**BLE Beacon 활용 통학버스재실 관리**

**시스템 구축 가이드**

**2019. 02**



목 차

[1. Beacon 및 Receiver 구현을 위한 Raspberry pi개발환경 셋팅 1](#_Toc499868228)

[2. Raspberry pi를 이용한 BLE beacon 및 Beacon Receiver 구현](#_Toc499868232) 3

[3. 서버구축 - 비트나미(Bitnami WAMP)설치 및 웹 서버 구축](#_Toc499868233) 5

[4. 서버구축 - PHP 및 MySQL 동작 확인](#_Toc499868233) 7

[5. 데이터 스키마 설정](#_Toc499868239) 9

[6. Application & 실시간 모니터링 프로그램의 서버 정보 수정 설정](#_Toc499868240) 14

**1. Beacon 및 Receiver 구현을 위한 Raspberry pi개발환경 셋팅**

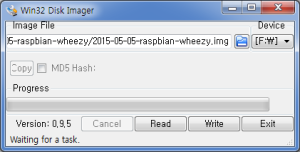
1. 라즈베리 파이 재단 홈페이지에서 리눅스 기반의 전용 운영체제 Raspbian을 다운받는다

토렌트로 다운로드:  
<http://downloads.raspberrypi.org/raspbian_latest.torrent>  
미러 사이트에서 다운로드:  
<http://downloads.raspberrypi.org/raspbian_latest>

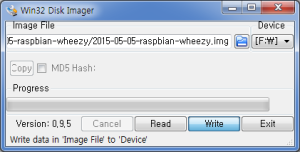
1. 포멧이 완료된 SD카드를 준비한다.
2. Raspbian 이미지를 SD 카드에 복사하기 위한 프로그램 Win32 Disk Imager를 다운 및 설치한다.

다운로드 : <http://sourceforge.net/projects/win32diskimager/>

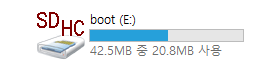
1. Win32 Disk Imager를 실행하여 폴더 아이콘을 선택 후 미리 다운받은 Raspbian을 선택한다.



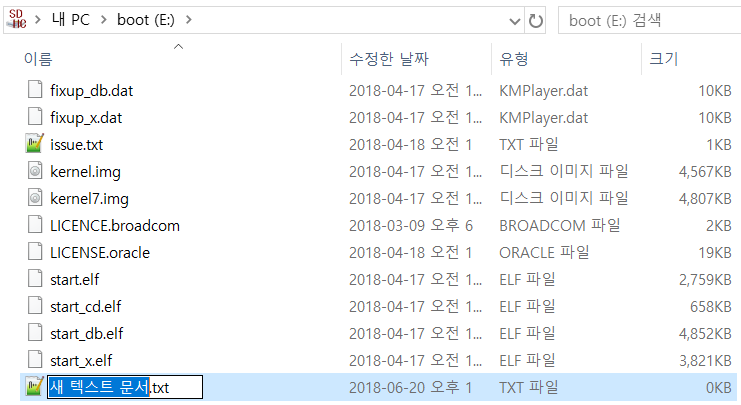
1. Write를 클릭한다.



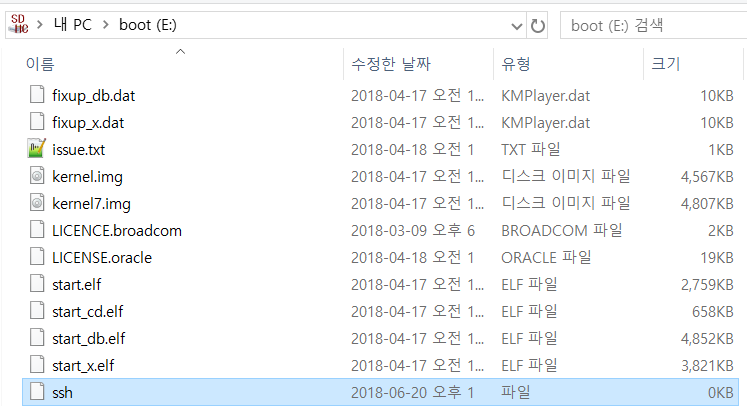
1. 이미지 복사가 완료된 후 마이크로 SD 카드의 마운트된 드라이브중 이름이 boot인 드라이브를 클릭한다



1. 파일 목록 빈공간에서 마우스 우클릭하여 보이는 메뉴에서 새로 만들기 > 텍스트 문서를 선택한다.



1. 확장자 명 “txt”를 포함하여 파일 이름을 모두 지우고 ssh라 입력하고 엔터를 입력한다.



1. 랜선을 통해 PC(혹은 노트북)으로 raspberry를 원격제어하기 위해서 mDNS를 사용한다.

mDNS란 Multicast Domain Name Service의 약자로 DNS(Domain Name Service)와 비슷한 개념입니다. DNS란 우리가 인터넷을 사용할 때 이용하고자 하는 사이트의 IP주소를 사용하는 것은 불편한 점이 많기 때문에 naver.com 같은 도메인 네임이라는 것을 만들어 IP주소에 연결해주는 서비스를 말합니다. mDNS를 받기 위해서는 간단하게 애플의 프린터 드라이버를 설치하면 됩니다. Bonjour(봉주르)란 애플이 만든 mDNS 소프트웨어의 등록된 상품명입니다.

Bonjour 다운로드 : http://support.apple.com/downloads/DL999/ko\_KR/BonjourPSSetup.exe

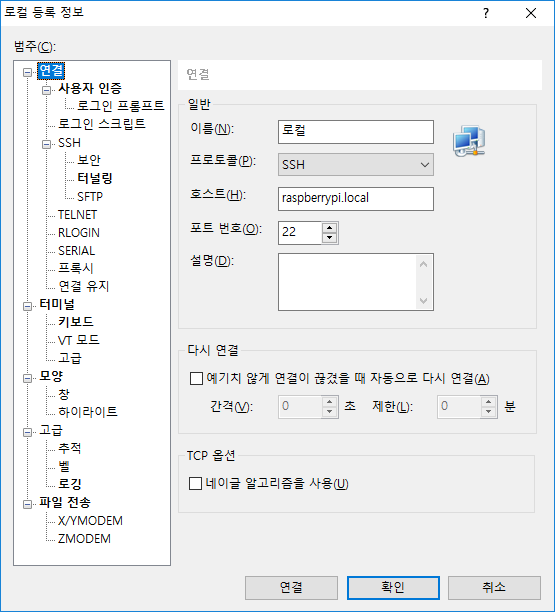
1. DHCP server for windows을 다운 및 압축을 풀어서 dhcpsrv.exe 바로 실행 (이더넷 카드가 AP모드로 변환)
2. SSH 원격제어를 위해 PC에 XShell 혹은 putty를 설치한다.

Xshell 다운로드 : https://www.netsarang.com/ko/free-for-home-school/

1. 이더넷 포트로 컴퓨터와 라즈베리파이를 연결한다.



1. Putty 혹은 Xshell를 켜고 아래와 같이 Host Name에 rapsberrypi.local을 입력



1. 설정한 정보를 통해서 라즈베리 파이에 접속

ID : pi

PW : raspberry

**2. Raspberry pi를 이용한 BLE beacon 구현**

1. 비콘과 리시버를 위한 라즈베리 파이에 SSH 원격 접속을 통해서 아래의 커맨드를 순서대로 입력하여 BlueZ 및 관련 라이브러리를 다운로드한다.

$ sudo apt-get -y install libusb-dev

$ sudo apt-get -y install libdbus-1-dev

$ sudo apt-get -y install libglib2.0-dev

$ sudo apt-get -y install libudev-dev

$ sudo apt-get -y install libical-dev

$ sudo apt-get -y install libreadline-dev

$ sudo apt-get -y install libdbus-glib-1-dev

$ sudo mkdir bluez

$ cd bluez

$ sudo wget www.kernel.org/pub/linux/bluetooth/bluez-5.19.tar.gz  
$ sudo gunzip bluez-5.19.tar.gz

$ sudo tar xvf bluez-5.19.tar cd bluez-5.19

$ sudo ./configure --disable-systemd

$ sudo make sudo make install

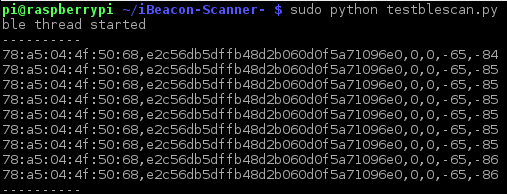
$ sudo apt-get install python-bluez

$ sudo shutdown -r now *(재부팅)*

1. 필요 프로그램 다운 및 비콘 스캔 예제 소스 실행하여 작동 확인

$ git clone https://github.com/switchdoclabs/iBeacon-Scanner-$ cd iBeacon-Scanner-

$ sudo python testblescan.py



위 그림처럼 나오면 BLE 기능이 잘 작동 하는 것이다.

