05_APM/NPM_Compile(CentOS6) 설치

1. Apache 설치

1.1 기존에 설치된 Apache 웹서버 제거

```
$ yum remove -y httpd httpd-*
```

1.2 빌드 환경 설정

```
$ yum -y install gcc gcc-c++ autoconf libjpeg libjpeg-devel libpng libpng-devel
freetype freetype-devel libxml2 libxml2-devel zlib zlib-devel glibc glibc-devel
glib2 glib2-devel bzip2 bzip2-devel  ncurses ncurses-devel curl curl-devel
e2fsprogs e2fsprogs-devel krb5 krb5-devel libidn libidn-devel openssl openssl-
devel libtool  libtool-libs openldap openldap-devel nss_ldap openldap-clients
openldap-servers libtool-ltdl libtool-ltdl-devel bison expat-devel
```

1.3 관련 모듈 다운로드 및 설치

- apr 다운로드

```
$ cd /usr/local/src

$ wget https://archive.apache.org/dist/apr/apr-1.5.2.tar.gz

$ tar xvfz apr-1.5.2.tar.gz

$ rm -rf apr-1.5.2.tar.gz

$ cd apr-1.5.2

$ ./configure --prefix=/usr/local/apr

$ make

$ make install
```

- apr-util 다운로드

```
$ cd /usr/local/src

$ wget http://archive.apache.org/dist/apr/apr-util-1.5.4.tar.gz

$ tar xvzf apr-util-1.5.4.tar.gz

$ rm -rf apr-util-1.5.4.tar.gz

$ cd apr-util-1.5.4

$ ./configure --with-apr=/usr/local/apr/
```

```
$ make

$ make install
```

- pcre 다운로드

```
$ cd /usr/local/src

$ wget http://downloads.sourceforge.net/project/pcre/pcre/8.37/pcre-
8.37.tar.gz

$ tar xvfz pcre-8.37.tar.gz

$ rm -rf pcre-8.37.tar.gz

$ cd pcre-8.37

$ ./configure --prefix=/usr/local/pcre

$ make

$ make install
```

1.4 Apache 웹서버 다운로드 및 설치

```
$ cd /usr/local/src

$ wget http://archive.apache.org/dist/httpd/httpd-2.4.23.tar.gz

$ tar xvfz httpd-2.4.23.tar.gz

$ rm -rf httpd-2.4.23.tar.gz

$ mv apr-1.5.2 httpd-2.4.23/srclib/apr

$ mv apr-util-1.5.4 httpd-2.4.23/srclib/apr-util

$ cd httpd-2.4.23

$ ./configure --prefix=/usr/local/apache \
    --enable-module=so \
    --enable-mods-shared=most \
    --enable-maintainer-mode \
    --enable-deflate \
    --enable-headers \
    --enable-rewrite \
    --enable-ssl \
    --enable-proxy \
    --enable-proxy-http \
    --enable-proxy-ajp \
    --enable-proxy-balance \
    --with-pcre=/usr/local/pcre

$ make
```

```
$ make install
```

1.5 Apache 웹서버 서비스 등록 및 실행

- httpd 서비스 파일 만들기

```
$ cp /usr/local/apache/bin/apachectl /etc/init.d/httpd
```

- vi 애디터로 파일을 열고 내용 추가

```
$ vi /etc/init.d/httpd
```

```
#! /bin/sh
# chkconfig: 2345 90 90
# description: init file for Apache server daemon
# processname: /usr/local/apache/bin/apachectl
# config: /usr/local/apache/conf/httpd.conf
# pidfile: /usr/local/apache/logs/httpd.conf
```

- httpd.conf 파일 수정

```
$ vi /usr/local/apache/conf/httpd.conf
```

  - ServerName 부분을 찾아 주석을 제거하고 사용할 서버명을 입력(여기서는 localhost)
  - `# LoadModule unique_id_module modules/mod_unique_id.so` 추가

- httpd 서비스 시작

```
$ service httpd start
```

1.6 방화벽 설정

- vi 애디터로 파일을 열고 #부분 내용 추가(추가할때는 # 제거)

```
$ vi /etc/sysconfig/iptables
```

```
-A INPUT -m state --state NEW -m tcp -p tcp --dport 22 -j ACCEPT
# -A INPUT -m state --state NEW -m tcp -p tcp --dport 80 -j ACCEPT
# -A INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT
-A INPUT -j REJECT --reject-with icmp-host-prohibited
```

