

Ubuntu에 WordPress 설치하기: 전체 명령어 포함 실무 매뉴얼

서버 준비부터 웹사이트 실행까지, 단계별 완벽 가이드



설치 프로세스: 5단계 로드맵

-  **1** **사전 준비 (Prerequisites)**
시스템 환경 및 버전 확인
-  **2** **데이터베이스 설정 (Database Setup)**
전용 DB 및 사용자 생성
-  **3** **워드프레스 설치 (WordPress Install)**
파일 다운로드 및 배치
-  **4** **서버 연동 설정 (Server Configuration)**
Apache 가상 호스트 및 DNS 설정
-  **5** **설치 완료 및 실행 (Finalization)**
서비스 재시작 및 워드프레스 초기 설정

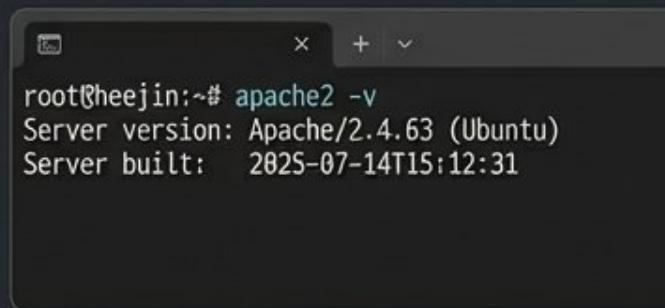
1단계: 필수 구성요소 버전 확인

WordPress 설치에 필요한 최소 사양은 Apache 2.4, PHP 8.3, MariaDB 10.6 이상입니다.
아래 명령어로 현재 서버의 버전을 확인하여 요구사항을 충족하는지 검증합니다.

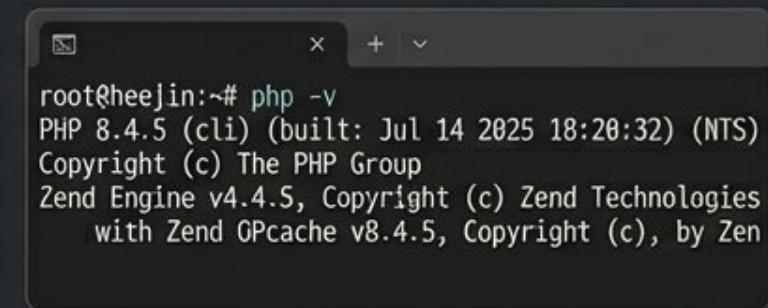
```
# Apache 버전 확인  
apache2 -v
```

```
# PHP 버전 확인  
php -v
```

```
# MariaDB 버전 확인  
mariadb --version
```



```
root@heejin:~# apache2 -v  
Server version: Apache/2.4.63 (Ubuntu)  
Server built: 2025-07-14T15:12:31
```



```
root@heejin:~# php -v  
PHP 8.4.5 (cli) (built: Jul 14 2025 18:20:32) (NTS)  
Copyright (c) The PHP Group  
Zend Engine v4.4.5, Copyright (c) Zend Technologies  
with Zend OPcache v8.4.5, Copyright (c), by Zen
```

1단계: 워드프레스 전용 홈 디렉터리 및 사용자 생성

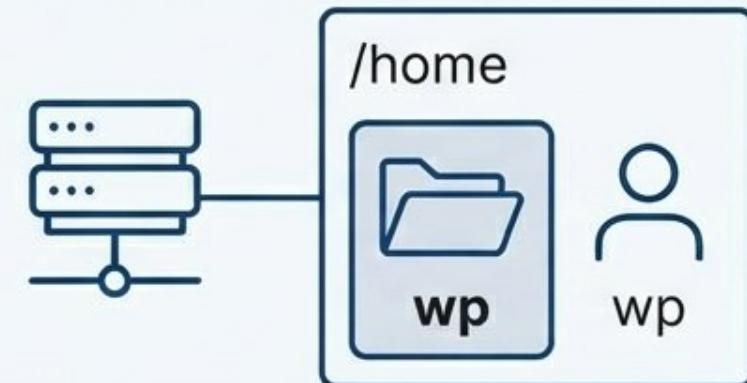
웹사이트 파일을 안전하게 격리하고 관리하기 위해 별도의 사용자 계정과 홈 디렉터리를 생성합니다. 이는 서버 보안의 기본적인 조치입니다.

```
# 워드프레스 파일이 위치할 전용 홈 디렉터리 생성
```

```
mkdir -p /home/wp
```

```
# 'wp' 사용자 생성
```

```
useradd wp
```