1. 비콘의 경우 아래의 파일을 위의 과정을 마친 비콘용 라즈베리 파이의 iBeacon-Scanner- 폴더에 넣어서 해당 파일을 실행하면 Beacon 모드가 활성화 됨. (BLE\_beacon.py 파일)

<https://drive.google.com/file/d/1uzeaQ2Q3vJWVoFxvBhHkXM8H-VVa7auF/view?usp=sharing>

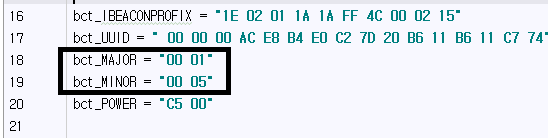
1. 리시버의 경우 아래의 파일을 위의 과정을 마친 리시버용 라즈베리 파이의 iBeacon-Scanner- 폴더에 넣고 해당 파일을 실행하면 Receiver 모드가 활성화 됨. (to\_db.py 파일)

<https://drive.google.com/file/d/1sx9UX-xl31UJlumFAlTUor8xkSbwzpf-/view?usp=sharing>

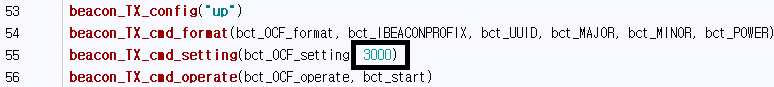
1. 추가로 리시버에서 온습조 센서를 사용하려면 아래의 파일을 리시버 라즈베리 파이의 pi/home에 넣어준다.

<https://drive.google.com/file/d/1qeu2WhavfeqUvf7WFLjm7rX8OnDqQRIp/view?usp=sharing>

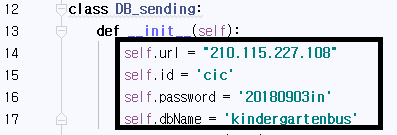
1. 비콘의 Major & Minor 정보를 바꾸고 싶다면 비콘에 넣어줬던 BLE\_beacon.py 소스의 아래 부분을 수정해 주면 됨. 여러 개의 비콘을 사용시 비콘마다 다른 Minor 정보가 반드시 입력되어야 한다.



1. 비콘의 전송주기를 바꾸고 싶다면 BLE\_beacon.py 소스의 아래 부분을 수정하면 됨. (현재 3초로 전송주기가 설정되어 있는 상태)



1. 새로운 서버를 구축하였다면 to\_db.py 파일의 아래 부분을 수정해 주어야 한다.

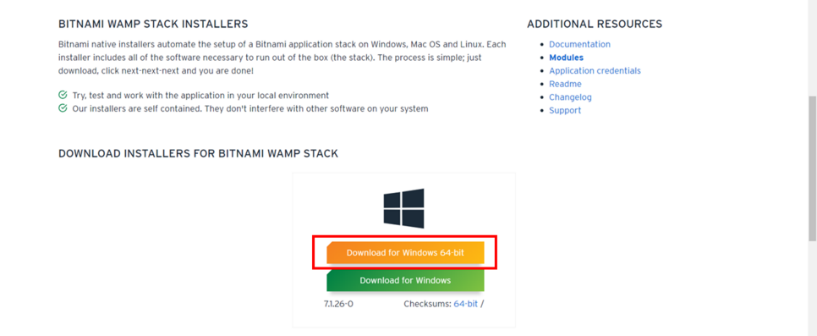


1. 라즈베리 파이가 부팅하고 바로 비콘과 리시버 역할을 수행하기 위해서는 각각 iBeacon-Scanner-폴더에 있는 BLE\_beacon.py와 to\_db.py파일을 자동 실행할 수 있도록 해야한다. 아래의 사이트를 참고하여 라즈베리 파이가 부팅시 자동으로 python 파일을 실행할 수 있도록 설정해준다.

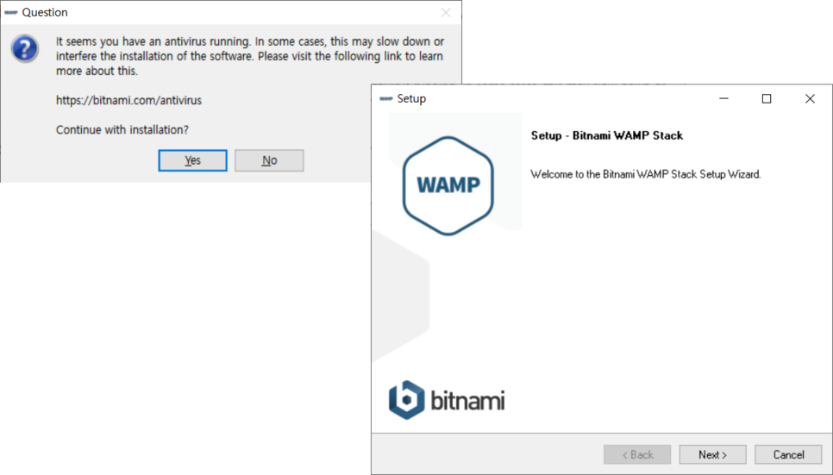
<https://www.dexterindustries.com/howto/run-a-program-on-your-raspberry-pi-at-startup/>

**3. 서버구축 - 비트나미(Bitnami WAMP)설치 및 웹 서버 구축**

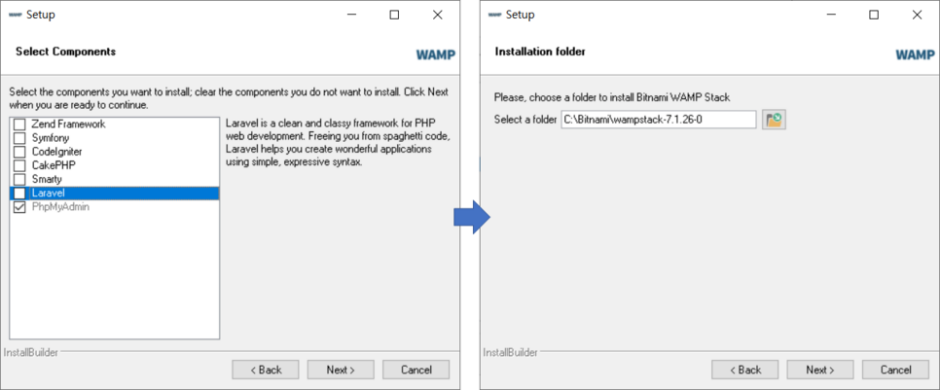
1. WAMP 설치를 위해서 https://bitnami.com/stack/wamp 사이트로 이동후 자신의 사양에 맞는 WAMP를 다운로드 한다..



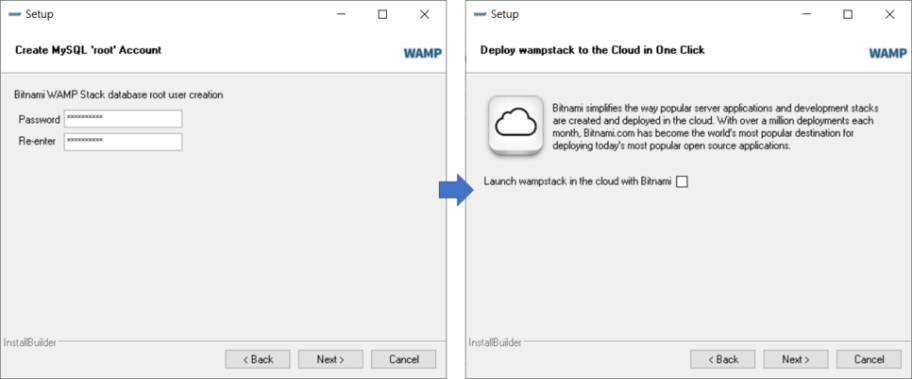
1. 다운로드 한 프로그램을 실행하면 왼쪽 상단과 같은 창이 뜨는데, Yes를 누르고 그 다음 창에서 Next를 누른다.

