

install guide

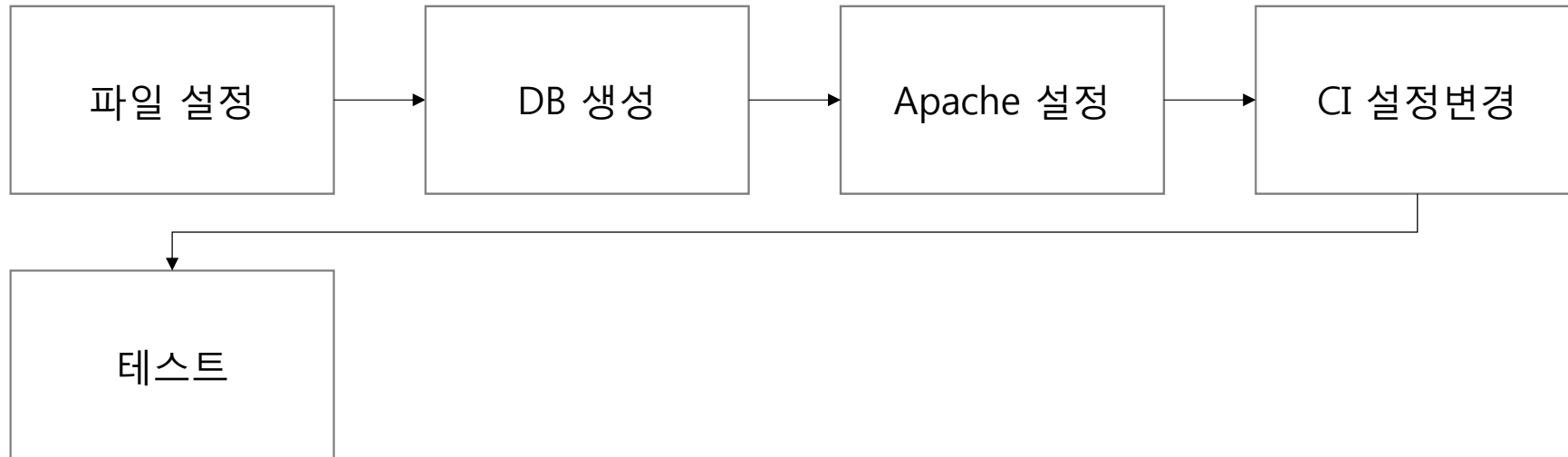
Written by naiyumie
ver 1.0.0

개 정 이 력

No	버전	작성자	작성일자	변경사유	변경내용	승인자
1	1.0	원상필	2015.03.02	최초작성		

0 워킹 프로세스

퍼블리싱 워킹 프로세스에 대해 설명 한다.