명령어 정리 (1/6)

1단계에서 확인/생성한 명령어

```
# 필수 구성요소 버전 확인
apache2 -v
php -v
mariadb --version
```

```
# 워드프레스 전용 홈 디렉터리/사용자 생성
mkdir -p /home/wp
useradd wp
```

2단계: 전용 데이터베이스 및 사용자 생성 (1/2)

워드프레스 데이터를 저장할 DB와 접근 권한을 가진 사용자를 생성합니다. 먼저 root 계정으로 DB 서버에 접속하여 WordPress 애플리케이션이 사용할 'bs' 사용자를 생성하고 초기 권한을 부여합니다.



```
# DB 서버(172.16.254.20)에 root로 접속
mysql -h 172.16.254.20 -uroot -p --ssl=0

# 'bs' 사용자를 위한 초기 데이터베이스 생성
CREATE DATABASE bs;

# 'bs' 사용자 생성 및 암호 설정
# '172.16.8.20'은 워드프레스가 설치된 웹서버의 IP입니다.
CREATE USER 'bs'@'172.16.8.20' IDENTIFIED BY 'asd123!@';

# 'bs' 사용자에게 'bs' DB에 대한 모든 권한 부여
GRANT ALL PRIVILEGES ON bs.* TO 'bs'@'172.16.8.20';

# 변경사항 적용
FLUSH PRIVILEGES;

# 접속 종료
QUIT;
```

2단계: 워드프레스 전용 DB 생성 (2/2)

앞서 생성한 'bs' 사용자로 다시 DB 서버에 접속하여, 실제 워드프레스가 사용할 'bs_wp' 데이터베이스를 생성하고 해당 데이터베이스에 대한 모든 권한을 'bs' 사용자에게 부여합니다.



```
# 생성된 'bs' 사용자로 DB 서버에 접속
mysql -h 172.16.254.20 -u bs -p --ssl=0

# 워드프레스 전용 'bs_wp' 데이터베이스 생성
CREATE DATABASE bs_wp;

# 'bs' 사용자에게 'bs_wp' DB에 대한 모든 권한 부여
GRANT ALL PRIVILEGES ON bs_wp.* TO 'bs'@'172.16.8.20';

# 변경사항 적용
FLUSH PRIVILEGES;

# 접속 종료
QUIT;
```

명령어 정리 (2/6)

2단계 데이터베이스 설정

```
# DB 서버 접속 (root) 및 bs 사용자/DB 생성
mysql -h 172.16.254.20 -uroot -p --ssl=0
CREATE DATABASE bs;
CREATE USER 'bs'@'172.16.8.20' IDENTIFIED BY 'asd123!@';
GRANT ALL PRIVILEGES ON bs.* TO 'bs'@'172.16.8.20';
FLUSH PRIVILEGES;
QUIT;

# DB 서버 재접속 (bs) 및 워드프레스용 DB 생성
mysql -h 172.16.254.20 -ubs -p --ssl=0
CREATE DATABASE bs_wp;
GRANT ALL PRIVILEGES ON bs_wp.* TO 'bs'@'172.16.8.20';
FLUSH PRIVILEGES;
QUIT;
```

3단계: 워드프레스 코어 파일 다운로드 및 배치 (1/2)

