

Chapter 13 DB 서버와 웹 서버

목차

01 데이터베이스

02 MariaDB 설치와 사용

03 웹 서버 설치와 사용

학습목표



- 데이터베이스가 무엇인지 이해하고 설명할 수 있다.
- MariaDB를 설치하고 데이터베이스를 구축하여 사용할 수 있다.
- 아파치 웹 서버를 설치하고 웹 페이지를 구축하여 사용할 수 있다.
- APM이 연동되도록 설치할 수 있다.
- 공개 게시판을 설치하고 웹 사이트에 연결할 수 있다.

00 Preview

00 Preview

■ 13장의 내용 구성

- MariaDB를 이용하여 데이터베이스를 구축하는 방법을 알아본다
- 아파치 웹 서버를 설치하고, 시스템과 사용자별로 웹 사이트를 구축하는 방법을 알아본다
- APM을 연동하고, 이를 활용한 대표적 웹 요소인 게시판을 설치하고 사용해 본다



01 데이터베이스

01 데이터베이스

■ 데이터베이스

- DB는 데이터를 모아놓은 것을 의미. 서로 관련성 있는 데이터들을 데이터 간의 중복성을 최소화해서 체계적으로 모아놓은 것

■ 관계형 데이터베이스의 기본 개념

- 관계형 데이터베이스는 데이터를 테이블(표)로 표현
- 관계형 데이터베이스는 데이터를 항목으로 구분하여 테이블로 구성.
이렇게 구성하면 테이블의 생성이나 항목의 분할과 결합을 자유롭게 할 수 있어서 편리

01 데이터베이스

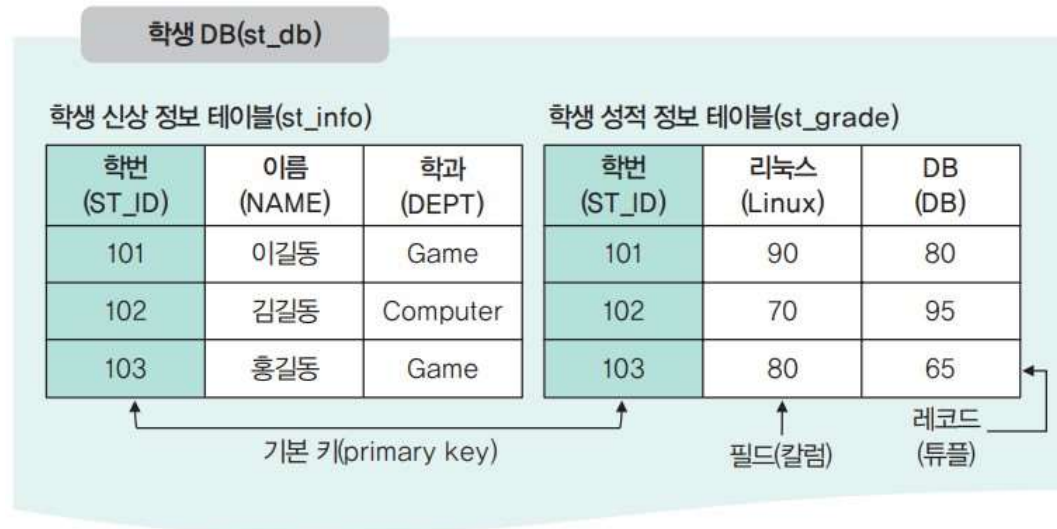
■ 관계형 데이터베이스 관련 용어

- 데이터: 각 항목에 저장되는 값
- 테이블: 데이터를 체계화하여 행과 열의 형태로 구성한 것으로 테이블 이름을 붙임
- 데이터베이스: 관련 있는 데이터를 체계적으로 저장한 것으로, 데이터는 테이블 형태로 저장. 데이터베이스에는 하나 이상의 테이블이 있을 수 있음. 데이터베이스 이름을 붙임
- 필드: 테이블의 열을 의미하며 칼럼이라고도 함
- 레코드: 테이블에 저장한 한 행의 정보를 의미. 튜플이라고도 함. 한 행에는 여러 필드의 값이 저장될 수 있음
- 키: 레코드를 구성하는 필드에서 각 레코드를 구분할 수 있는 필드 값. 키로 선택된 필드의 값은 중복될 수 없음. 키는 기본 키와 그 외 다른 키가 있을 수 있음

01 데이터베이스

■ 관계형 데이터베이스의 예

- 학생 데이터베이스를 예로 데이터베이스와 테이블의 개념을 살펴보자.
학생과 관련된 데이터에는 학생의 신상 데이터나 성적 데이터, 장학 데이터 등 여러 가지가 있음
- 일반적으로 데이터베이스를 구축할 때는 하나의 테이블에 모든 데이터 필드를 만들지 않고 적절히 여러 개의 테이블로 분산함