- 방화벽 재시작

```
$ service iptables restart
```

2. MySQL 설치

2.1 의존성 파일 설치

```
$ yum -y install ncurses-devel zlib curl libtermcap-devel lib-client-devel
bzip2-devel cmake bison perl perl-devel
```

- cmake 다운로드 및 설치

```
$ cd /usr/local/src

$ wget https://cmake.org/files/v3.5/cmake-3.5.2.tar.gz

$ tar xvfz cmake-3.5.2.tar.gz

$ rm -rf cmake-3.5.2.tar.gz

$ cd cmake-3.5.2

$ ./bootstrap

$ make && make install
```

2.2 리눅스 계정 생성

```
$ groupadd mysql
$ useradd -g mysql mysql
```

2.3 MySQL 다운로드

```
$ cd /usr/local/src

$ wget https://dev.mysql.com/get/Downloads/MySQL-5.5/mysql-5.5.14.tar.gz

$ tar xvfz mysql-5.5.14.tar.gz

$ rm -rf mysql-5.5.14.tar.gz

$ cd mysql-5.5.14
```

2.4 MySQL cmake 컴파일

```
$ cmake \
    -DCMAKE_INSTALL_PREFIX=/usr/local/mysql \
    -DMYSQL_DATADIR=/usr/local/mysql/data \
    -DMYSQL_UNIX_ADDR=/tmp/mysql.sock \
    -DSYSCONFDIR=/etc \
    -DDEFAULT_CHARSET=utf8 \
    -DMYSQL_TCP_PORT=3306 \
    -DWITH_EXTRA_CHARSETS=all \
    -DDFAULT_COLLATION=utf8_general_ci \
    -DDOWNLOAD_BOOST=1 \
    -DWITH_BOOST=/usr/include/boost \
    -DWITH_INNOBASE_STORAGE_ENGINE=1 \
    -DWITH_PERFSCHEMA_STORAGE_ENGINE=1 \
    -DENABLED_LOCAL_INFILE=1
```

```
$ make -j 2 && make install
```

2.5 환경설정 기본 파일 복사 및 수정

```
$ cd /usr/local/mysql/support-files

$ cp my-huge.cnf /etc/my.cnf
```

- vi 애디터로 파일 열고 [mysqld] 부분에 추가

```
$ vi /etc/my.cnf
```

```
[mysqld]
character-set-server = utf8
collation-server = utf8_general_ci
character-set-client-handshake = false
```

2.6 서비스 스크립트 및 서비스 등록

```
$ cp mysql.server /etc/rc.d/init.d/mysqld
```

- vi 애디터로 파일 열고 내용 추가

```
$ vi /etc/rc.d/init.d/mysqld
```

```
basedir=/usr/local/mysql
datadir=/usr/local/mysql/data
```

2.7 MySQL 초기화

```
$ cd /usr/local/mysql

$ ./scripts/mysql_install_db --user=mysql --basedir=/usr/local/mysql --
datadir=/usr/local/mysql/data --defaults-file=/etc/my.cnf
```

2.8 MySQL 그룹/계정 권한 주기

```
$ chown -R mysql:mysql /usr/local/mysql

$ chown -R mysql:mysql /usr/local/mysql/data

$ chmod 755 /etc/rc.d/init.d/mysqld
```

2.9 MySQL 구동 시작

```
$ service mysqld start

$ service mysqld stop
```

2.10 부팅 시 자동 실행하기 설정

```
$ chkconfig --add mysqld
```

2.11 주요 기능 PATH 등록

```
$ ln -s /usr/local/mysql/bin/mysql /usr/bin/mysql

$ ln -s /usr/local/mysql/bin/mysqldump /usr/sbin/mysqldump

$ ln -s /usr/local/mysql/bin/mysql_config /usr/sbin/mysql_config

$ ln -s /usr/local/mysql/bin/mysqladmin /usr/sbin/mysqladmin

$ ln -s /usr/local/mysql/support-files/mysql.server /etc/rc.d/init.d/mysql
```

- vi 애디터로 환경 변수 설정

  ```
  $ vi /root/.bash_profile
  ```

  ```
  PATH=$PATH$HOME/bin:/usr/local/mysql/bin:
  ```

  ```
  $ source /root/.bash_profile
  ```