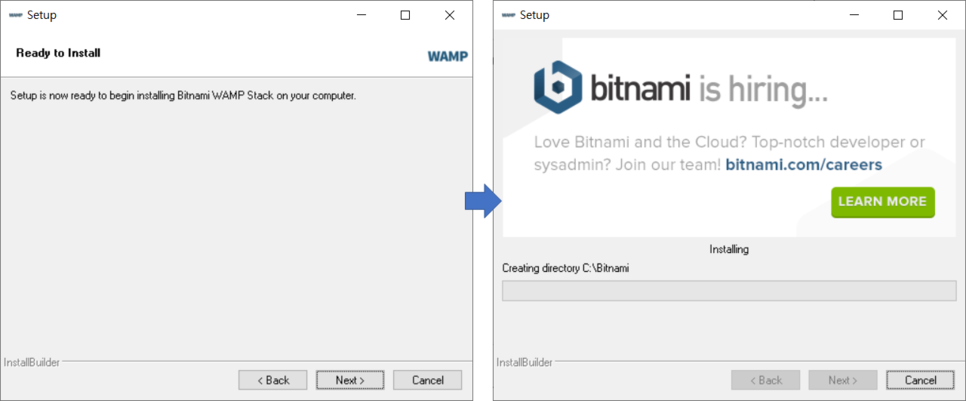


1. 왼쪽 창에서 체크를 다 해제하고 Next를 누른 후 기본 경로를 설정하고 Next를 누른다.

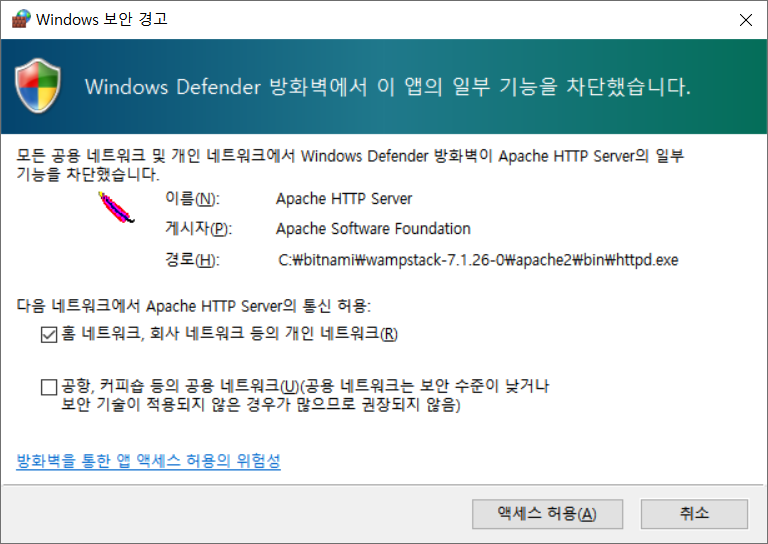


1. 그 후에 뜨는 창에서 데이터베이스의 root 계정 패스워드를 생성한다. 이 때 입력한 패스워드는 MySQL관리 계정이므로 기록해 둔다. 그 다음 화면에서는 넘어가서는 체크박스를 해제하고 설치가 시작될 때까지 Next를 누른다.

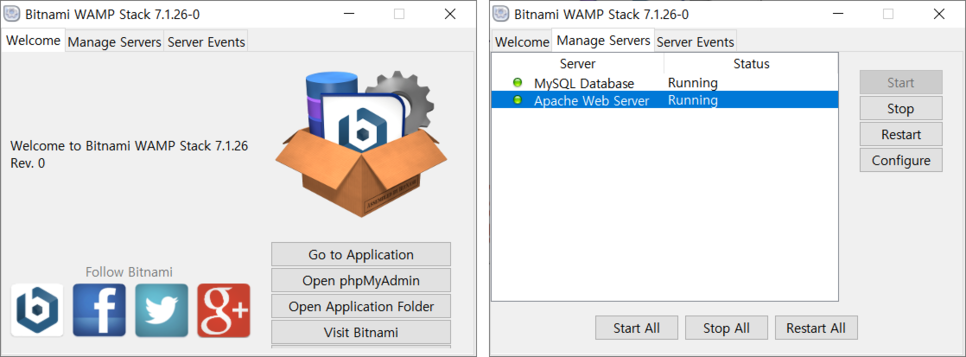




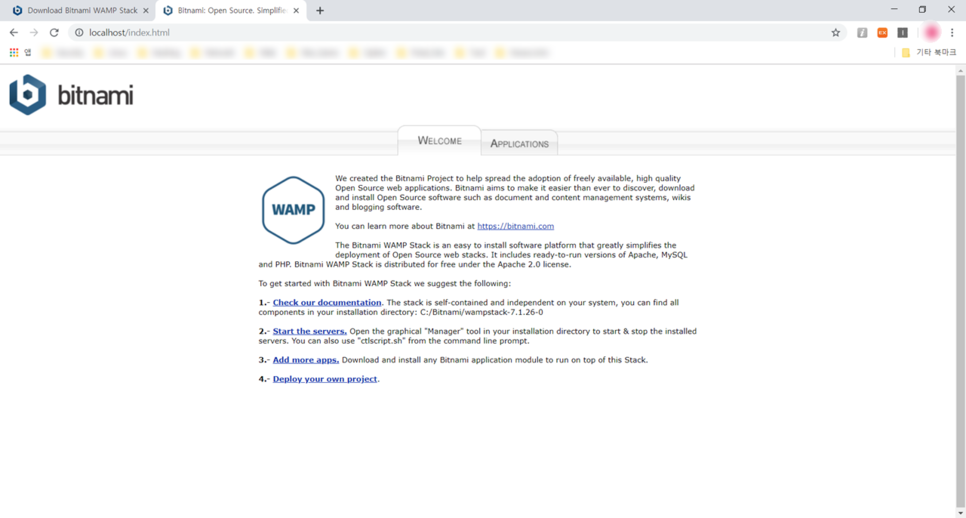
1. 엑세스 허용을 묻는 화면이 뜨면 엑세스를 허용해준다.



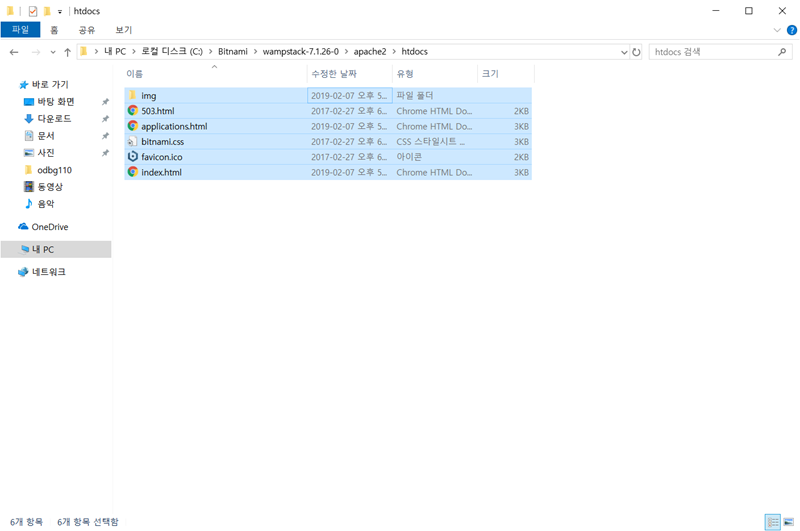
1. 설치 후 비트나미를 실행한다. C:\Bitnami\wampstack-7.1.26-0 (설치할 때 설정해준 경로) 경로에 manager-windows.exe를 누르면 실행된다.
2. 아래와 같은 창이 뜨면 Manage Servers 탭을 클릭하고, Apache Web Server가 Running 상태인지 확인한다.