- 여기서는 [그림 13-1]과 같이 학생 신상 데이터와 성적 데이터로 두 개의 테이블을 만들었음

그림 13-1 학생 데이터베이스의 예

01 데이터베이스

■ 관계형 데이터베이스의 예

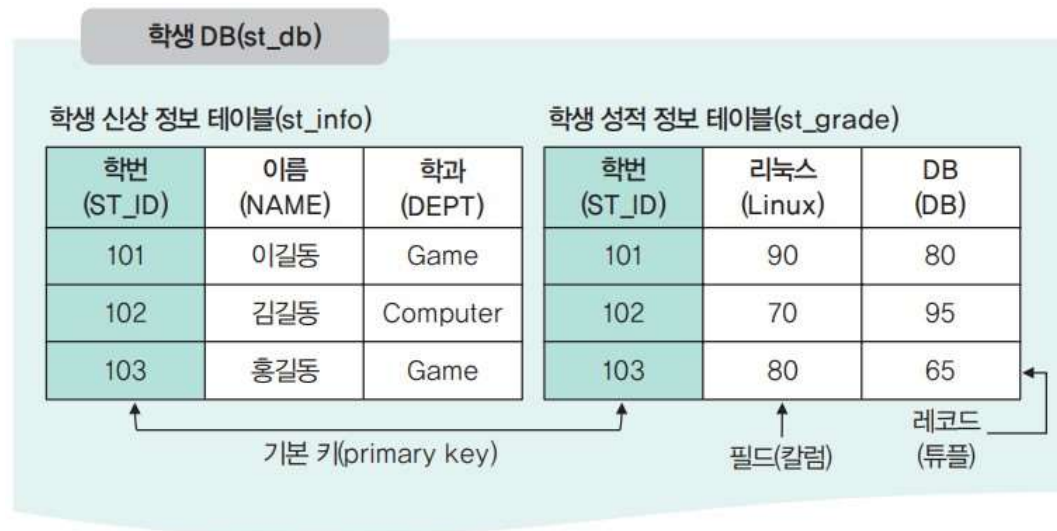


그림 13-1 학생 데이터베이스의 예

- [그림 13-1]을 보면 데이터베이스의 이름이 '학생 DB'이고 이 데이터베이스에 학생 신상 정보와 학생 성적 테이블이 있음
- 학생 신상 정보 테이블은 학번, 이름, 학과라는 세 필드로 구성되어 있고, 학생 성적 테이블은 학번, 리눅스, DB라는 세 필드로 구성되어 있음

01 데이터베이스

■ 관계형 데이터베이스의 예

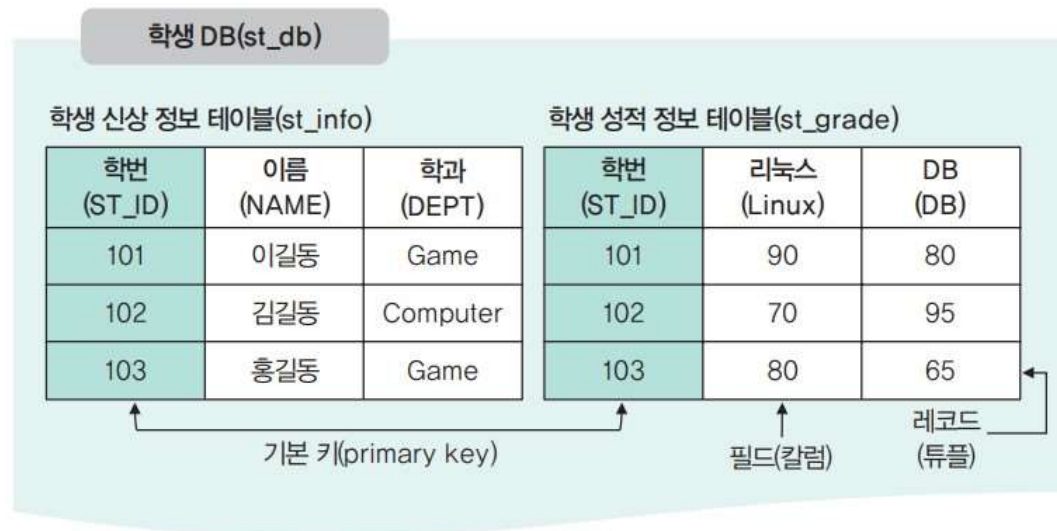


그림 13-1 학생 데이터베이스의 예

- 각 테이블에는 레코드가 세 개씩 저장되어 있음. 또한 각 테이블은 학번을 기본 키로 사용하고 있음
- 기본 키는 테이블에서 각 레코드를 구분할 때도 사용하고 테이블과 테이블의 데이터를 연결할 때도 사용함

01 데이터베이스

■ SQL 기초

- SQL: 관계형 데이터베이스를 생성하고, 테이블을 만들고, 데이터를 입력·삭제·수정할 때 사용하는 언어. SQL을 알아야 관계형 데이터베이스를 사용 할 수 있음

■ 데이터베이스 관련 SQL

- 먼저 데이터베이스의 생성, 삭제, 조회, 사용과 관련된 SQL을 살펴보자
- SQL의 각 문장은 C 언어처럼 ;으로 끝나야 함
- SQL은 특별히 지정하지 않으면 기본적으로 대문자·소문자를 구별하지 않음

01 데이터베이스

■ 데이터베이스 관련 SQL

- ① 데이터베이스 목록 확인: 기존 데이터베이스 목록을 출력

```
show databases;
```

- ② 데이터베이스 생성: 새로운 데이터베이스를 생성

```
create database 데이터베이스명;
```

예를 들어 다음과 같이 데이터베이스를 생성할 수 있음

```
create database st_db;
```

데이터베이스를 새로 생성하면 이름만 있을 뿐 내용이 아무것도 없음

01 데이터베이스

■ 데이터베이스 관련 SQL

- ③ 데이터베이스 삭제: 지정한 데이터베이스를 삭제

```
drop database 데이터베이스명;
```

데이터베이스를 삭제하면 해당 데이터베이스에 있던 테이블들도 전부 삭제됨

- ④ 사용할 데이터베이스 지정: 여러 데이터베이스 중에서 작업할 데이터베이스를 선택

```
use 데이터베이스명;
```

01 데이터베이스

■ 테이블 관련 SQL

- ① 데이터베이스 목록 확인: 기존 데이터베이스 목록을 출력

```
show databases;
```

- ② 테이블 생성: 새로운 테이블을 생성.