2.12 MySQL root 계정 비밀번호 변경

```
$ service mysqld start

$ mysqladmin -u root password root
```

2.13 리눅스 시작시 MySQL 구동되도록 설정

```
$ chkconfig --add mysqld

$ chkconfig mysqld on

$ chkconfig --list mysqld
```

3. PHP 설치

3.1 설치에 필요한 패키지

```
$ yum -y install libxml2-devel

$ yum -y install openssl-devel

$ yum -y install libjpeg-devel

$ yum -y install libpng-devel

$ yum install gmp-devel
```

- sqlite3 최신버전 교체

```
$ cd /usr/local/src

$ wget https://www.sqlite.org/2020/sqlite-autoconf-3310100.tar.gz

$ tar xvfz sqlite-autoconf-3310100.tar.gz

$ cd sqlite-autoconf-3310100

$ ./configure --prefix=/usr/local/src/sqlite

$ make && make install
```

```
# 기존 sqlite3와 방금 설치한 sqlite3를 교체
$ /usr/local/src/sqlite/bin/sqlite3 --version   # 방금 설치한 sqlite3 버전 확인
3.31.1

$ /usr/bin/sqlite3 --version    # CentOS6과 함께 제공되는 sqlite3 버전
3.6.20

$ sqlite3 --version # sqlite3의 버전이 여전히 이전 버전임으로 업데이트 필요
3.6.20

# 이전 sqlite3 옮기기
$ mv /usr/bin/sqlite3 /usr/bin/sqlite3_old

# 방금 설치한 버전 링크
$ ln -s /usr/local/src/sqlite/bin/sqlite3 /usr/bin/sqlite3

# 시스템의 버전 확인
$ sqlite3 --version
3.31.1

$ export LD_LIBRARY_PATH=/usr/local/src/sqlite/lib

$ echo $PKG_CONFIG_PATH

$ export PKG_CONFIG_PATH=/usr/lib64/pkgconfig

$ export PKG_CONFIG_PATH=/usr/local/src/sqlite/lib/pkgconfig
```

3.2 PHP 다운로드 및 설치

```
$ cd /usr/local/src

$ wget https://www.php.net/distributions/php-7.4.3.tar.gz

$ tar xvfz php-7.4.3.tar.gz

$ cd php-7.4.3  # sqlite3 문제 발생 시 해결 후 cd /usr/local/src/php-7.4.3

$ ./configure \
    --prefix=/usr/local/php \
    --with-config-file-path=/usr/local/php/etc \
    --with-apxs2=/usr/local/apache/bin/apxs \
    --with-mysqli=/usr/local/mysql/bin/mysql_config \
    --with-pdo-mysql=/usr/local/mysql \
    --with-zlib-dir=/usr/local \
    --with-openssl \
    --with-imap-ssl \
    --disable-debug \
    --with-iconv \
    --enable-fpm

$ make

$ make test

$ make install

$ php -version # 버전 확인
# PHP 7.4.3 나오면 성공
```

- 오류 발생 시 vi 애디터로 파일 열고 수정

```
$ vi /usr/local/apache/bin/apxs
```

```
# 첫줄을 변경
! /usr/bin/perl -w
```

3.3 Apache와 PHP 연동

- vi 애디터 열고 확인 및 추가

```
$ vi /usr/local/apache/conf/httpd.conf
```

```
LoadModule php7_module  modules/libphp7.so  # 확인

AddType application/x-compress .Z
AddType application/x-gzip .gz .tgz
AddType application/x-httpd-php .php .html
```

3.4 PHP 설정 및 테스트

- vi 애디터 열고 수정

```
$ vi /usr/local/src/php-7.4.3/php.ini-production
```

```
short_open_tag = On # php 코드 작성 시 <?php ?> 말고 <? ?> 작성 가능, 하위 호환성을 위함

... 중략

opcache.enable=0    # 캐시를 끄게 되어 개발 시 수정내용이 바로바로 반영되어 개발시 편리
```

- 설정 파일을 잘 관리할 수 있도록 /usr/local/lib 디렉토리에 옮김

```
$ cp /usr/local/src/php-7.4.3/php.ini-production /usr/local/lib/php.in
```