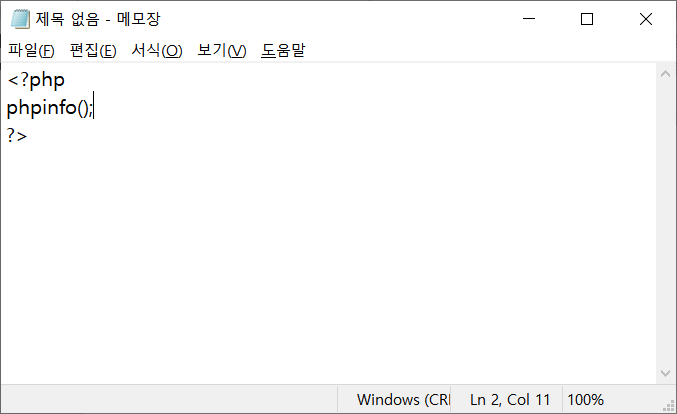


1. 웹 브라우저(Chrome, Internet Explorer 등)을 띄우고URL창에 localhost/index.html 또는 127.0.0.1/index.html을 쳐서 아래와 같은 창이 뜨면 아파치 웹 서버가 구축이 잘 된 것이다. ( localhost/index.html 또는 127.0.0.1/index.html이 안되는 경우 localhost:81/index.html 또는 127.0.0.1:81/index.html을 쳐서 확인한다. )

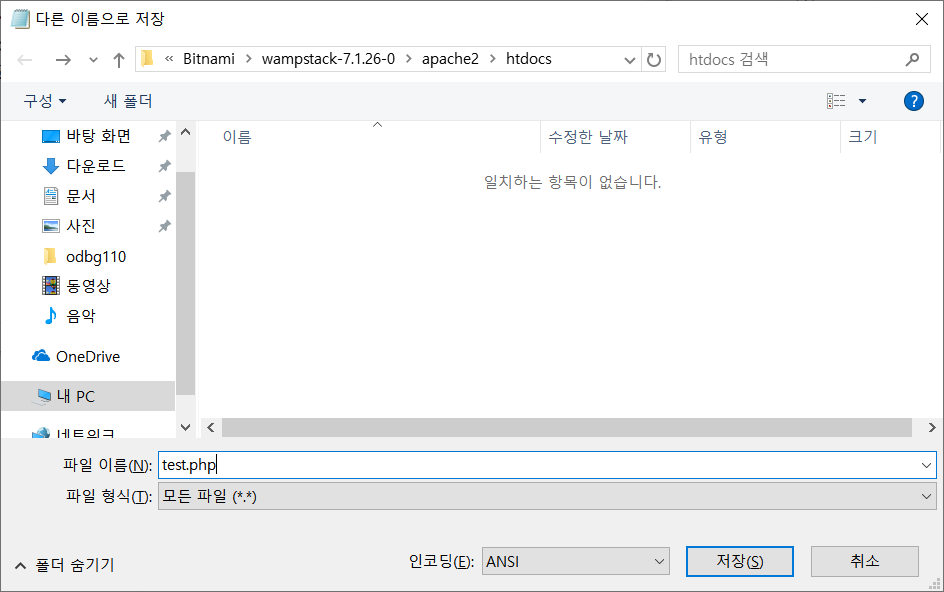


**4. 서버구축 - PHP 및 MySQL 동작 확인**

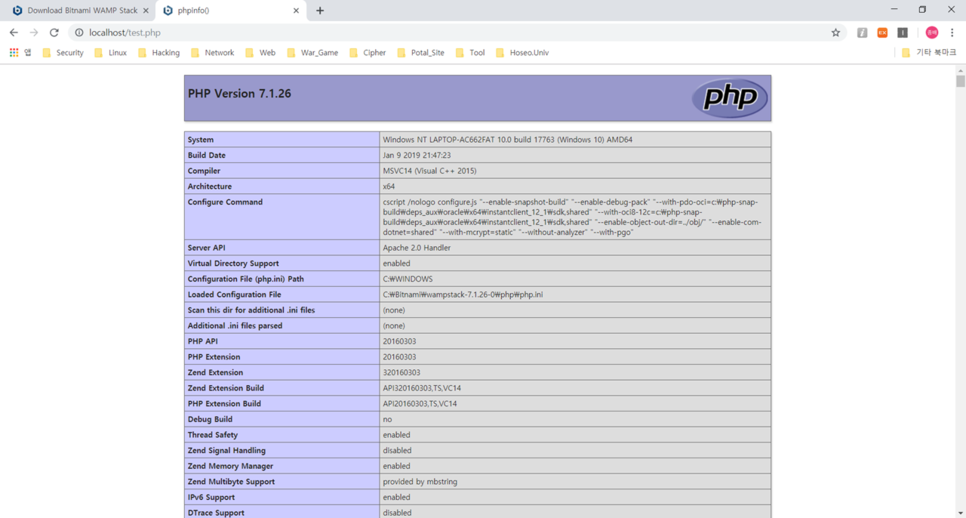
1. 아래와 같이 비트나미를 설치한 경로에서 apache2 폴더 안의 htdocs 폴더에 들어가면 파일들이 여러 개 있을 텐데 이 파일들을 삭제한다. (이 파일들은 웹 브라우저에서 localhost/index.html을 쳤을 때 나오는 화면을 구성해주는 파일이다.)
2. 메모장을 켜서 아래와 같은 내용을 입력한다.



1. htdocs폴더에 파일 이름을 “test.php”로 하고 파일 형식을 모든 파일(\*.\*)로 바꿔서 저장한다.



1. 웹 브라우저를 띄워서 “localhost/test.php” 또는 “127.0.0.1/test.php” 를 입력하고 아래와 같이 PHP 정보가 뜨면 잘 설치가 된 것을 확인 할 수 있다. ( “localhost/test.php” 또는 “127.0.0.1/test.php” 이 안되는 경우 “localhost:81/test.php” 또는 “127.0.0.1:81/test.php” 을 쳐서 확인한다. )



1. PHP가 잘 작동되는 것을 확인했으면 통학버스 승하치 시스템을 위한 PHP 파일을 비트나미를 설치한 경로에서 apache2 폴더 안의 htdocs폴더에 넣어준다.

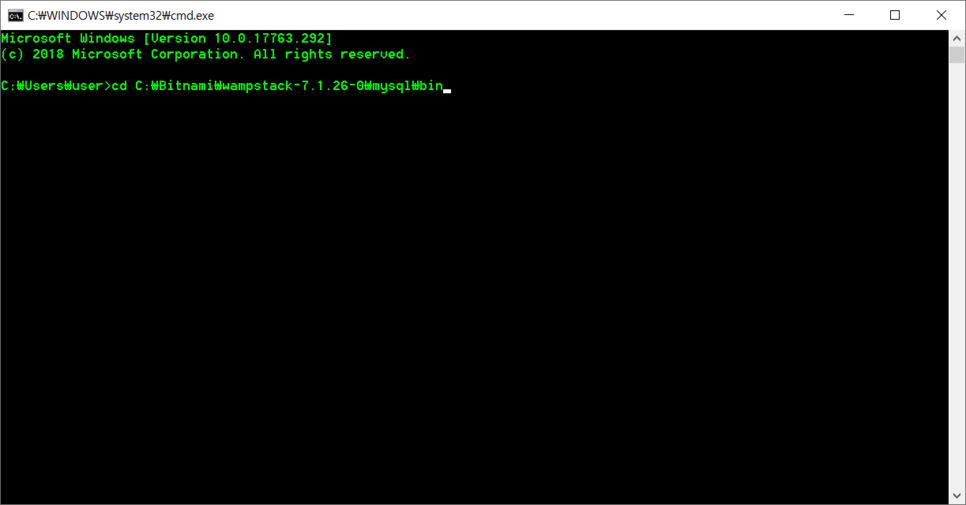
PHP 파일은 아래의 URL을 통해서 다운받을 수 있다. (안드로이드 프로젝트 포함 파일)

<https://drive.google.com/file/d/1CrGyKsfWLR2_dwLN1kmFgmtZEYTQH90d/view?usp=sharing>