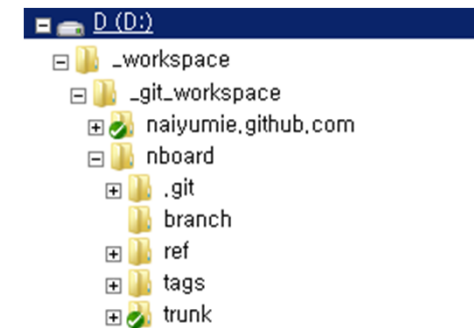
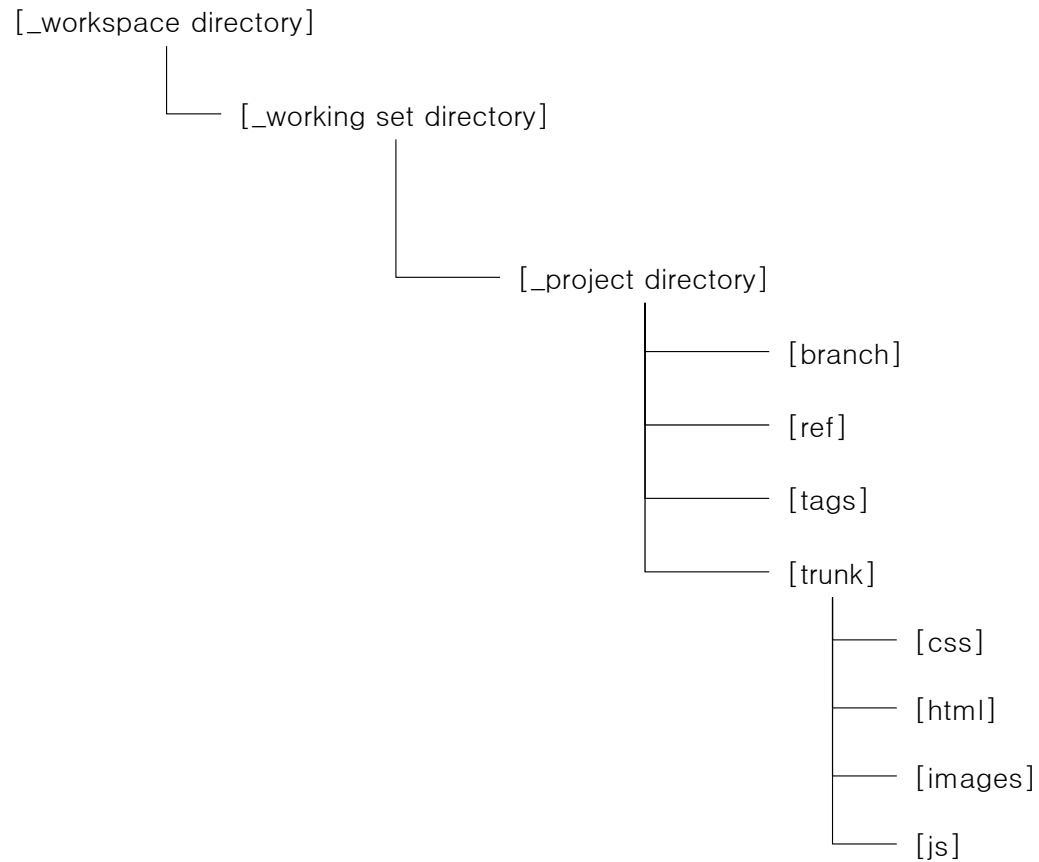
1. 인스톨할 파일을 얻는다.
2. DB를 생성한다.
3. Apache의 httpd.conf를 설정한다.
4. 파일을 브라우저에서 테스트한다.

이곳에 적혀있는 테이블명이나 비밀번호는 예시이므로 그대로 따라하면 안됩니다.
본인이 익숙해진뒤 테이블명, 비밀번호의 세팅이 필요합니다.