공식 사이트에서 최신 한국어 버전의 워드프레스 압축 파일을 서버로 직접 다운로드하고 압축을 해제합니다.
'unzip' 패키지가 설치되어 있어야 합니다.



```
# 공식 배포판 다운로드
wget https://ko.wordpress.org/latest-ko_KR.zip

# 압축 해제
unzip latest-ko_KR.zip
```

```
:~$ unzip latest-ko_KR.zip
files ls
latest-ko_KR.zip  wordpress
wordpress:
index.php          wp-includes
license.txt        wp-links-opml.php
readme.html        wp-load.php
wp-activate.php   wp-login.php
wp-admin          wp-mail.php
wp-blog-header.php wp-mail.php
wp-comments-post.php wp-settings.php
wp-config-sample.php wp-signup.php
wp-content         wp-trackback.php
wp-cron.php        xmlrpc.php
wp-includes
```

3단계: 파일 이동 및 소유권/권한 설정 (2/2)

압축 해제 후 생성된 wordpress 폴더의 모든 파일을 이전에 생성한 /home/wp 디렉터리로 옮깁니다. 그 후 웹 서버가 정상적으로 파일을 처리할 수 있도록 소유권과 접근 권한을 설정합니다.

```
# 압축 해제 후 생성된 wordpress 디렉터리로 이동
cd ~/wordpress

# 모든 파일을 /home/wp 디렉터리로 이동
mv * /home/wp

# 상위 디렉터리(/home)로 이동하여 권한 작업 수행
cd /home

# /home/wp 디렉터리 및 하위 요소에 권한 부여 (rwxr-xr-x)
chmod 755 -R wp

# /home/wp 디렉터리 및 하위 요소의 소유자를 'wp' 사용자 및 그룹으로 변경
chown -R wp:wp wp
```



명령어 정리 (3/6)

3단계 파일 배치

```
# 워드프레스 코어 다운로드/압축 해제
wget https://ko.wordpress.org/latest-ko_KR.zip
unzip latest-ko_KR.zip

# 파일 이동 및 소유권/권한 설정
cd ~/wordpress
mv * /home/wp
cd /home
chmod 755 -R wp
chown -R wp:wp wp
```

4단계: 데이터베이스 연결 설정 (1/2) - wp-config.php 생성

WordPress가 데이터베이스에 접속할 수 있도록 설정 파일을 준비합니다.

제공된 샘플 설정 파일(wp-config-sample.php)의 이름을 wp-config.php로 변경하여 원본은 보존하고 사본으로 작업합니다.

```
# 워드프레스 홈 디렉터리로 이동  
cd /home/wp  
  
# 샘플 설정 파일을 실제 설정 파일로 복사 (이름 변경)  
mv wp-config-sample.php wp-config.php
```



4단계: `wp-config.php` 파일에 DB 정보 입력 (2/2)

`vim`이나 선호하는 편집기로 `wp-config.php` 파일을 열고, 2단계에서 생성한 데이터베이스 이름, 사용자, 비밀번호, 호스트(DB 서버 IP) 정보를 정확하게 입력합니다.

```
vim /home/wp/wp-config.php
```

수정 전

```
<?php
define( 'DB_NAME', 'database_name_here' );
define( 'DB_USER', 'username_here' );
define( 'DB_PASSWORD', 'password_here' );
define( 'DB_HOST', 'localhost' );
```

수정 후

```
<?php
define( 'DB_NAME', 'bs_wp' );
define( 'DB_USER', 'bs' );
define( 'DB_PASSWORD', 'asd123!@' );
define( 'DB_HOST', '172.16.254.20' );
```

명령어 정리 (4/6)

4단계 DB 연결 설정

```
# wp-config.php 생성
cd /home/wp
mv wp-config-sample.php wp-config.php

# DB 정보 입력 (편집)
vim /home/wp/wp-config.php

define('DB_NAME', 'bs_wp');
define('DB_USER', 'bs');
define('DB_PASSWORD', 'asd123!@');
define('DB_HOST', '172.16.254.20');
```

5단계: 웹 서버(Apache) 가상 호스트 설정

wp.bs.com 도메인으로 접속 시 `/home/wp`` 디렉터리의 파일을 서비스하도록 Apache 가상 호스트 설정을 추가합니다. 이 설정은 주로 `/etc/apache2/sites-available/` 경로의 설정 파일에 추가합니다.