새로운 테이블을 생성하려면 사용할 수 있는 자료형을 알아야 함

표 13-1 테이블 필드의 자료형

자료형	설명	예
varchar(n)	최대 n개의 크기를 가진 가변 문자열	varchar(10)
char(n)	n개의 크기를 가진 고정 문자열	char(5)
int	정수형 숫자	10
float	실수형 숫자	10.1
date	날짜	2020-01-26
time	시간	02:56:30

01 데이터베이스

■ 테이블 관련 SQL

- ② 테이블을 생성할 때는 테이블의 구조를 정의해야 함
테이블명 다음에 괄호를 열고 필드명과 필드명의 자료형을 지정함

```
create table 테이블명(필드명1 필드명1의 자료형, 필드명2 필드명2의 자료형, ...);
```

예를 들어 다음과 같이 테이블을 생성할 수 있음

```
create table st_info (ST_ID int, NAME varchar(20), DEPT varchar(25));
```

- ③ 테이블 구조 확인: 지정한 테이블의 구조를 출력

```
explain 테이블명;
```

- ④ 테이블 수정: 테이블의 구조를 변경

```
alter table 테이블명 수정 명령;
```


01 데이터베이스

■ 테이블 관련 SQL

④ 수정할 내용에 따라 사용 예를 살펴보자

- 테이블에 필드 추가: alter table 테이블명 add 필드명 자료형

예) alter table st_info add AGE int;

- 필드의 자료형 변경: alter table 테이블명 modify 필드명 자료형

예) alter table st_info modify AGE float;

- 필드 삭제: alter table 테이블명 drop column 필드명

예) alter table st_info drop column AGE;

- 기본 키 추가: alter table 테이블명 add constraint 제약 조건명 primary key (필드명)

예) alter table st_info add constraint pk_stinfo primary key (ST_ID);

제약 조건명은 사용자가 정하면 된다. 처음 테이블을 생성하면서 기본 키를 지정할 때 다음과 같이 작성한다.

예) create table st_info (ST_ID int Not NULL primary key, NAME varchar(20), DEPT varchar(25));

01 데이터베이스

■ 테이블 관련 SQL

- ⑤ 테이블 삭제: 지정한 테이블을 삭제

```
drop table 테이블명;
```

01 데이터베이스

■ 레코드 삽입·삭제·수정 관련 SQL

- 테이블에 데이터를 입력하거나 삭제할 때는 레코드 단위로 작업
- 레코드를 삽입하거나 삭제 및 수정하려면 다음 SQL 문을 사용해야 함

① 레코드 삽입: 테이블에 새로운 레코드를 추가

```
insert into 테이블명 values (값1, 값2, ...);
```

삽입하는 레코드는 테이블을 생성할 때 지정한 자료형에 맞는 데이터를 입력해야 함
다음 예와 같이 레코드를 삽입

```
insert into st_info values (101, '이길동', 'Game');
```

② 레코드 수정: 기존 레코드를 수정

```
update 테이블명 set 필드명1= 수정할 값1, 필드명2= 수정할 값2, ... where 조건;
```

01 데이터베이스

■ 레코드 삽입·삭제·수정 관련 SQL

- ② 조건에 맞는 레코드를 찾아 지정한 필드의 값을 수정

다음은 학생 신상 정보 테이블 (st_info)에서 학번이 103인 레코드의 학과(DEPT)를 Computer로 수정한 것

```
update st_info set DEPT='Computer' where ST_ID=103;
```

- ③ 레코드 삭제: 테이블에서 해당 레코드를 삭제

```
delete from 테이블명 where 조건;
```

예를 들어 다음과 같이 삭제

```
delete from st_info where ST_ID=103;
```

01 데이터베이스

■ 레코드 검색 관련 SQL

- 테이블에 저장된 레코드를 검색할 때는 다음과 같이 select 문을 사용

```
select 필드명1, 필드명2, ... from 테이블명 where 조건;
```

- 예를 들어 다음과 같이 작성

```
① select * from st_info;
```

```
② select NAME from st_info where ST_ID=101;
```

```
③ select Linux, DB from st_grade where ST_ID=101;
```

①은 학생 신상 정보 테이블(st_info)의 모든 레코드를 검색하는 것

②는 학생 신상 정보 테이블에서 학번(ST_ID)이 101인 학생의 이름(NAME)을 검색하는 것

③은 학생 성적 테이블(st_grade)에서 학번이 101인 학생의 리눅스와 DB 성적을 검색하는 것

01 데이터베이스

■ 레코드 검색 관련 SQL

- 만약 학번이 101인 학생의 이름과 리눅스 성적을 알고 싶다면 어떻게 검색할까?
이럴 때는 양쪽 테이블에 공통으로 적용되는 기본 키를 이용하여 검색할 수 있음
- 학생의 이름과 리눅스 성적을 조회하는데, 이렇게 테이블을 연결하는 것을 조인이라고 함

```
select st_info.NAME, st_grade.Linux  
from st_info, st_grade  
where st_info.ST_ID=101 and st_grade.ST_ID=101;
```

01 데이터베이스

■ 접근 권한 관련 SQL

- 시스템 관리자가 데이터베이스를 만들고 일반 사용자들이 사용하도록 설정해야 할 때 grant문을 사용
- 예를 들어 그누보드 같은 게시판을 설치하려면 데이터베이스가 필요
이때 관리자가 데이터베이스를 만들고, 이 데이터베이스에 대한 모든 권한을 해당 사용자에게 부여하는 것

```
grant all privileges on DB명.* to 사용자 계정@localhost identified by '패스워드';
```

- 앞에서 생성한 학생 DB(st_db)에 대한 사용권을 user1 사용자에게 부여하려면 다음과 같이 하면 됨

```
grant all privileges on st_db.* to user1@localhost identified by '123456';
```

02 MariaDB 설치와 사용

02 MariaDB 설치와 사용

■ MariaDB 설치 및 사용

- MariaDB 서버는 dnf 명령으로 설치할 수 있음. MariaDB 서버의 패키지 이름은 mariadbserver

```
[root@localhost ~]# dnf install mariadb-server
```

마지막 메타자료 만료확인(1:16:59 이전): 2023년 10월 03일 (화) 오후 01시 58분 33초.

종속성이 해결되었습니다.