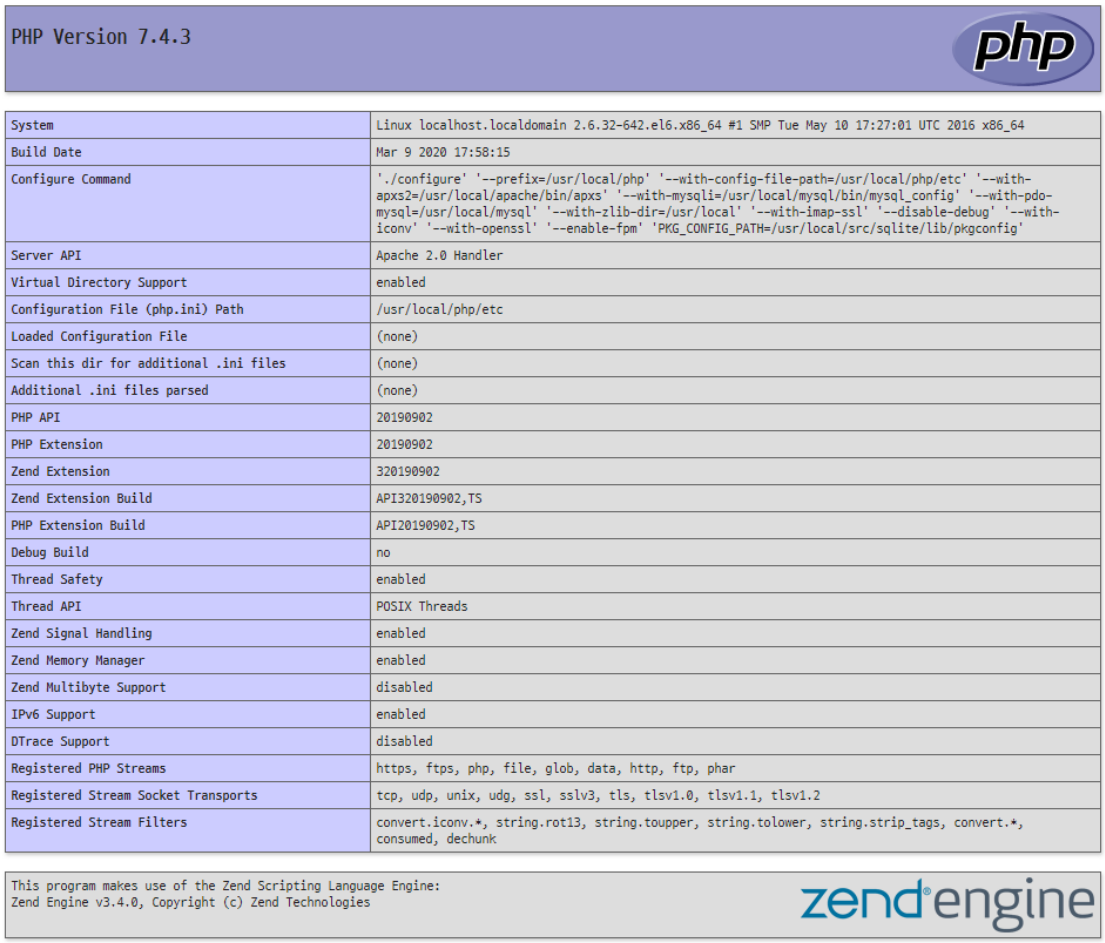
- 테스트

```
$ cd /usr/local/apache/htdocs

$ vi phpinfo.php
```

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

- 아파치 실행

```
$ /usr/local/apache/bin/apachectl stop

$ /usr/local/apache/bin/apachectl start
```



PHP Version 7.4.3

System	Linux localhost.localdomain 2.6.32-642.el6.x86_64 #1 SMP Tue May 10 17:27:01 UTC 2016 x86_64
Build Date	Mar 9 2020 17:58:15
Configure Command	'./configure' '--prefix=/usr/local/php' '--with-config-file-path=/usr/local/php/etc' '--with-apxs2=/usr/local/apache/bin/apxs' '--with-mysqli=/usr/local/mysql/bin/mysql_config' '--with-pdo-mysql=/usr/local/mysql' '--with-zlib-dir=/usr/local' '--with-imap-ssl' '--disable-debug' '--with-iconv' '--with-openssl' '--enable-fpm' 'PKG_CONFIG_PATH=/usr/local/src/sqlite/lib/pkgconfig'
Server API	Apache 2.0 Handler
Virtual Directory Support	enabled
Configuration File (php.ini) Path	/usr/local/php/etc
Loaded Configuration File	(none)
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902,TS
PHP Extension Build	API20190902,TS
Debug Build	no
Thread Safety	enabled
Thread API	POSIX Threads
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