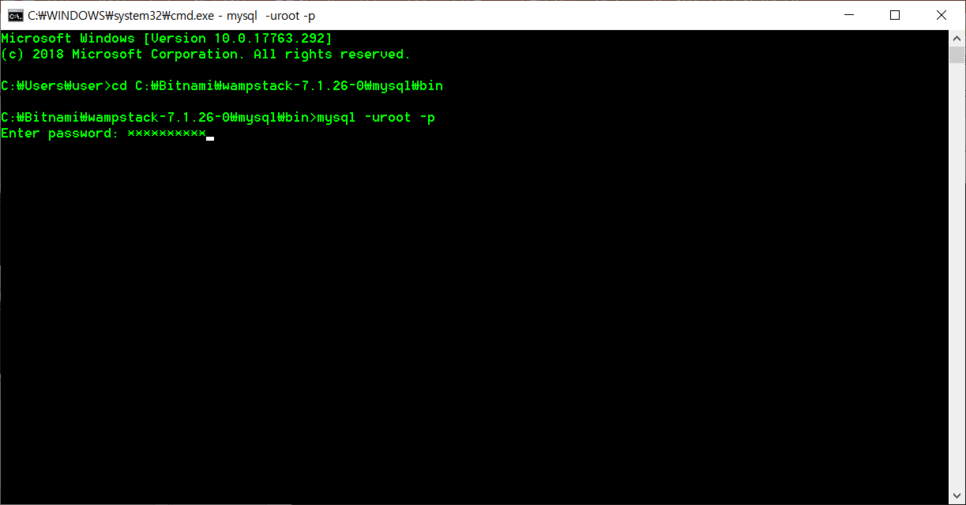
다운로드 받은 ZIP파일의 압축을 풀고 webserver/PHP 경로 안에 있는 모든 PHP 파일을 htdocs 폴더로 복사해주면 된다.

1. 아래와 같이 비트나미를 설치한 경로에서 mysql 폴더 안의 bin 폴더로 들어간 뒤 그 경로를 명령프롬프트창(cmd)에 아래의 예시처럼 입력한다.

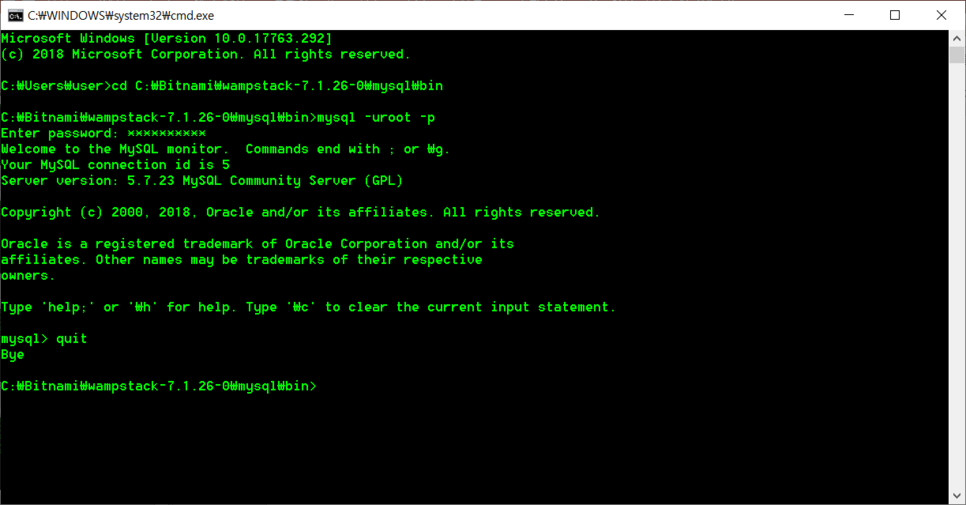
예시) cd C:\Bitnami\wampstack-7.1.26-0\mysql\bin



1. 명령프롬프트창(cmd)에 “mysql -uroot –p” 를 입력한다.



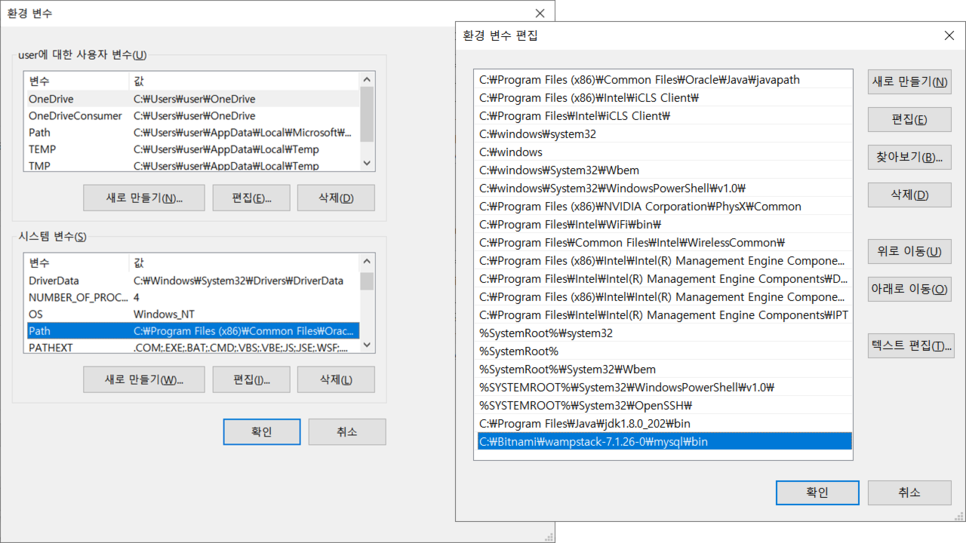
1. 비트나미를 설치할 때 등록한 MySQL 패스워드를 입력한다. (비트나미 설치 방법 4번 참고)
2. MySQL이 제대로 설치가 되었다면 아래와 같은 창이 보인다.



1. quit을 입력해 MySQL을 종료한다.
2. MySQL 환경변수 설정 방법은 아래와 같다

* 제어판 ▶ 시스템 및 보안 ▶시스템 ▶ 고급 시스템 설정 ▶환경변수를 클릭하면 아래의 왼쪽같은 창이 뜬다.
* 시스템 변수 에서 Path를 찾아서 편집을 누르고, 오른쪽 창과 같이 새로만들기를 눌러 bin 폴더까지의 경로를 입력한다.
* ( 예시와 같이 비트나미를 설치한 경로에서 mysql 폴더 안의 bin 폴더로 들어간 뒤 그 경로를 입력한다 .)

예시) C:\Bitnami\wampstack-7.1.26-0\mysql\bin



**5. 데이터 스키마 설정**

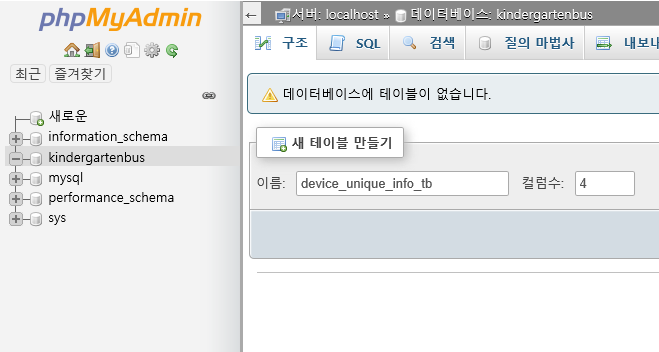
1. “http://localhost/phpmyadmin/” 또는 http://localhost:80/phpmyadmin으로 이동한다.
2. 계정과 비밀번호를 입력한다. 계정: root, 비밀번호: (비트나미를 설치할 때 설정했던 PW)
3. 아래와 같이 메뉴에서 ‘새로운’을 눌러 데이터베이스를 생성한다

* 데이터베이스 이름 : [kindergartenbus](http://210.115.227.108/phpmyadmin/db_structure.php?db=kindergartenbus)
* utf8\_general\_ci 항목은 체크해준다.