1. 파일설정

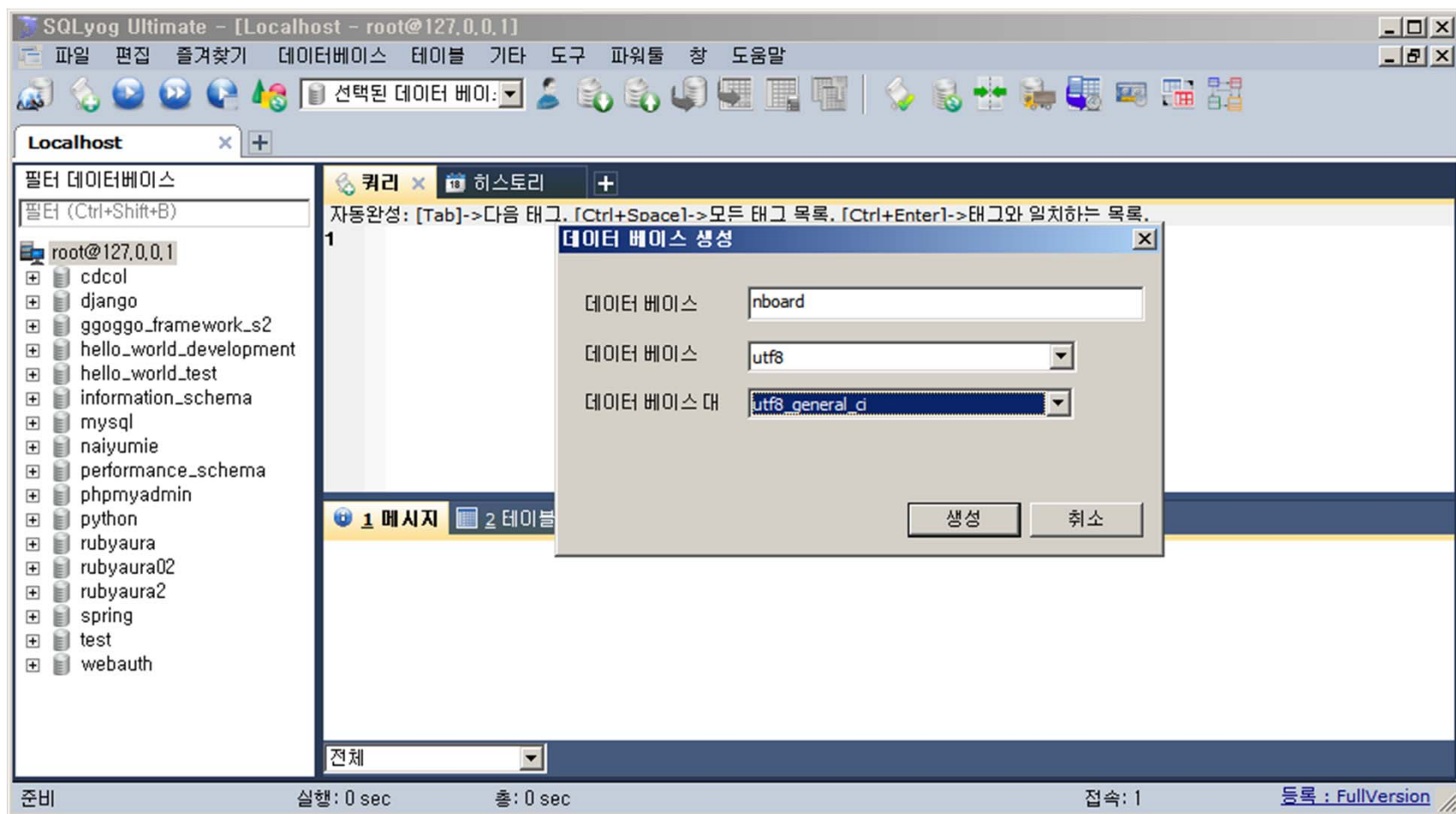
나이유미 게시판 파일을 github, 혹은 다운로드 방법으로 얻었다면 아래와 같은 형태로 배치한다.

이는 로컬호스트 기준이며 웹서버 혹은 웹호스팅을 쓰고 있다면 public_html, www와 같은 웹 루트 디렉터리에 trunk 안의 파일들을 배치한다.
즉 웹 루트 디렉터리에 trunk 안의 파일을 배치 하면된다.



