

ECE20023, Spring 2021



오픈소스 소프트웨어 실습

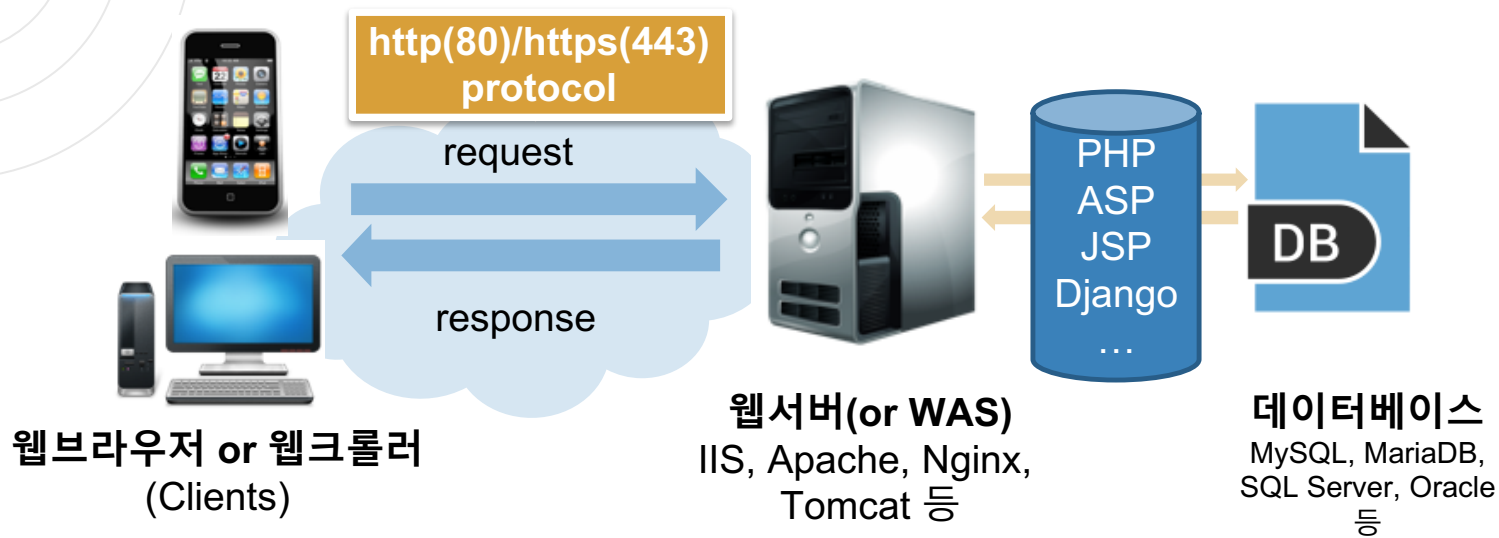




20

LEMP

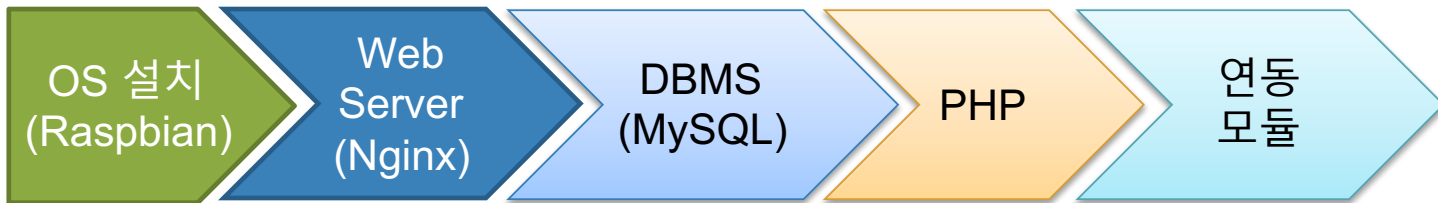
Web Service



LEMP stack 설치

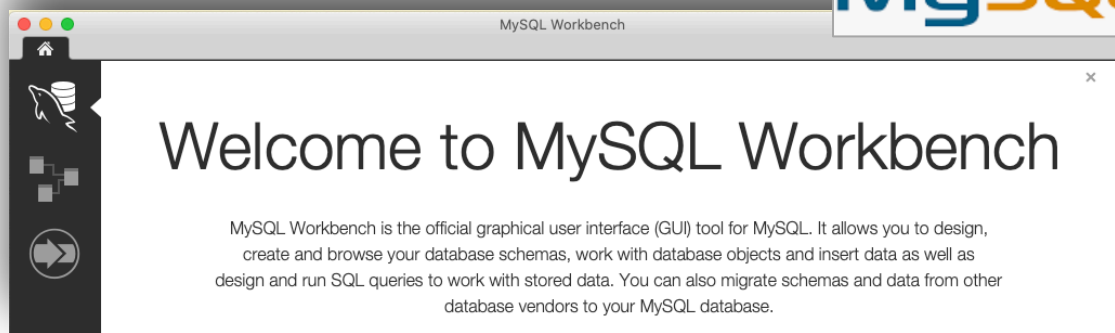


- 동적 웹 어플리케이션 개발
- **LEMP(LNMP)** = linux + **nginx** + MySQL (MariaDB) + php
- **LAMP** = linux + **apache** + MySQL (MariaDB) + php