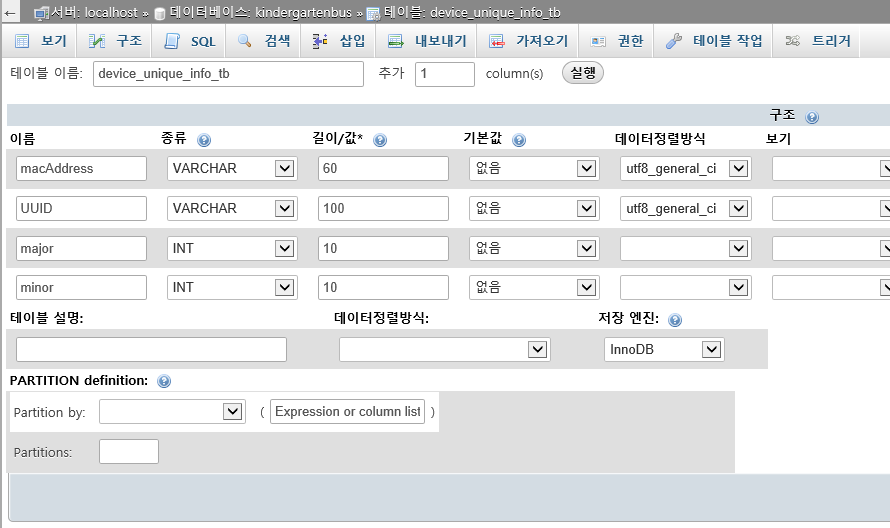


1. 아래와 같이 첫 번째 테이블을 생성한다.

* 테이블명 : device\_unique\_info\_tb
* 컬럼수 : 4

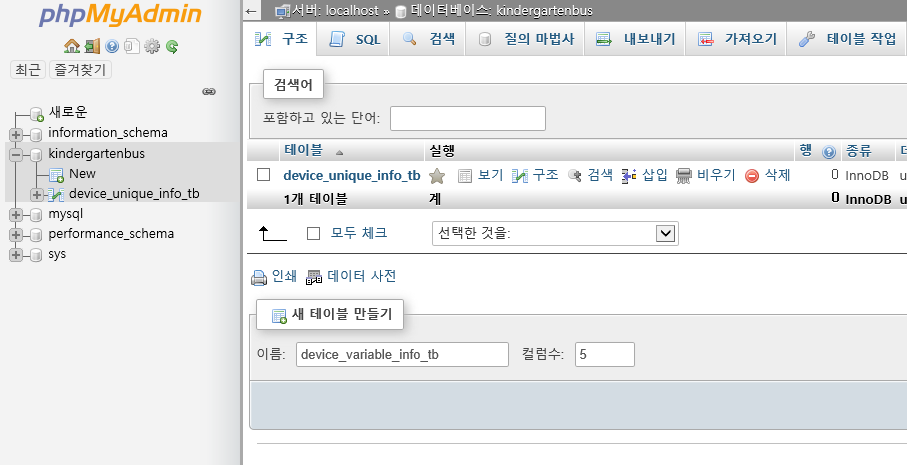


1. 첫 번째 테이블의 스키마 구조는 아래와 같이 입력한다.

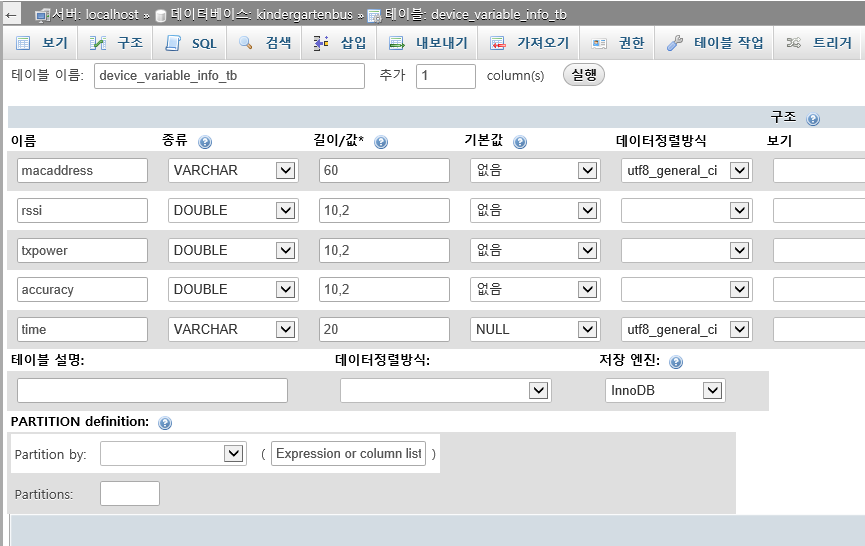


1. 아래와 같이 두 번째 테이블을 생성한다.

* 테이블명: device\_variable\_info\_tb
* 컬럼수: 5

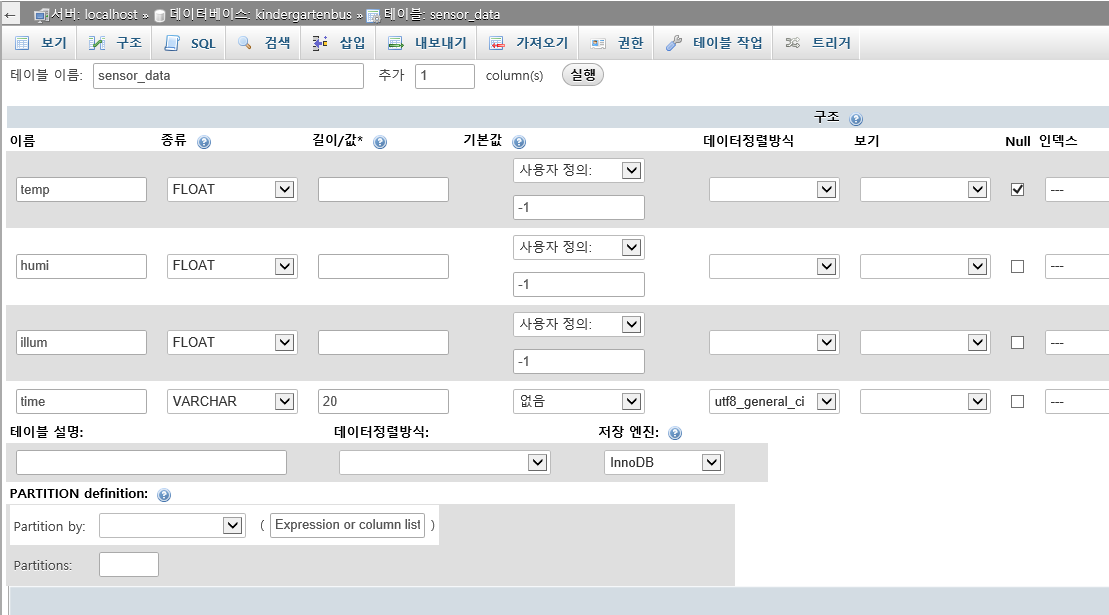


1. 두 번째 테이블의 스키마 구조는 아래와 같이 입력한다.