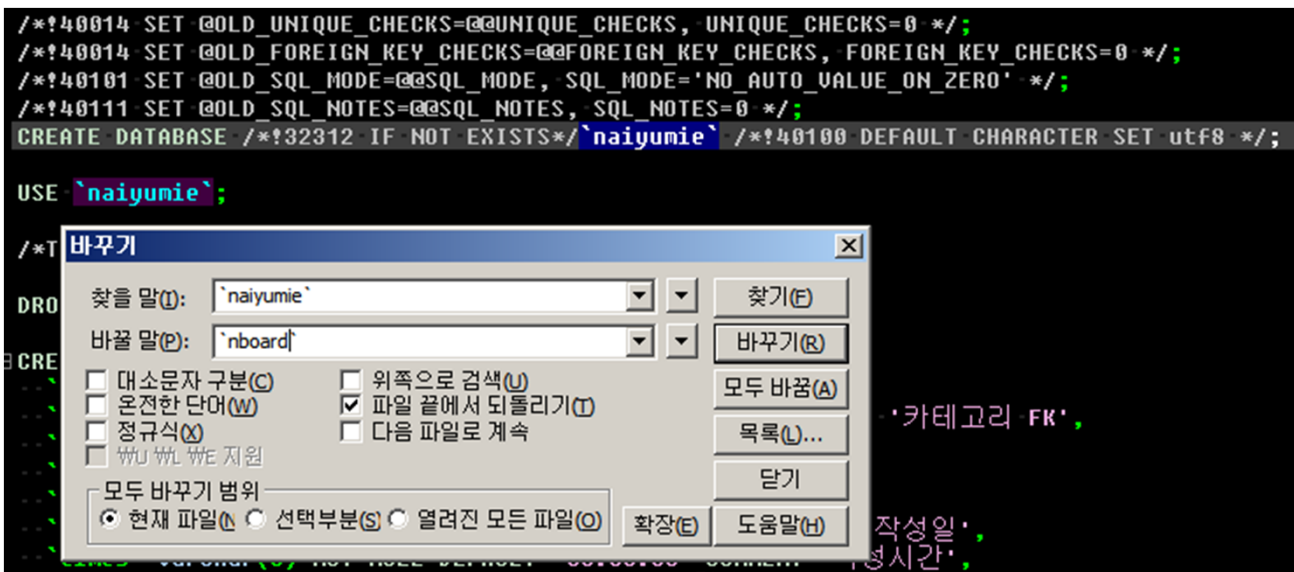
2. DB 생성

데이터 베이스를 생성한다. 만약 자신이 호스팅을 사용하고 있다면 데이터베이스 생성이 불가 할 수 있다.
자신의 데이터베이스가 nboard를 예시로 한다.



