => false,
        'allow_self_signed' => true
    )
);
$mail->IsHTML(true);
$mail->Username = "@gmail.com"; // your address
$mail->Password = ""; // password
$from_name = $_POST['name'];
$from_email = $_POST['email'];
$mail->FromName = $from_name;
$mail->From = $from_email;
```

```
$mail->addReplyTo("$from_email", "$from_name");
$mail->Subject = $_POST['subject'];
$mail->Body = $_POST['content'];
$mail->AddAddress($toAddress);
if (!$mail->Send()) {
    echo "Fail";
} else {
    $prevPage = $_SERVER['HTTP_REFERER'];
    header('location:'.$prevPage);
}

?>
```