MariaDB(MySQL)

- RDBMS (Relational Database management system)
- CLI 환경 제공, 데이터베이스 관리
- MySQL Workbench (GUI 툴)



MariaDB(MySQL) 설치



- 패키지 최신버전 업데이트 및 업그레이드
 - 업그레이드는 package 다운로드 및 설치
- MariaDB 설치: mariadb-server, mariadb-client
- MariaDB root 암호 설정 : mysql_secure_installation
- MariaDB 재시작
- MariaDB 실행 및 로그인

```
pi@myrpi:~ $ sudo apt update
pi@myrpi:~ $ sudo apt upgrade
pi@myrpi:~ $ sudo apt install mariadb-server
pi@myrpi:~ $ sudo apt install mariadb-client
pi@myrpi:~ $ sudo mysql_secure_installation
pi@myrpi:~ $ sudo mariadb -u root -p
pi@myrpi:~ $ sudo systemctl restart mariadb
```

Mariadb 실습 : DB생성



- MySQL 로그인(root)
- 새 데이터베이스 생성(testdb)
- 사용자 계정 생성 및 권한 부여(testuser)
 - testuser@localhost / testuser@'%' (외부접속)
- 변경된 권한 적용
- 로그아웃
- MySQL 로그인 테스트
- DB 확인

```
pi@myrpi:~ $ sudo mariadb -uroot -p
MariaDB [(none)]> create database testdb default
CHARACTER SET utf8;
MariaDB [(none)]> grant all on testdb.* to
testuser@localhost identified by '1234';
MariaDB [(none)]> flush privileges;
MariaDB [(none)]> quit
pi@myrpi:~ $ sudo mariadb -utestuser -p
```

Mariadb 실습



- 데이터베이스 관리 (DDL)
 - 생성 : create database [DB이름];
 - 확인 : show databases;
 - 선택 : use [DB이름];
 - 삭제 : drop database [DB이름];
- 테이블 / 필드 목록
 - 확인 : show tables;
 - 필드확인 : desc [Table이름];
- 사용자 관리 테이블 (root 로그인)
 - Mysql 데이터베이스 / user 테이블

```
MariaDB [mysql]> select host, user,  
password from user;
```


PHP



- 동적웹페이지 개발을 위한 프로그래밍 언어(server-side script)
- 최신버전 PHP 8.0.6 (21.05.06)
- 인터프리터로 해석
- php-fpm(PHP FastCGI Process Manager)를 통해 실행
- 절차지향, 객체지향 개발 가능
- 다양한 데이터베이스 연동, 파일 업로드, 메일전송, Excel 다운로드 등
- PHP 웹 프레임워크 : CodeIgniter, Laravel
- WordPress, Drupal, XE, Zeroboard 등의 개발언어로 사용

PHP 설치 및 테스트(1)



- 패키지 최신버전 업데이트
- php-fpm 설치 : 웹페이지 동적 처리(nginx 연동모듈)
- php-fpm 서비스 이름 찾기
- php-fpm 서비스 시작

```
pi@myrpi:~ $ sudo apt install php-fpm
pi@myrpi:~ $ systemctl | grep php
pi@myrpi:~ $ sudo service php7.3-fpm start
pi@myrpi:~ $ sudo service php7.3-fpm status
```

PHP 설치 및 테스트

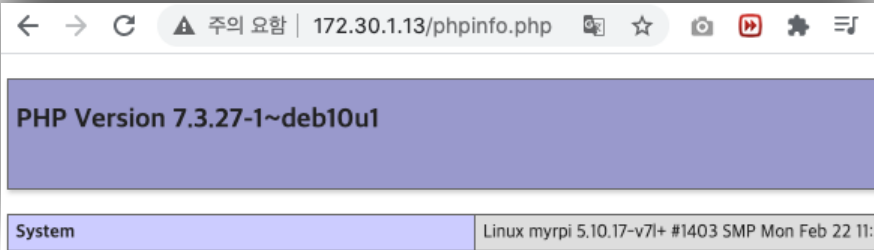


- php 서비스를 위한 웹서버 설정파일 변경
- Nginx 설정 파일 테스트
- Nginx 서비스 재시작
- php 파일 실행 테스트
 - Home directory
 - phpinfo.php