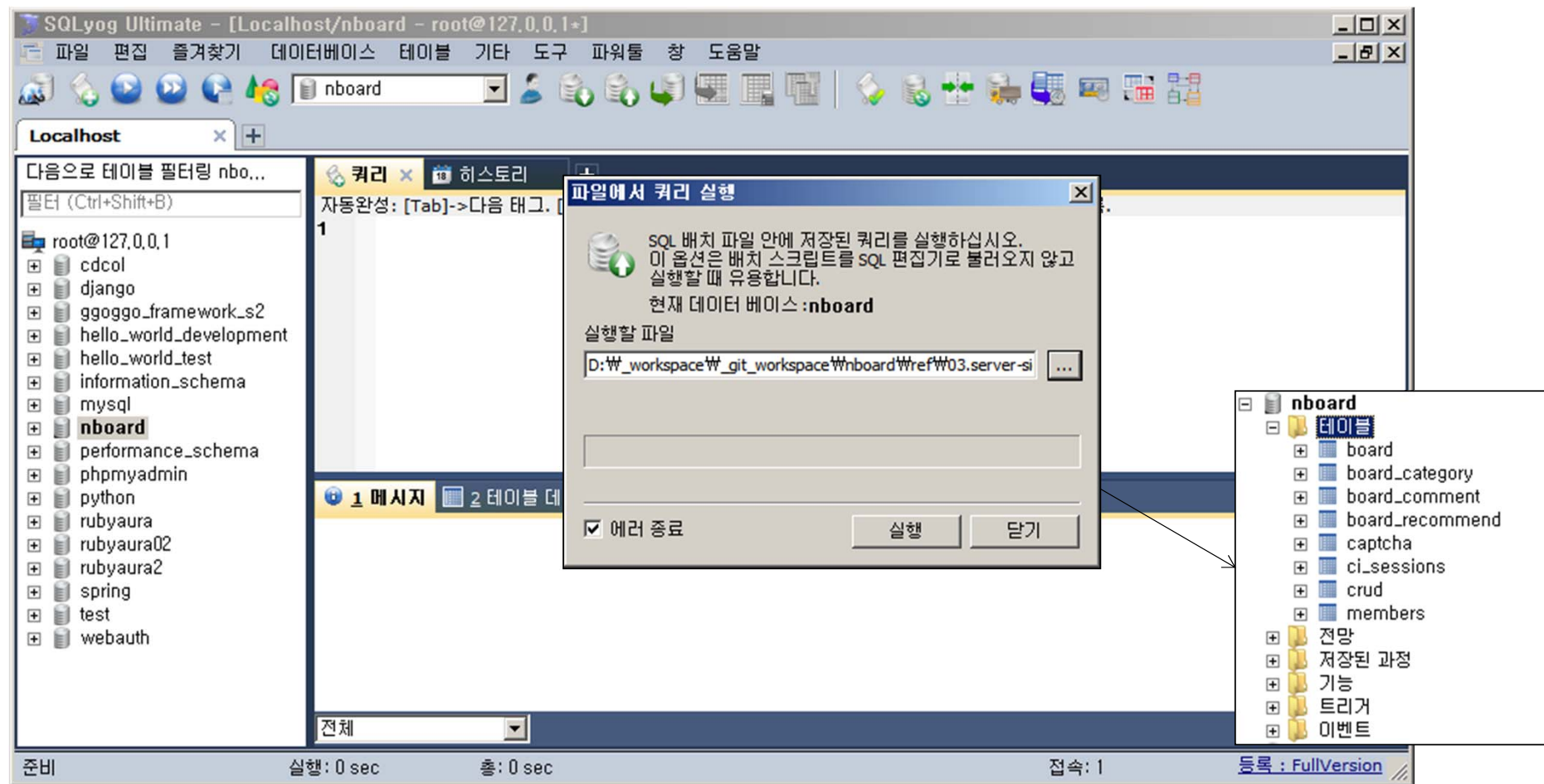
2.1 SQL 파일 수정

nboard/ref/03.server-side_sql/ 디렉터리에 sql 파일들이 있다. 여기서 nboard_naiyumie.sql을 에디터로 연다.
그리고 'naiyumie'를 'nboard'로 교체한다. (당연히 이곳은 자신의 데이터베이스 명이 되어야 하며 이미 nboard로 되어 있을수 있다)



2.2 SQL import

그리고 import 한다.



2.3 기본 회원 정보를 확인한다.

생성된 db에 members 테이블에 내용을 조회 하면 아래와 같은 회원을 볼 수 있다.
패스워드는 암호화 되어 알 수 없다.

user_id	user_pw	유형	부가설명
member1	12345	사용자	임시
admin	admin12345@	관리자	실제 관리자
tester	12345	관리자	방문객 테스트 관리자
guest	12345	사용자	방문객 테스트 사용자
naiyumie	12345	사용자	실제 사용자

uid	user_id	user_pw	nick
5	member1	xM0niBKBAXjwUmVZdMI2361SdI+FBBYf5PU0XC8UayBqH6FyUHQeT0ws/t50YMqB77k	멤버테스트
4	admin	U0XFmWHqSzqAgG76bLEZrtMhQPc/8iWyN1+cxcJ64S+HdfGgxhH962XuIYP6MWQrpKS	관리자
3	tester	bHE0tXod1ZPotyYurynqML1WFrTcni2d04BBaupL0It+bviXYfcXBqfsthJ8/yFAZm/	테스트관리자
2	guest	qMpCxz5GmqY4Q2C2Tmt/yTBi6YJnSuG7e107XXGvM2eHW8B4XN0zPQoDaTa7GLFDHqC	방문객
1	naiyumie	tuav7QUwBRJYUod0hZUGE1o9/60ECtR6p9t6H+tgSjKfuJ/B0W0K1+PNTLrb2e8T5h6	나이유미

2.4 CI용 계정을 생성한다.#1

접속을 root로 하게된다면 보안에 문제가 있으므로 CI용 계정을 생성한다. 아래와 같이 전체적 권한이 USAGE로 되어있어야 안전하다. 호스팅을 사용할 경우 같은 세팅할 필요 없다.

	사용자	호스트	암호	전체적 권한	Grant	실행
<input type="checkbox"/>	nboard	%	예	USAGE	아니오	 권한 수정  내보내기
<input type="checkbox"/>	nboard	localhost	예	USAGE	아니오	 권한 수정  내보내기

2.5 CI용 계정을 생성한다.#2

테이블 권한만 주는것이다. nboard_mysql_user.sql로 해당 쿼리를 남겨두었다. 참조하여 변경한다.

권한 수정: 사용자 'nboard'@'localhost' - 데이터베이스 - 테이블

전체적 권한 (모두 체크 / 모두 체크안함)

주의: MySQL 권한 이름은 영어로 표기되어야 합니다.