(생략)

설치되었습니다:

mariadb-3:10.5.16-2.el9_0.x86_64	mariadb-backup-3:10.5.16-2.el9_0.x86_64
mariadb-common-3:10.5.16-2.el9_0.x86_64	mariadb-connector-c-3.2.6-1.el9_0.x86_64
mariadb-connector-c-config-3.2.6-1.el9_0.noarch	mariadb-errmsg-3:10.5.16-2.el9_0.x86_64
mariadb-gssapi-server-3:10.5.16-2.el9_0.x86_64	mariadb-server-3:10.5.16-2.el9_0.x86_64
mariadb-server-utils-3:10.5.16-2.el9_0.x86_64	mysql-selinux-1.0.5-1.el9_0.noarch
perl-DBD-MariaDB-1.21-16.el9_0.x86_64	perl-DBI-1.643-9.el9.x86_64
perl-File-Copy-2.34-480.el9.noarch	perl-Math-BigInt-1:1.9998.18-460.el9.noarch
perl-Math-Complex-1.59-480.el9.noarch	perl-Sys-Hostname-1.23-480.el9.x86_64

완료되었습니다!

02 MariaDB 설치와 사용

■ MariaDB 활성화하기

- mariadb.service를 활성화하면 MariaDB 서버가 활성화됨
- systemctl 명령으로 mariadb.service를 활성화하고 ps 명령으로 확인해 보면 mariadb 데몬의 소유자 이름이 mysql로 나오는데, 이는 이전에 MySQL로 구현된 프로그램들과의 호환성을 위한 것

```
[root@localhost ~]# systemctl start mariadb.service
[root@localhost ~]# ps -ef | grep mariadb
mysql    41898      1    0 15:18 ?        00:00:00 /usr/libexec/mariabdb --basedir=/usr
root     41945   40419    0 15:18 pts/0    00:00:00 grep --color=auto mariadb
```

- MariaDB 서버를 동작시켰으니 MariaDB에 접속하기만 하면 됨
- MariaDB를 사용하기 위한 명령은 mariadb이나 mysql로 입력해도 똑같이 동작
- 지금은 설정된 계정과 암호가 없으므로 그냥 사용

02 MariaDB 설치와 사용

■ MariaDB 활성화하기

- MariaDB의 기본 프롬프트는 MariaDB [(none)]>
- MariaDB에서 종료하려면 exit를 입력함

```
[root@localhost ~]# mysql
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 4
Server version: 10.5.16-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> exit
Bye
[root@localhost ~]# mariadb
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 5
Server version: 10.5.16-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

02 MariaDB 설치와 사용

■ 데이터베이스 목록 확인하기

- 데이터베이스 목록을 확인하는 명령은 `show databases`
- MariaDB에는 기본적으로 `information_schema` 등 세 개의 데이터베이스가 있음

```
MariaDB [(none)]> show databases;
+-----+
| Database          |
+-----+
| information_schema |
| mysql             |
| performance_schema |
+-----+
3 rows in set (0.001 sec)
```

02 MariaDB 설치와 사용

■ 데이터베이스 생성 및 사용하기

- SQL 명령이 오류 없이 실행되면 'Query OK' 메시지가 출력됨
- st_db를 생성하고 show databases 명령으로 확인하면 데이터베이스가 생성되었음을 알 수 있음

```
MariaDB [(none)]> create database st_db;
```

```
Query OK, 1 row affected (0.000 sec)
```

```
MariaDB [(none)]> show databases;
```

```
+-----+
```

```
| Database |
```

```
+-----+
```

```
| information_schema |
```

```
| mysql |
```

```
| performance_schema |
```

```
| st_db |
```

```
+-----+
```

```
4 rows in set (0.001 sec)
```

02 MariaDB 설치와 사용

■ 데이터베이스 생성 및 사용하기

- 데이터베이스에 테이블 만드는 작업을 하려면 사용할 데이터베이스를 use 명령으로 지정해야 함
- 앞에서 생성한 st_db를 사용하도록 명령을 실행해 보자
- 프롬프트의 (none)이 st_db로 바뀌었는데, 이는 현재 사용하고 있는 데이터베이스가 st_db임을 알려줌

```
MariaDB [(none)]> use st_db;  
Database changed  
MariaDB [st_db]>
```

02 MariaDB 설치와 사용

■ 테이블 생성 및 구조 확인하기

- 테이블이 있는지 확인해 보면 아무것도 없음을 알 수 있음

```
MariaDB [st_db]> show tables;  
Empty set (0.000 sec)
```

- [그림 13-1]의 학생 신상 정보 테이블(st_info)을 생성해 보자
- 테이블을 생성할 때 기본 문자 집합을 utf8로 설정해야 한글이 깨지지 않음
- 필드 정보 다음에 'default charset=utf8'을 추가하면 기본 문자 집합이 설정됨

```
MariaDB [st_db]> create table st_info (ST_ID int, NAME varchar(20), DEPT varchar(25))  
default charset=utf8;  
Query OK, 0 rows affected (0.004 sec)
```

02 MariaDB 설치와 사용

■ 테이블 생성 및 구조 확인하기

- 학생 성적 테이블(st_grade)을 생성해 보자. 이 테이블에는 한글 데이터를 입력하지 않을 것이므로 문자 집합을 설정하지 않아도 됨. 학번과 성적으로 이루어지므로 자료형을 모두 int로 지정

```
MariaDB [st_db]> create table st_grade (ST_ID int, Linux int, DB int);  
Query OK, 0 rows affected (0.004 sec)
```

- show tables 명령을 사용하여 테이블이 정상적으로 생성되었는지 확인해 보자. st_db에 테이블 두 개가 생성되었음을 알 수 있음

```
MariaDB [st_db]> show tables;  
+-----+  
| Tables_in_st_db |  
+-----+  
| st_grade         |  
| st_info          |  
+-----+  
2 rows in set (0.000 sec)
```


02 MariaDB 설치와 사용

■ 테이블 생성 및 구조 확인하기

```
MariaDB [st_db]> explain st_info;
```

Field	Type	Null	Key	Default	Extra
ST_ID	int(11)	YES		NULL	
NAME	varchar(20)	YES		NULL	
DEPT	varchar(25)	YES		NULL	

3 rows in set (0.002 sec)