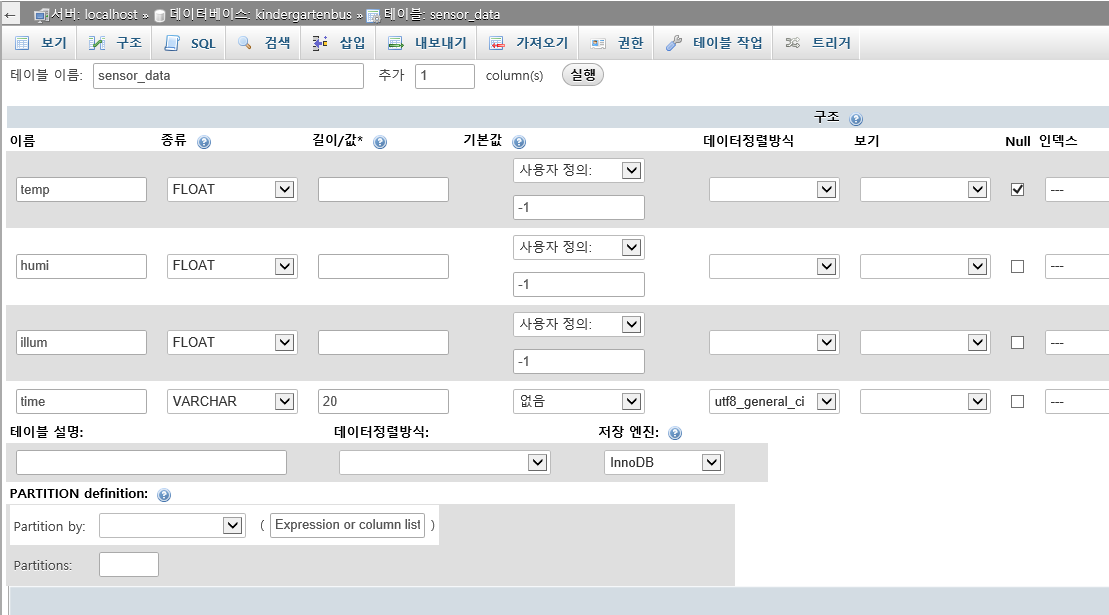


1. 아래와 같이 세 번째 테이블을 생성한다

* 테이블명 : sensor\_data
* 컬럼수 : 4

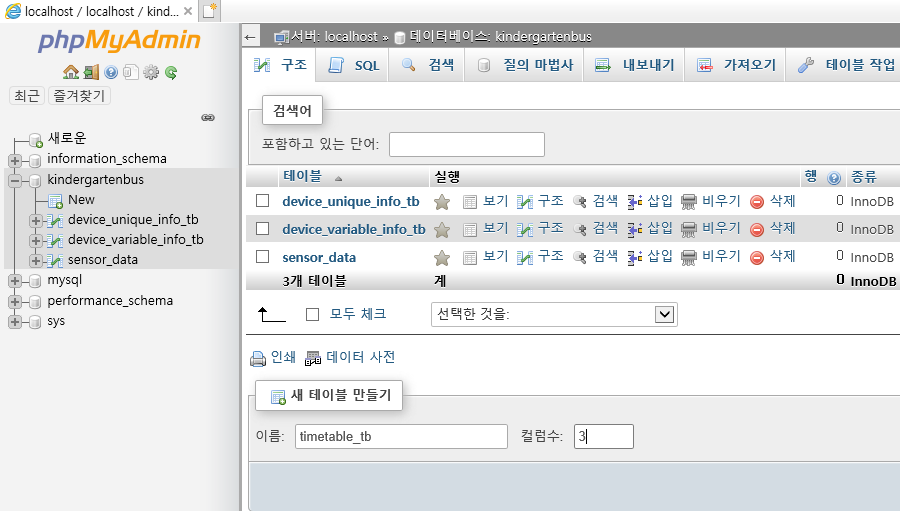


1. 세 번째 테이블의 스키마 구조는 아래와 같이 입력한다

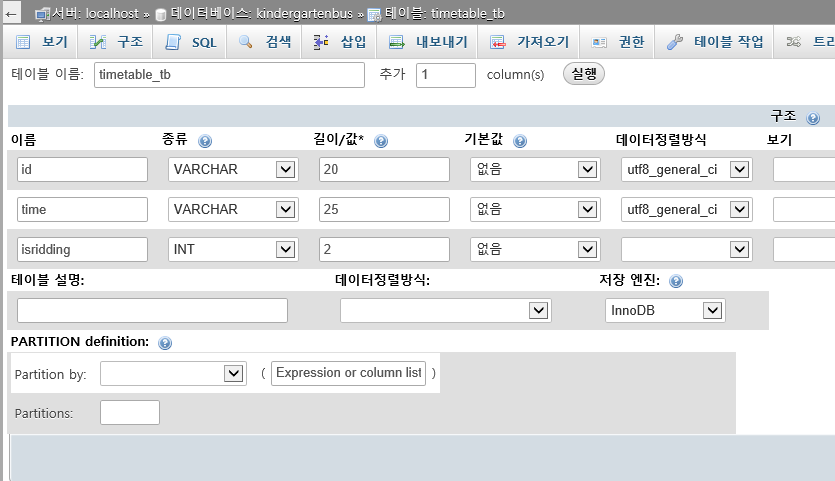


1. 아래와 같이 네 번째 테이블을 생성한다

* 테이블명 : timetable\_tb
* 컬럼수 : 3

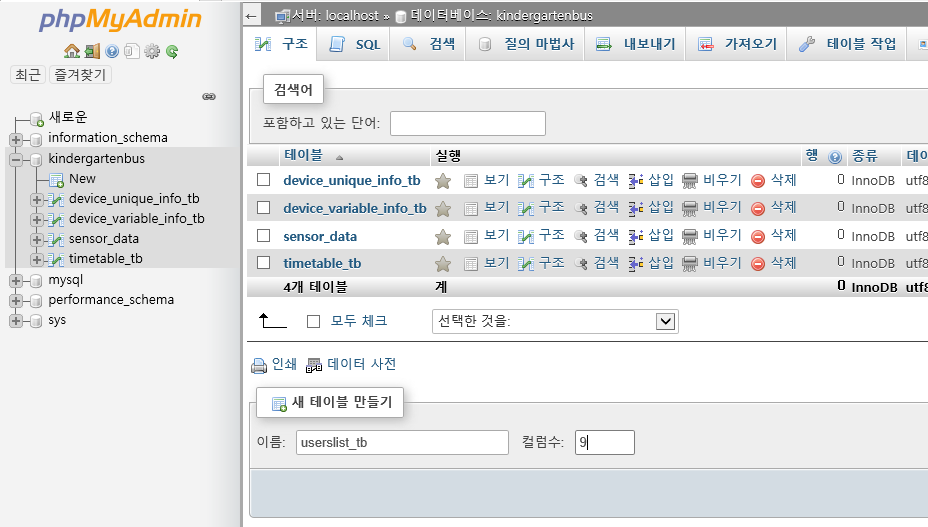


1. 네 번째 테이블의 스키마 구조는 아래와 같이 입력한다

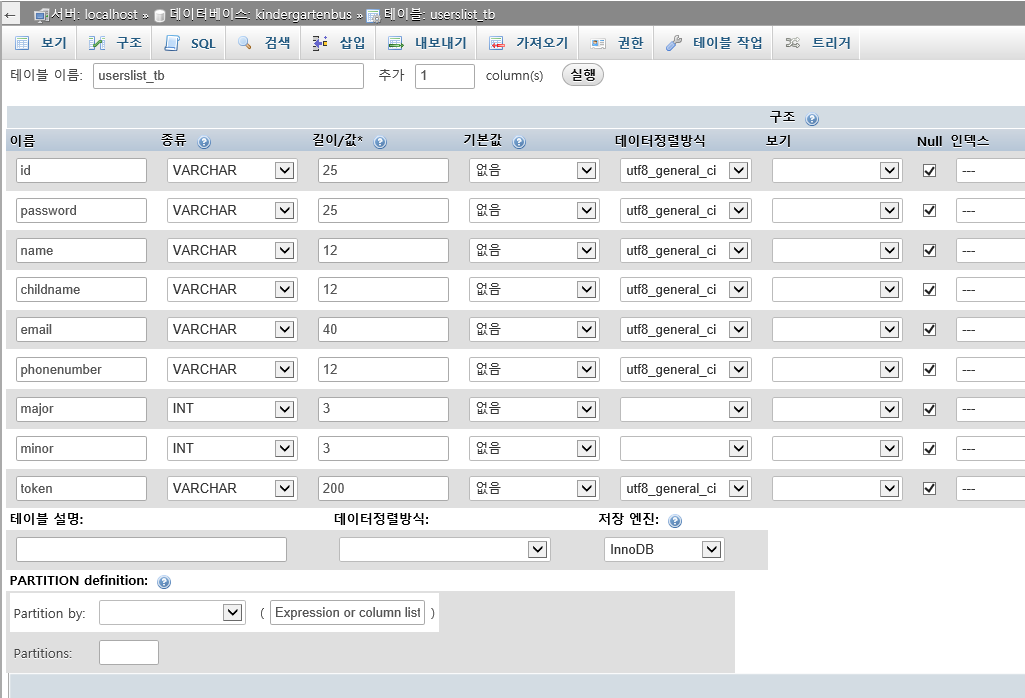


1. 아래와 같이 다섯 번째 테이블을 생성한다.