데이터	구조	관리	리소스 제한
<input type="checkbox"/> SELECT <input type="checkbox"/> INSERT <input type="checkbox"/> UPDATE <input type="checkbox"/> DELETE <input type="checkbox"/> FILE	<input type="checkbox"/> CREATE <input type="checkbox"/> ALTER <input type="checkbox"/> INDEX <input type="checkbox"/> DROP <input type="checkbox"/> CREATE TEMPORARY TABLES <input type="checkbox"/> SHOW VIEW <input type="checkbox"/> CREATE ROUTINE <input type="checkbox"/> ALTER ROUTINE <input type="checkbox"/> EXECUTE <input type="checkbox"/> CREATE VIEW <input type="checkbox"/> EVENT <input type="checkbox"/> TRIGGER	<input type="checkbox"/> GRANT <input type="checkbox"/> SUPER <input type="checkbox"/> PROCESS <input type="checkbox"/> RELOAD <input type="checkbox"/> SHUTDOWN <input type="checkbox"/> SHOW DATABASES <input type="checkbox"/> LOCK TABLES <input type="checkbox"/> REFERENCES <input type="checkbox"/> REPLICATION CLIENT <input type="checkbox"/> REPLICATION SLAVE <input type="checkbox"/> CREATE USER	주의: 이 옵션을 0으로 하면 제한이 없어집니다. MAX QUERIES PER HOUR: <input type="text" value="0"/> MAX UPDATES PER HOUR: <input type="text" value="0"/> MAX CONNECTIONS PER HOUR: <input type="text" value="0"/> MAX USER CONNECTIONS: <input type="text" value="0"/>

실행

데이터베이스에 관한 권한

데이터베이스: nboard 사용권한: ALL PRIVILEGES 아니요 아니요 권한 수정 제거

다음 데이터베이스에 권한 추가하기:

실행

권한 수정: 사용자 'nboard'@'localhost' - 데이터베이스
nboard - 테이블

데이터베이스에 관한 권한 (모두 체크 / 모두 체크안함)

주의: MySQL 권한 이름은 영어로 표기되어야 합니다.

데이터	구조	관리
<input checked="" type="checkbox"/> SELECT <input checked="" type="checkbox"/> INSERT <input checked="" type="checkbox"/> UPDATE <input checked="" type="checkbox"/> DELETE	<input checked="" type="checkbox"/> CREATE <input checked="" type="checkbox"/> ALTER <input checked="" type="checkbox"/> INDEX <input checked="" type="checkbox"/> DROP <input checked="" type="checkbox"/> CREATE TEMPORARY TABLES <input checked="" type="checkbox"/> SHOW VIEW <input checked="" type="checkbox"/> CREATE ROUTINE <input checked="" type="checkbox"/> ALTER ROUTINE <input checked="" type="checkbox"/> EXECUTE <input checked="" type="checkbox"/> CREATE VIEW <input checked="" type="checkbox"/> EVENT <input checked="" type="checkbox"/> TRIGGER	<input type="checkbox"/> GRANT <input checked="" type="checkbox"/> LOCK TABLES <input checked="" type="checkbox"/> REFERENCES

실행

3. Apache 설정

httpd.conf에 가상호스트를 지정한다. 보안과 안정성을 위하여 절대 서브디렉터리에서 설치되어서는 안된다. 호스팅 사용자라면 그냥 웹 루트에 올리면 된다.

```
Listen 8088
NameVirtualHost *:8088
<VirtualHost *:8088>
    ServerName CI_naiyumie
    DocumentRoot "D:\workspace\w_git_workspace\wboard\wtrunk"
</VirtualHost>
```

4. CI 설정 #1

index.php에 ENVIRONMENT를 설정한다. Codeigniter매뉴얼을 참조하면 알 수 있다.
사이트가 개발중이라면 development로 값을 설정하고 실제 시연이라면 production으로 설정한다.

```
3 曰/*
4  -*-----
5  -*APPLICATION ENVIRONMENT
6  -*-----
7  -*
8  -*You can load different configurations depending on your
9  -*current environment. Setting the environment also influences
10 -*things like logging and error reporting.
11 -*
12 -*This can be set to anything, but default usage is:
13 -*
14 -*-----development
15 -*-----testing
16 -*-----production
17 -*
18 -*NOTE: If you change these, also change the error_reporting() code below
19 -*
20 曰*/
21 >> define('ENVIRONMENT', 'development');
22 >> //define('ENVIRONMENT', 'production');
```