/etc/nginx/sites-available/default

```
# pass PHP scripts to FastCGI server
#
location ~ \.php$ {
    include snippets/fastcgi-php.conf;
    #
    # With php-fpm (or other unix sockets):
    fastcgi_pass unix:/run/php/php7.3-fpm.sock;
    # With php-cgi (or other tcp sockets):
    fastcgi_pass 127.0.0.1:9000;
}
```

```
<?php
    phpinfo();
?>
```



php-mysql / phpMyAdmin

- php-mysql 설치
 - php에서 mysql 사용을 위한 연동모듈
- phpMyAdmin 설치
 - MySQL 관리하도록 php로 제작된 웹서비스
 - 테이블 관리, 데이터 관리, 데이터 백업 등

```
pi@myrpi:~ $ sudo apt install php-mysql
pi@myrpi:~ $ sudo apt install phpmyadmin
```



phpMyAdmin 설정

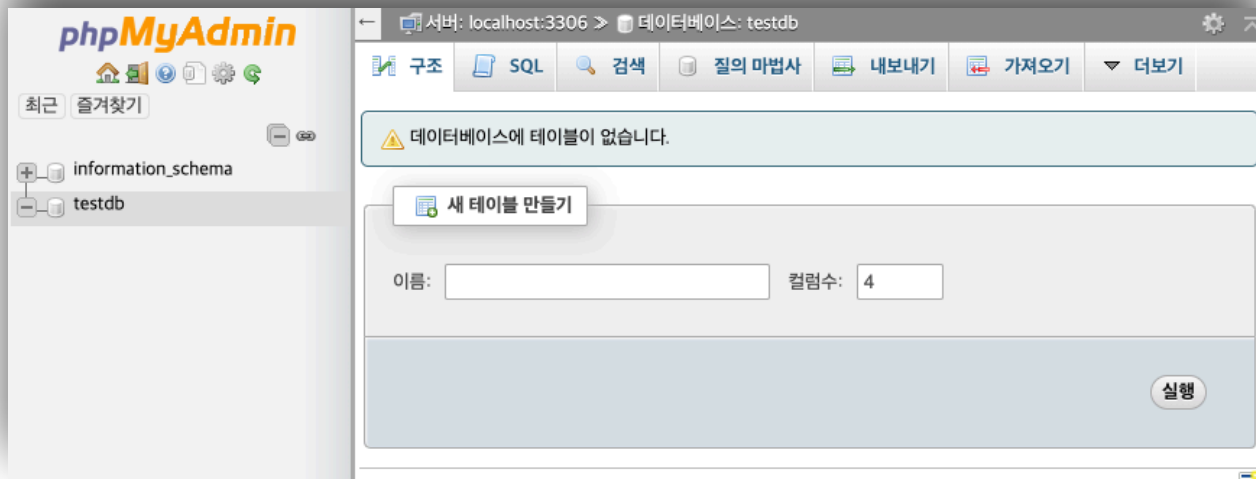
- phpMyAdmin 설치
- 홈디렉터리에 심볼릭 링크 생성 (phpmyadmin)
 - /usr/share/phpmyadmin
 - 브라우저 확인
- Nginx 설정파일에 설정 / 테스트
- Nginx 재시작

```
location /phpmyadmin {  
    index index.html index.php index.htm;  
}
```

```
pi@myrpi:/var/www/html $ sudo ln -s /usr/share/phpmyadmin phpmyadmin  
pi@myrpi:/var/www/html $ sudo vi /etc/nginx/sites-a*/default  
pi@myrpi:/var/www/html $ sudo nginx -t  
pi@myrpi:/var/www/html $ sudo service nginx restart
```

phpMyAdmin 로그인

- <http://라즈베리파이ip/phpmyadmin>
- 데이터베이스 id/pwd로 로그인 (testuser /1234)



Web hosting 실습(LEMP)

신청 정보 : LEMP 환경

- 계정정보 : ossuser / 1234
- 도메인명 : ossweb.com
- DB계정 : ossdbuser / 1234
- DB이름 : ossdb

- 사용자 계정 추가
 - 사용자 계정, 암호설정, home directory, index.html 추가, phpinfo.php, 소유자 변경, phpmyadmin 링크 추가
- Virtual host 설정
 - 설정파일 추가, php , phpmyadmin 설정 추가
 - 설정파일 테스트, 웹서버 재시작
- MySQL DB / 계정 추가
 - db추가, 계정 추가, 로그인 테스트
- 사용자 테스트
 - 웹페이지 테스트, ftp테스트, phpmyadmin테스트

과제설명

1. 다음의 계정 정보를 이용하여 라즈베리파이에 LEMP 환경을 구축하세요.

- 계정정보 : ossuser / 1234
- 도메인명 : ossweb.com
- DB계정 : ossdbuser / 1234
- DB이름 : ossdb

1. 사용자 계정 추가(터미널에서 사용된 명령어만 텍스트 복사)

a. 사용자 계정, 암호설정, home directory, index.html 추가, phpinfo.php, 소유자 변경, phpmyadmin 링크 추가

2. Virtual host 설정(터미널에서 사용된 명령어만 텍스트 복사)

a. 설정파일 추가, php , phpmyadmin 설정 추가

3. 설정파일 테스트(터미널에서 사용된 명령어만 텍스트 복사)

4. 웹서버 재시작(각 페이지 브라우저 화면 캡처)

a. index.html 페이지, phpinfo.php 페이지, myadmin 링크

5. MySQL 데이터베이스 / 계정 추가(터미널에서 사용된 명령어만 텍스트 복사)

a. db추가, 계정 추가, 계정권한 저장, 로그인 테스트

6. 사용자 테스트(각 페이지 브라우저 화면 캡처)

a. 웹페이지 테스트, ftp테스트, phpmyadmin테스트