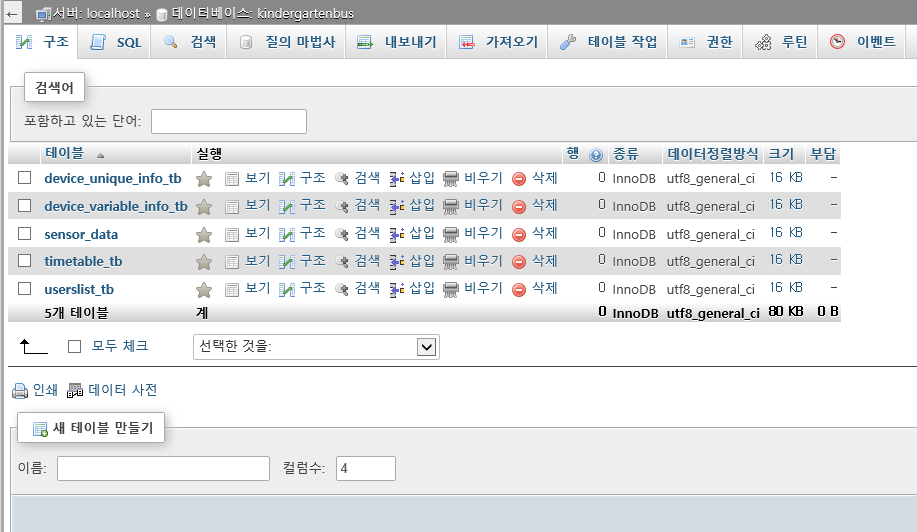
* 테이블명 : userslist\_tb
* 컬럼수 : 9



1. 다섯 번째 테이블의 스키마 구조는 아래와 같이 입력한다.



1. 최종적으로 아래 사진과 같이 5개의 테이블이 생성되었는지 확인한다.

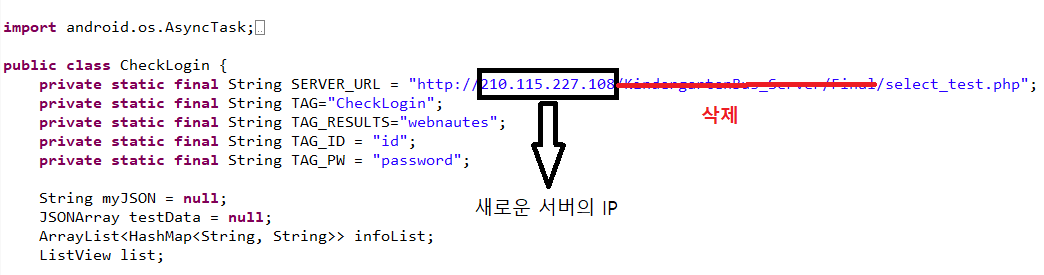


**6. Application & 실시간 모니터링 프로그램의 서버 정보 수정 설정**

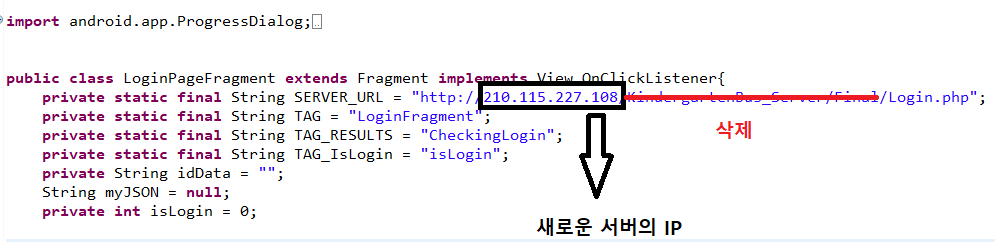
1. “**4. 서버구축 - PHP 및 MySQL 동작 확인**”의 **⑤**번 항목에서 이미 Android 프로젝트 파일과 “실시간 모니터링 및 푸시” 프로그램을 다운받았다.

* 참고로 “실시간 모니터링 및 푸시 프로그램”은 서버 혹은 안드로이드와는 독립적인 프로그램임으로 따로 실행시켜 주어야 한다.

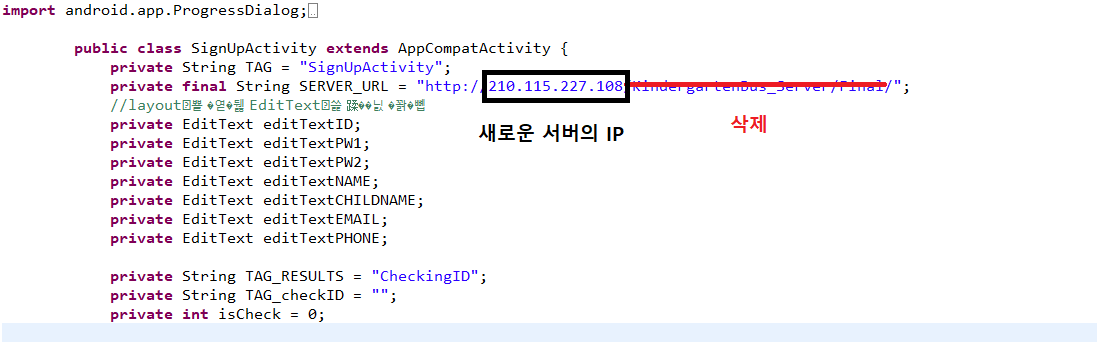
1. 새로운 서버를 구축했으므로 Android application과 “실시간 모니터링 및 푸쉬 프로그램”에 저장된 서버 정보를 새로 갱신해줘야 한다.
2. Android 프로젝트의 java 파일 중 “CheckLogin”, “LoginPageFragment”, “SignUpActivity” 이 세 파일을 수정해 주어야 한다 수정 방법은 아래와 같다.



**CheckLogin.java**

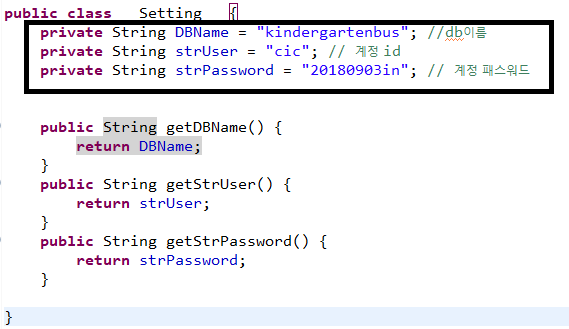
****

**LoginPageFragment.java**

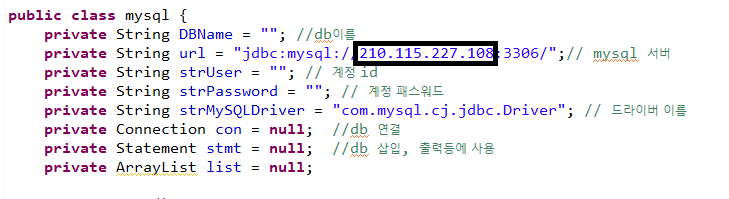
****

**SignUpActivity.java**

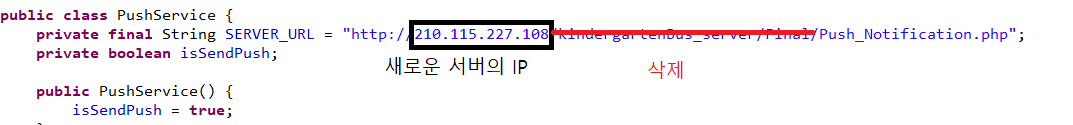
1. 그리고 “실시간 모니터링 및 푸시”의 서버 정보를 바꿔줘야 한다. “실시간 모니터링 및 푸시”는 Zip파일의 압축을 푼 후 webserber/java 폴더 안에 프로젝트 파일이 있다.
2. “실시간 모니터링 및 푸시” 프로젝트 안에 있는 자바 소스 중 총 3개의 소스를 수정해 주어야 한다.
3. \_\_Setting\_\_.java 코드 중 아래 네모 박스 안의 정보 중 유저 이름을 root, 비밀번호를 root의 비밀번호 비밀번호(비트나미 설치 때 설정함)로 바꿔준다. 만약 이 문서대로 DB 스키마를 설정했다면 DBName은 수정할 필요없다.



1. mysql.java 코드 중 아래 네모 박스 안의 IP주소를 새로운 서버의 IP로 바꿔준다.



1. PushService.java 코드 중 아래 네모 박스 안의 IP주소를 새로운 서버의 IP로 바꿔주고 빨간 선 부분은 지워준다.



1. “**2. Raspberry pi를 이용한 BLE beacon 및 Beacon Receiver 구현**”의 ⑧번 항목에서 Beacon Receiver 소스코드의 서버정보를 수정해줬는지 다시 한번 확인한다. IP는 새로운 서버의 IP, 유저 이름은 root, 비밀번호는 root의 비밀번호 비밀번호(비트나미 설치 때 설정함)로 바꿔준다. (to\_db.py)