4.1 CI 설정 #2

database.php에 호스팅 정보를 입력한다. username, password, database정도만 수정하면 된다.

```
$active_group = 'default';
$active_record = TRUE;

$db['default']['hostname'] = 'localhost';
$db['default']['username'] = 'nboard';
$db['default']['password'] = 'nboard1004gg';
$db['default']['database'] = 'nboard';
$db['default']['dbdriver'] = 'mysqli';
$db['default']['dbprefix'] = '';
$db['default']['pconnect'] = TRUE;
$db['default']['db_debug'] = TRUE;
$db['default']['cache_on'] = FALSE;
$db['default']['cachedir'] = '';
$db['default']['char_set'] = 'utf8';
$db['default']['dbcollat'] = 'utf8_general_ci';
$db['default']['swap_pre'] = '';
$db['default']['autoinit'] = TRUE;
$db['default']['stricton'] = FALSE;
```


4.2 CI 설정 #3

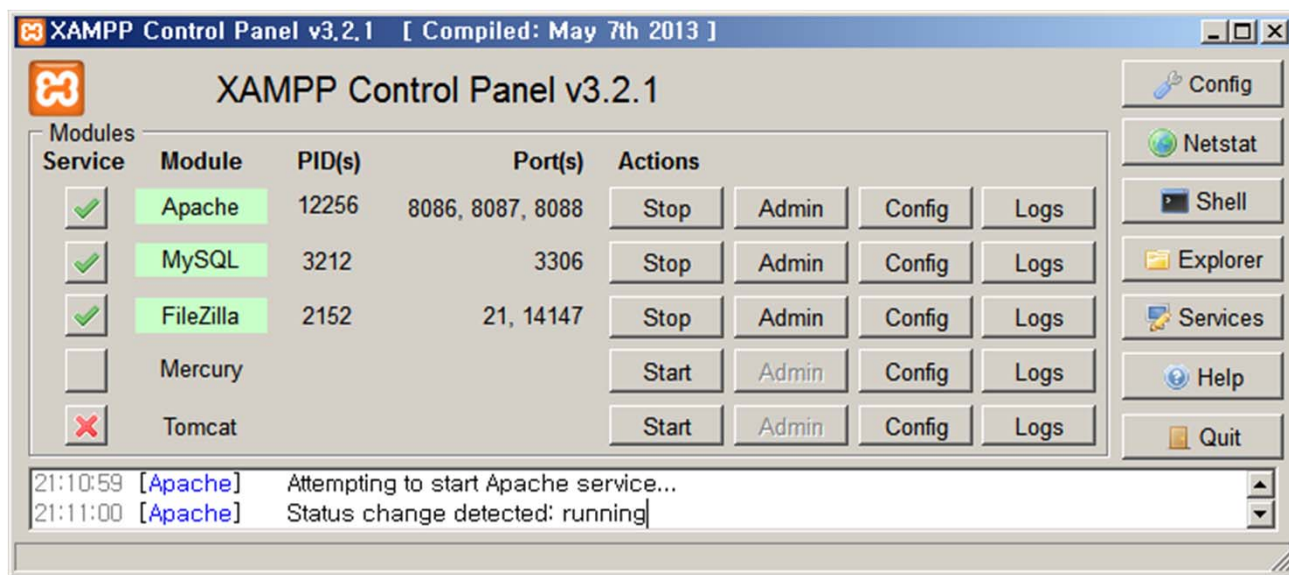
.htaccess는 딱히 건드릴 것이 없으나 참조한다.

```
<IfModule mod_rewrite.c>
> RewriteEngine On
> RewriteBase /
> RewriteCond $1 !^(index\.php|css|html|js|images|captcha|manual_assets|attach|admin_assets|admin_attach|assets|robots\.txt)
> RewriteCond %{REQUEST_FILENAME} !-f
> RewriteCond %{REQUEST_FILENAME} !-d
> RewriteRule ^(.*)$ /index.php/$1 [L]
</IfModule>

<FilesMatch "\.(htm|html|php)$">
SetHandler application/x-httpd-php
</FilesMatch>
```

5. 테스트#1

아파치서버를 구동한다. 호스팅 사용자의 경우 이미 아파치는 켜져 있을 것이다.



5.1 테스트#2

웹 브라우저를 띄워 접속해본다. 서브디렉터리 없이 가상호스트 루트에 설치되어야 올바르게 동작한다.