```
MariaDB [st_db]> explain st_grade;
```

Field	Type	Null	Key	Default	Extra
ST_ID	int(11)	YES		NULL	
Linux	int(11)	YES		NULL	
DB	int(11)	YES		NULL	

3 rows in set (0.001 sec)

- 테이블의 구조를 확인해 보자
- 'explain 테이블명'을 사용하면 구조를 확인할 수 있음

02 MariaDB 설치와 사용

■ 기본 키 추가하기

- ST_ID 필드를 기본 키로 설정해 보자. Null은 값이 없다는 뜻이므로 기본 키 값에 Null을 허용하면 이상함. 먼저 Null 값을 허용하지 않도록 수정함

```
MariaDB [st_db]> alter table st_info modify ST_ID int Not Null;
```

```
Query OK, 0 rows affected (0.006 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [st_db]> alter table st_grade modify ST_ID int Not Null;
```

```
Query OK, 0 rows affected (0.008 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

02 MariaDB 설치와 사용

■ 기본 키 추가하기

- ST_ID 필드를 기본 키로 설정

```
MariaDB [st_db]> alter table st_info add constraint pk_stinfo primary key(ST_ID);  
Query OK, 0 rows affected, 1 warning (0.008 sec)  
Records: 0 Duplicates: 0 Warnings: 1
```

```
MariaDB [st_db]> alter table st_grade add constraint pk_stgrade primary key(ST_ID);  
Query OK, 0 rows affected, 1 warning (0.007 sec)  
Records: 0 Duplicates: 0 Warnings: 1
```

02 MariaDB 설치와 사용

■ 기본 키 추가하기

- 제대로 설정되었는지 explain 명령으로 확인해 보자
- explain 명령의 결과를 보면 두 테이블 모두 ST_ID 필드의 Null을 허용하지 않고 기본 키로 설정되었음

```
MariaDB [st_db]> explain st_info;
```

Field	Type	Null	Key	Default	Extra
ST_ID	int(11)	NO	PRI	NULL	
NAME	varchar(20)	YES		NULL	
DEPT	varchar(25)	YES		NULL	

```
3 rows in set (0.001 sec)
```

```
MariaDB [st_db]> explain st_grade;
```

Field	Type	Null	Key	Default	Extra
ST_ID	int(11)	NO	PRI	NULL	
Linux	int(11)	YES		NULL	
DB	int(11)	YES		NULL	

```
3 rows in set (0.001 sec)
```

02 MariaDB 설치와 사용

■ 레코드 입력하기

```
MariaDB [st_db]> insert into st_info values (101, '이길동', 'Game');  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [st_db]> insert into st_info values (102, '김길동', 'Computer');  
Query OK, 1 row affected (0.000 sec)
```

```
MariaDB [st_db]> insert into st_info values (103, '홍길동', 'Game');  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [st_db]> insert into st_grade values (101, 90, 80);  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [st_db]> insert into st_grade values (102, 70, 95);  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [st_db]> insert into st_grade values (103, 80, 65);  
Query OK, 1 row affected (0.001 sec)
```

02 MariaDB 설치와 사용

■ 레코드 검색하기

- ① st_info 테이블의 전체 레코드를 출력

```
MariaDB [st_db]> select * from st_info;
```

ST_ID	NAME	DEPT
101	이길동	Game
102	김길동	Computer
103	홍길동	Game

```
3 rows in set (0.000 sec)
```

02 MariaDB 설치와 사용

■ 레코드 검색하기

② 학번이 101인 학생의 이름과 학과를 검색

```
MariaDB [st_db]> select NAME, DEPT from st_info where ST_ID=101;
```

NAME	DEPT
이길동	Game

1 row in set (0.000 sec)

③ 학번이 101인 학생의 리눅스 성적을 검색

```
MariaDB [st_db]> select Linux from st_grade where ST_ID=101;
```

Linux
90

1 row in set (0.000 sec)

02 MariaDB 설치와 사용

■ 레코드 검색하기

- ④ 학번이 101인 학생의 이름과 학과, DB 성적을 한 번에 검색

```
MariaDB [st_db]> select st_info.NAME, st_info.DEPT, st_grade.DB  
-> from st_info, st_grade  
-> where st_info.ST_ID=101 and st_grade.ST_ID=101;
```

NAME	DEPT	DB
이길동	Game	80

1 row in set (0.000 sec)

02 MariaDB 설치와 사용

■ 데이터 수정하기

- 테이블의 데이터가 잘못되었다면 수정해야 함
- 예를 들어 학번이 101인 학생의 DB 성적을 80점에서 90점으로 수정해야 한다면 다음과 같이 update 문장을 사용. select로 확인해 보면 90점으로 수정되었음을 알 수 있음

```
MariaDB [st_db]> update st_grade set DB=90 where ST_ID=101;  
Query OK, 1 row affected (0.001 sec)  
Rows matched: 1  Changed: 1  Warnings: 0
```

```
MariaDB [st_db]> select * from st_grade where ST_ID=101;
```

ST_ID	Linux	DB
101	90	90

```
1 row in set (0.000 sec)
```

02 MariaDB 설치와 사용

■ MariaDB 관리

- MariaDB를 관리하기 위해 사용하는 명령으로 mariadb-admin이 있음
- 이 명령으로 MariaDB 계정의 암호를 설정하고 상태도 확인할 수 있음
- 기존 mysqladmin은 mariadbadmin에 대한 심볼릭 링크로 존재

mariadb-admin

- 기능 MariaDB 서버를 관리한다.
- 형식 mariadb-admin [옵션]
- 옵션 status: MariaDB 서버의 현재 상태 정보를 출력한다.
version: MariaDB 서버의 버전 정보를 출력한다.
password 암호: 계정의 암호를 지정한다.
- 사용 예 mariadb-admin password '123456'

02 MariaDB 설치와 사용

■ MariaDB 상태 정보 출력하기

- mariadb-admin status 명령은 MariaDB 서버의 현재 상태 정보를 출력

```
[root@localhost ~]# mariadb-admin status
Uptime: 1526  Threads: 1  Questions: 45  Slow queries: 0  Opens: 26  Open tables: 15
Queries per second avg: 0.029
```