This program makes use of the Zend Scripting Language Engine:
Zend Engine v3.4.0, Copyright (c) Zend Technologies

zend engine

Configuration

apache2handler

4. NGINX 설치

4.1 NGINX 다운로드

```
$ cd /usr/local/src

$ wget https://nginx.org/download/nginx-1.12.0.tar.gz

$ tar xvfz nginx-1.12.0.tar.gz

$ rm -rf nginx-1.12.0.tar.gz
```

4.2 PCRE 다운로드

```
$ cd /usr/local/src/nginx-1.12.0

$ wget http://downloads.sourceforge.net/project/pcre/pcre/8.37/pcre-8.37.tar.gz

$ tar xvfz pcre-8.37.tar.gz
```

4.3 zlib 다운로드

```
$ cd /usr/local/src/nginx-1.12.0

$ wget http://zlib.net/zlib-1.2.11.tar.gz

$ tar xvfz zlib-1.2.11.tar.gz
```

4.4 OpenSSL 다운로드

```
$ cd /usr/local/src/nginx-1.12.0

$ wget http://www.openssl.org/source/openssl-1.0.2f.tar.gz

$ tar xvfz openssl-1.0.2f.tar.gz
```

4.5 NGINX 설치

```
$ ./configure \
    --prefix=/usr/local/src/nginx \
    --with-zlib=./zlib-1.2.11 \
    --with-pcre=./pcre-8.37 \
    --with-openssl=./openssl-1.0.2f \
    --with-http_ssl_module \
    --with-http_stub_status_module

$ make && make install

$ cd /usr/local/src

$ rm -rf nginx-1.12.0
```

4.6 실행권한 설정

```
$ cd /usr/local/src/nginx/sbin

$ sudo chown root nginx

$ sudo chmod +s nginx
```

4.7 실행 및 테스트

```
$ cd /usr/local/src/nginx/sbin

$ ./nginx   # 시작

# 포트 오류시
$ vi /usr/local/src/nginx/conf/nginx.conf   # listen 80 수정

$ ./nginx -s stop   # 종료
```

- 방화벽 설정

  ```
  $ vi /etc/sysconfig/iptables
  ```

```
# -A INPUT -m state --state NEW -m tcp -p tcp --dport 9090 -j ACCEPT
```

4.8 nginx 서비스 등록

- vi 애디터 실행 후 아래 내용 작성

```
$ vi /etc/init.d/nginx
```

```
#!/bin/sh
#
# nginx - this script starts and stops the nginx daemon
#
# chkconfig:   - 85 15
# description:  Nginx is an HTTP(S) server, HTTP(S) reverse \
#               proxy and IMAP/POP3 proxy server
# processname: nginx
# config:      /etc/nginx/nginx.conf
# config:      /etc/sysconfig/nginx
# pidfile:     /var/run/nginx.pid

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ "$NETWORKING" = "no" ] && exit 0

nginx="/usr/local/src/nginx/sbin/nginx"
prog=$(basename $nginx)

NGINX_CONF_FILE="/usr/local/src/nginx/conf/nginx.conf"

[ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx

lockfile=/var/lock/subsys/nginx

make_dirs() {
   # make required directories
   user=`nginx -V 2>&1 | grep "configure arguments:" | sed 's/[^*]*--user=\
([^ ]*\).*/\1/g' -`
   options=`$nginx -V 2>&1 | grep 'configure arguments:'`
   for opt in $options; do
       if [ `echo $opt | grep '.*-temp-path'` ]; then
           value=`echo $opt | cut -d "=" -f 2`
           if [ ! -d "$value" ]; then
               # echo "creating" $value
               mkdir -p $value && chown -R $user $value
           fi
       fi
   done
}

start() {
    [ -x $nginx ] || exit 5
```

```
    [ -f $NGINX_CONF_FILE ] || exit 6
    make_dirs
    echo -n $"Starting $prog: "
    daemon $nginx -c $NGINX_CONF_FILE
    retval=$?
    echo
    [ $retval -eq 0 ] && touch $lockfile
    return $retval
}

stop() {
    echo -n $"Stopping $prog: "
    killproc $prog -QUIT
    retval=$?
    echo
    [ $retval -eq 0 ] && rm -f $lockfile
    return $retval
}

restart() {
    configtest || return $?
    stop
    sleep 1
    start
}

reload() {
    configtest || return $?
    echo -n $"Reloading $prog: "
    killproc $nginx -HUP
    RETVAL=$?
    echo
}

force_reload() {
    restart
}

configtest() {
  $nginx -t -c $NGINX_CONF_FILE
}

rh_status() {
    status $prog
}

rh_status_q() {
    rh_status >/dev/null 2>&1
}

case "$1" in
    start)
        rh_status_q && exit 0
        $1
        ;;
    stop)
        rh_status_q || exit 0
        $1
```

```
        ;;
    restart|configtest)
        $1
        ;;
    reload)
        rh_status_q || exit 7
        $1
        ;;
    force-reload)
        force_reload
        ;;
    status)
        rh_status
        ;;
    condrestart|try-restart)
        rh_status_q || exit 0
            ;;
    *)
        echo $"Usage: $0 {start|stop|status|restart|condrestart|try-
restart|reload|force-reload|configtest}"
        exit 2
esac
```