설정 파일 경로: `vim /etc/apache2/sites-available/vhost.conf`

```
<VirtualHost *:80>
  ServerName wp.bs.com
  DocumentRoot "/home/wp"

  <Directory "/home/wp/">
    AllowOverride None
    Options Indexes FollowSymLinks
    Require all granted
  </Directory>

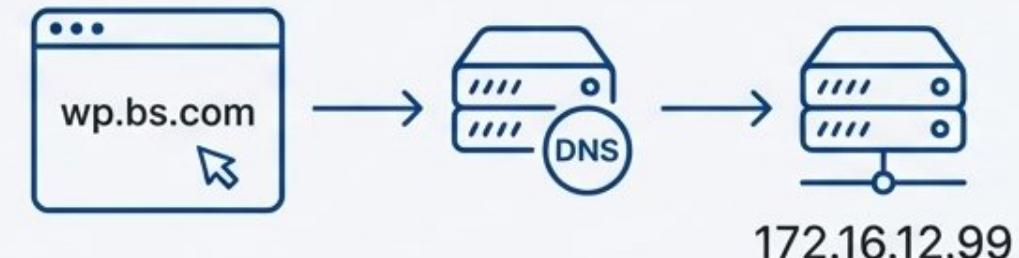
  ErrorLog ${APACHE_LOG_DIR}/error.log
  CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

6단계: DNS 서버에 도메인 레코드 추가

‘wp.bs.com’ 도메인 요청이 워드프레스가 설치된 웹 서버의 IP(‘172.16.12.99’)로 해석되도록, 관할 DNS 서버의 Zone 파일에 A 레코드를 추가합니다.

설정 파일 경로: vim /etc/bind/db.heejin.com

```
wp    IN    A    172.16.12.99
```



명령어 정리 (5/6)

5~6단계 웹서버/DNS 설정

```
# Apache 가상 호스트 설정 파일 편집
vim /etc/apache2/sites-available/vhost.conf

<VirtualHost *:80>
    ServerName wp.bs.com
    DocumentRoot /home/wp
    <Directory "/home/wp/">
        AllowOverride None
        Options Indexes FollowSymLinks
        Require all granted
    </Directory>
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>

# DNS Zone 파일에 A 레코드 추가
vim /etc/bind/db.heejin.com
wp    IN    A    172.16.12.99
```

7단계: 서비스 재시작 및 웹 설치 화면 접속

지금까지 변경한 DNS와 Apache 설정을 시스템에 적용하기 위해 **관련 서비스를 재시작합니다.** 재시작이 완료되면 웹 브라우저에서 설정한 도메인으로 접속하여 설치를 마무리할 수 있습니다.

```
# Bind9 DNS 서버 재시작  
systemctl restart bind9
```

```
# Apache2 웹 서버 재시작  
systemctl restart apache2
```

안내: 이제 웹 브라우저 주소창에  <http://wp.bs.com> → 을 입력하세요.

설치 완료! 웹에서 마지막 설정을 진행하세요.

서버 설정이 모두 성공적으로 완료되었습니다. 브라우저에 나타난 '환영합니다' 화면에서 사이트 제목, 관리자 계정 정보 등을 입력하면 모든 설치 과정이 끝납니다.



환영합니다

유명한 워드프레스는 물론 범위 넓은 웹사이트를 운영합니다! 이에의 정보를 입력하기만 하면 세계에서 가장 확장성 있고 강력한 개인 웹사이트 플랫폼을 사용할 수 있습니다.

정보가 필요합니다

다음 정보를 제공해주세요. 걱정하지 마세요. 이 설정을 나중에 언제든지 바꿀 수 있습니다.

사이트 제목

사용자명

비밀번호 oTnCK*180@hh6@ahdn

이메일 주소 계속하기 전에 이메일 주소를 다시 확인하세요.

김박 얀친 가짜성 김박 얀친이 이 사이트를 운영하는 것을 확인

이 요청이 받아들이치는 것은 전적으로 김박 얀친의 죄입니다.

아드민데스 설치

명령어 정리 (6/6)

마무리

```
# 변경사항 적용을 위한 서비스 재시작
systemctl restart bind9
systemctl restart apache2

# 브라우저 접속
http://wp.bs.com
```