- Uptime: 서버가 동작한 시간을 초 단위로 표시
- Threads: 현재 동작 중인 MariaDB 서버의 스레드 수
- Questions: 서버가 동작한 이후 처리한 질의 수
- Slow queries: 일정 시간보다 처리 시간이 길어진 질의 수
- Opens: 서버가 열었던 테이블 수
- Flush tables: flush, refresh, reload 명령을 수행한 횟수
- Open tables: 현재 열려 있는 테이블 수

02 MariaDB 설치와 사용

■ MariaDB 버전 정보 출력하기

- mariadb-admin version 명령은 MariaDB 서버의 버전 정보를 출력
- 이때 MariaDB 서버의 상태 정보도 같이 출력됨

```
[root@localhost ~]# mariadb-admin version
mariadb-admin  Ver 9.1 Distrib 10.5.16-MariaDB, for Linux on x86_64
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Server version          10.5.16-MariaDB
Protocol version        10
Connection              Localhost via UNIX socket
UNIX socket             /var/lib/mysql/mysql.sock
Uptime:                 26 min 35 sec

Threads: 1  Questions: 46  Slow queries: 0  Opens: 26  Open tables: 15  Queries per
second avg: 0.028
```

02 MariaDB 설치와 사용

■ MariaDB 서버 암호 설정하기

- mariadb-admin password 명령으로 MariaDB 서버에 암호를 설정할 수 있음

```
[root@localhost ~]# mariadb-admin password '123456'
```

- 암호를 설정하기 전에는 일반 사용자도 mariadb 명령으로 바로 접속할 수 있었지만, 암호를 설정한 뒤 mariadb 명령으로 접속하려고 하면 다음과 같이 오류 메시지가 출력됨

```
[user1@localhost ~]$ mariadb  
ERROR 1045 (28000): Access denied for user 'user1'@'localhost' (using password: NO)
```

02 MariaDB 설치와 사용

■ MariaDB 서버 암호 설정하기

- 암호를 입력하기 위해 -u 옵션으로 계정 이름을 지정하고, -p 옵션으로 암호를 입력할 것이라고 알려주어야 함. 그러면 암호를 물어보는데, 정상적인 암호를 입력하면 접속됨

```
[user1@localhost ~]$ mariadb -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.5.16-MariaDB MariaDB Server

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]>
```

02 MariaDB 설치와 사용

■ 테이블 생성 및 레코드 입력·검색·수정

① 앞에서 사용한 st_db 데이터베이스를 그대로 사용

```
MariaDB [(none)]> use st_db;
```

② 새로운 테이블을 추가한다. 테이블 구조는 오른쪽을 참조

```
MariaDB [st_db]> create table st_janghak (ST_ID int Not Null primary key, Money int);  
Query OK, 0 rows affected (0.003 sec)
```

③ st_janghak 테이블의 구조를 확인

```
MariaDB [st_db]> explain st_janghak;
```

```
+-----+-----+-----+-----+-----+-----+  
| Field  | Type    | Null  | Key  | Default | Extra |  
+-----+-----+-----+-----+-----+-----+  
| ST_ID  | int(11) | NO    | PRI  | NULL    |       |  
| Money  | int(11) | YES   |      | NULL    |       |  
+-----+-----+-----+-----+-----+-----+  
2 rows in set (0.001 sec)
```

테이블 이름: st_janghak

학번(ST_ID)	장학금(Money)
101	500000
102	1000000
103	1500000

02 MariaDB 설치와 사용

■ 테이블 생성 및 레코드 입력·검색·수정

④ 앞의 ②번 테이블을 참조하여 데이터를 입력

```
MariaDB [st_db]> insert into st_janghak values (101, 5000000);  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [st_db]> insert into st_janghak values (102, 10000000);  
Query OK, 1 row affected (0.001 sec)
```

```
MariaDB [st_db]> insert into st_janghak values (103, 15000000);  
Query OK, 1 row affected (0.002 sec)
```


02 MariaDB 설치와 사용

■ 테이블 생성 및 레코드 입력·검색·수정

- ⑤ st_janghak 테이블의 모든 레코드를 검색하여 데이터가 제대로 입력되었는지 확인

```
MariaDB [st_db]> select * from st_janghak;
```

```
+-----+-----+  
| ST_ID | Money |  
+-----+-----+  
| 101   | 500000 |  
| 102   | 1000000 |  
| 103   | 1500000 |  
+-----+-----+  
3 rows in set (0.000 sec)
```

02 MariaDB 설치와 사용

■ 테이블 생성 및 레코드 입력·검색·수정

⑥ st_info 테이블에 나이(Age) 필드를 추가

```
MariaDB [st_db]> alter table st_info add Age int;
```

```
Query OK, 0 rows affected (0.002 sec)
```

```
Records: 0 Duplicates: 0 Warnings: 0
```

```
MariaDB [st_db]> explain st_info;
```

Field	Type	Null	Key	Default	Extra
ST_ID	int(11)	NO	PRI	NULL	
NAME	varchar(20)	YES		NULL	
DEPT	varchar(25)	YES		NULL	
Age	int(11)	YES		NULL	

```
4 rows in set (0.001 sec)
```

02 MariaDB 설치와 사용

■ 테이블 생성 및 레코드 입력·검색·수정

⑦ 새로 추가한 Age 필드에 값을 입력

```
MariaDB [st_db]> update st_info set Age=20 where ST_ID=101;  
Query OK, 1 row affected (0.046 sec)  
Rows matched: 1  Changed: 1  Warnings: 0
```

```
MariaDB [st_db]> update st_info set Age=21 where ST_ID=102;  
Query OK, 1 row affected (0.001 sec)  
Rows matched: 1  Changed: 1  Warnings: 0
```

```
MariaDB [st_db]> update st_info set Age=19 where ST_ID=103;  
Query OK, 1 row affected (0.001 sec)  
Rows matched: 1  Changed: 1  Warnings: 0
```

```
MariaDB [st_db]> select * from st_info;
```

ST_ID	NAME	DEPT	Age
101	이길동	Game	20
102	김길동	Computer	21
103	홍길동	Game	19

```
3 rows in set (0.001 sec)
```