- 실행권한 설정

```
$ sudo chmod +x /etc/init.d/nginx

$ sudo chkconfig nginx on

$ sudo chkconfig --add nginx

$ sudo chkconfig --list nginx
# 실행 시 다음과 같이 출력되어야 함
# nginx 0:off 1:off 2:on 3:on 4:on 5:on 6:off
```

- 서비스가 등록되면 service 사용 가능

```
$ service nginx (start|stop|restart)
```

4.9 NGINX와 PHP 연동

- php-fpm

```
$ cp /usr/local/src/php-7.4.3/php.ini-production /usr/local/lib/php.ini

$ cp /usr/local/php/etc/php-fpm.conf.default /usr/local/php/etc/php-fpm.conf

$ cp /usr/local/php/etc/php-fpm.d/www.conf.default /usr/local/php/etc/php-
fpm.d/www.conf
```

- chkconfig 설정

```
$ cp /usr/local/src/php-7.4.3/sapi/fpm/init.d/php-fpm /etc/init.d/php-fpm

$ chmod 700 /etc/init.d/php-fpm

$ chkconfig php-fpm on
```

- vi 에디터 실행 후 아래 내용 수정

```
$ vi /usr/local/src/nginx/conf/nginx.conf
```

  - php 부분 주석 제거 후 수정

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

location ~ \.php$ {
    root           /usr/local/src/nginx/html;
    fastcgi_pass   127.0.0.1:9000;
    fastcgi_index  index.php;
    fastcgi_param SCRIPT_FILENAME
/usr/local/src/nginx/html$fastcgi_script_name;
    # /scripts
    include        fastcgi_params;
}
```

- index.php 작성

```
$ vi /usr/local/src/nginx/html/index.php
```

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

- 재시작

```
$ service nginx restart

$ /etc/init.d/php-fpm start
```

PHP Version 7.4.3

System	Linux localhost.localdomain 2.6.32-642.el6.x86_64 #1 SMP Tue May 10 17:27:01 UTC 2016 x86_64
Build Date	Mar 9 2020 17:58:15
Configure Command	'./configure' '--prefix=/usr/local/php' '--with-config-file-path=/usr/local/php/etc' '--with-apxs2=/usr/local/apache/bin/apxs' '--with-mysqli=/usr/local/mysql/bin/mysql_config' '--with-pdo-mysql=/usr/local/mysql' '--with-zlib-dir=/usr/local' '--with-imap-ssl' '--disable-debug' '--with-iconv' '--with-openssl' '--enable-fpm' 'PKG_CONFIG_PATH=/usr/local/src/sqlite/lib/pkgconfig'
Server API	FPM/FastCGI
Virtual Directory Support	enabled
Configuration File (php.ini) Path	/usr/local/php/etc
Loaded Configuration File	(none)
Scan this dir for additional .ini files	(none)
Additional .ini files parsed	(none)
PHP API	20190902
PHP Extension	20190902
Zend Extension	320190902
Zend Extension Build	API320190902,TS
PHP Extension Build	API20190902,TS
Debug Build	no
Thread Safety	enabled
Thread API	POSIX Threads
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	disabled
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls, tlsv1.0, tlsv1.1, tlsv1.2
Registered Stream Filters	convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

Configuration

cgi-fcgi