02 MariaDB 설치와 사용

■ 테이블 생성 및 레코드 입력·검색·수정

⑧ 학번, 이름, 나이, 리눅스 성적, 장학금이 출력되도록 검색

```
MariaDB [st_db]> select st_info.ST_ID, st_info.NAME, st_info.Age, st_grade.Linux, st_
janghak.Money
  -> from st_info, st_grade, st_janghak
  -> where st_info.ST_ID=st_grade.ST_ID and st_info.ST_ID=st_janghak.ST_ID;
```

ST_ID	NAME	Age	Linux	Money
101	이길동	20	90	500000
102	김길동	21	70	1000000
103	홍길동	19	80	1500000

3 rows in set (0.002 sec)

02 MariaDB 설치와 사용

■ 데이터베이스 생성, 테이블 생성, 레코드 입력·검색

혼자해보기 데이터베이스 생성, 테이블 생성, 레코드 입력·검색

다음 예를 기준으로 데이터베이스를 구축해 보자.

- 데이터베이스 이름: BookDB
- 테이블:

1) 이름: Author

저자 번호(a_no)	이름(a_name)	나이(age)
201	이저자	35
202	김저자	40

2) 이름: Book

책 번호(b_no)	서명(b_name)	저자 번호(a_no)
A01	Linux 실습	201
A02	DB 개론	201
B01	소설 A	301

- 1 데이터베이스를 생성한다.
- 2 테이블을 생성한다.
- 3 테이블의 구조를 확인한다.
- 4 테이블에 데이터를 입력한다.
- 5 Author 테이블의 모든 데이터를 검색한다.
- 6 Author 테이블과 Book 테이블에서 '이저자'의 책 이름과 나이를 검색한다.
- 7 Book 테이블에서 '소설A'의 저자 번호를 301에서 202로 수정한다.
- 8 Book 테이블에서 책 번호가 B01인 레코드를 검색하여 제대로 수정되었는지 확인한다.

03 웹 서버 설치와 사용

03 웹 서버 설치와 사용

■ 아파치 설치 및 접속

- 아파치의 패키지 이름은 httpd. dnf 명령으로 httpd를 설치

```
[root@localhost ~]# dnf install httpd
마지막 메타자료 만료확인(0:35:08 이전): 2023년 10월 03일 (화) 오후 05시 13분 53초.
종속성이 해결되었습니다.
(생략)
설치되었습니다:
  apr-1.7.0-11.el9.x86_64          apr-util-1.6.1-20.el9_2.1.x86_64  apr-util-
bdb-1.6.1-20.el9_2.1.x86_64
  apr-util-openssl-1.6.1-20.el9_2.1.x86_64  httpd-2.4.53-11.el9_2.5.x86_64  httpd-
core-2.4.53-11.el9_2.5.x86_64
  httpd-filesystem-2.4.53-11.el9_2.5.noarch  httpd-tools-2.4.53-11.el9_2.5.x86_64  mod_http2-
1.15.19-4.el9_2.4.x86_64
  mod_lua-2.4.53-11.el9_2.5.x86_64          rocky-logos-httpd-90.14-1.el9.noarch

완료되었습니다!
```

03 웹 서버 설치와 사용

■ 아파치 활성화하기

- 아파치를 활성화하는 서비스의 이름은 httpd.service. systemctl 명령으로 시작하면 됨
- httpd는 독자형으로 동작하는 데몬으로, ps 명령을 사용해 확인해 보면 httpd 데몬이 여러 개 동작하고 있음

```
[root@localhost ~]# systemctl start httpd
[root@localhost ~]# ps -ef | grep httpd
root      42658      1  0 17:52 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    42659    42658  0 17:52 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    42660    42658  0 17:52 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    42661    42658  0 17:52 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
apache    42662    42658  0 17:52 ?        00:00:00 /usr/sbin/httpd -DFOREGROUND
root      42877    42116  0 17:52 pts/0    00:00:00 grep --color=auto httpd
```


03 웹 서버 설치와 사용

■ 웹 서버 접속하기

- 방화벽에서 http 서비스를 등록

```
[root@localhost ~]# firewall-cmd --add-service=http  
success
```

- 윈도우에서 웹 브라우저를 띄우고 리눅스의 웹 서버에 접속해 보자. 웹 브라우저에서 http://IP 주소를 입력하여 접속하면 [그림 13-2]와 같은 화면이 나옴

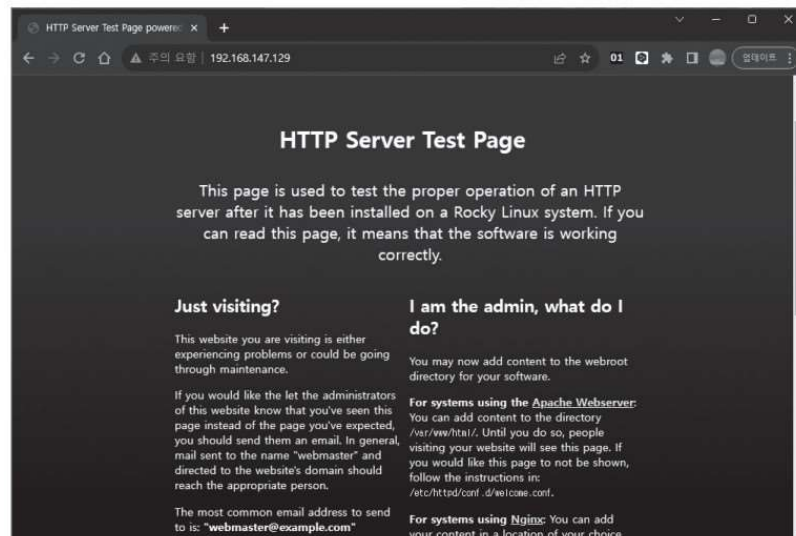


그림 13-2 로키 리눅스의 아파치 웹 서버 초기 화면

03 웹 서버 설치와 사용

■ 웹 서버 접속하기

- 아파치 웹 서버의 기본 디렉터리는 /var/www/html. 이 디렉터리에 html 파일이 있으면 그 파일이 출력됨
- 간단한 html 파일을 하나 만들어보자. 다음 내용을 vi로 작성하여 이 디렉터리에 저장

```
[root@localhost html]# vi index.html
<html>
<head>
<title>
html test
</title>
</head>
<body>
Rocky Linux Web Page Test!!!
</body>
</html>
~
:wq
```

03 웹 서버 설치와 사용

■ 웹 서버 접속하기

- 웹 브라우저에서 다시 리눅스에 접속해 보자.
- [그림 13-2]가 아니라 새로 작성한 내용이 [그림 13-3]과 같이 출력됨

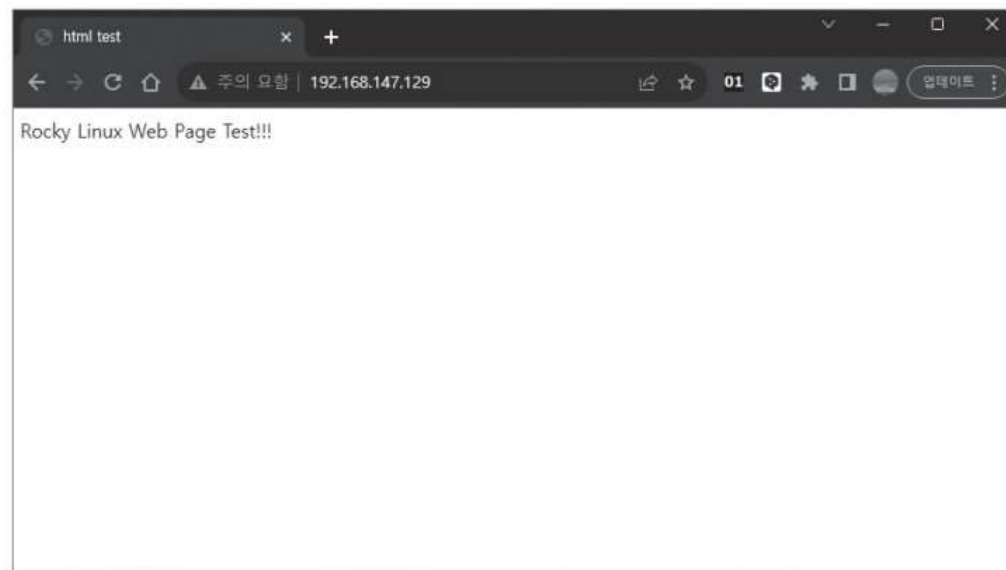


그림 13-3 웹 페이지 출력 화면

03 웹 서버 설치와 사용

■ 일반 사용자 계정을 위한 웹 설정하기

- /var/www/html 디렉터리는 웹 서버의 기본 디렉터리로, 리눅스를 사용하는 여러 사용자가 공용으로 사용할 수 있는 디렉터리가 아님
- 일반 사용자 계정에서 웹 서비스를 제공하려면 아파치 설정 파일을 수정해야 함
- 아파치 웹 서버의 설정 파일 중 사용자 계정의 웹 설정과 관련된 파일은 /etc/httpd/conf.d/userdir.conf 파일

03 웹 서버 설치와 사용

■ 일반 사용자 계정을 위한 웹 설정하기

- ① vi로 /etc/httpd/conf.d/userdir.conf 파일을 열어 수정(볼드체로 표시된 부분)

```
[root@localhost html]# vi /etc/httpd/conf.d/userdir.conf
(생략)
<IfModule mod_userdir.c>
    #
    # UserDir is disabled by default since it can confirm the presence
    # of a username on the system (depending on home directory
    # permissions).
    #
    # UserDir disabled    → 앞에 #를 추가한다.
    (생략)
    UserDir public_html → 앞의 #를 삭제한다.
</IfModule>
(생략)
~
:wq
```

'UserDir disabled'는 사용자 디렉터리에서 웹 페이지 구축을 허용할 것인지 확인하는 부분이므로 이 앞에 #를 넣어 주석 처리하면 허용한다는 의미가 됨.

'UserDir public_html'은 사용자 디렉터리에서 public_html을 웹 페이지의 기본 디렉터리로 사용한다는 뜻

03 웹 서버 설치와 사용

■ 일반 사용자 계정을 위한 웹 설정하기

- ② httpd의 설정 파일이 수정되었으므로 httpd.service를 다시 동작 시킴

```
[root@localhost html]# systemctl restart httpd.service
```

- ③ user1 계정에서 홈 디렉터리에 public_html 디렉터를 만들고 디렉터리의 접근 권한을 조정

```
[user1@localhost ~]$ mkdir public_html  
[user1@localhost ~]$ chmod 701 .  
[user1@localhost ~]$ chmod 701 public_html
```

03 웹 서버 설치와 사용

■ 일반 사용자 계정으로 웹 페이지 작성하기

- 사용자 계정의 public_html 디렉터리 아래에 index.html 파일을 만들고 확인

① public_html 디렉터리 아래에 index.html 파일을 생성

```
[user1@localhost ~]$ cd public_html
[user1@localhost public_html]$ vi index.html
<html>
<head>
<title>user1 html test</title>
</head>
<body>
User1 Web Page!!!
</body>
</html>
~
:wq
```

03 웹 서버 설치와 사용

■ 일반 사용자 계정에서 웹 페이지 작성하기

- ② 웹 브라우저에서 다음과 같이 입력하여 html 문서를 확인

IP 주소에는 본인 리눅스 시스템의 IP 주소를 입력. 사용자 계정 앞에 ~를 넣는 것을 잊으면 안 됨

```
http://IP주소/~user1
```


03 웹 서버 설치와 사용

■ 일반 사용자 계정에서 웹 페이지 작성하기

- ③ 만약 [그림 13-4]처럼 브라우저에서 Forbidden 오류가 발생한다면 앞의 과정에서 접근 권한을 잘못 조정했거나 SELinux로 보안이 걸려 있기 때문

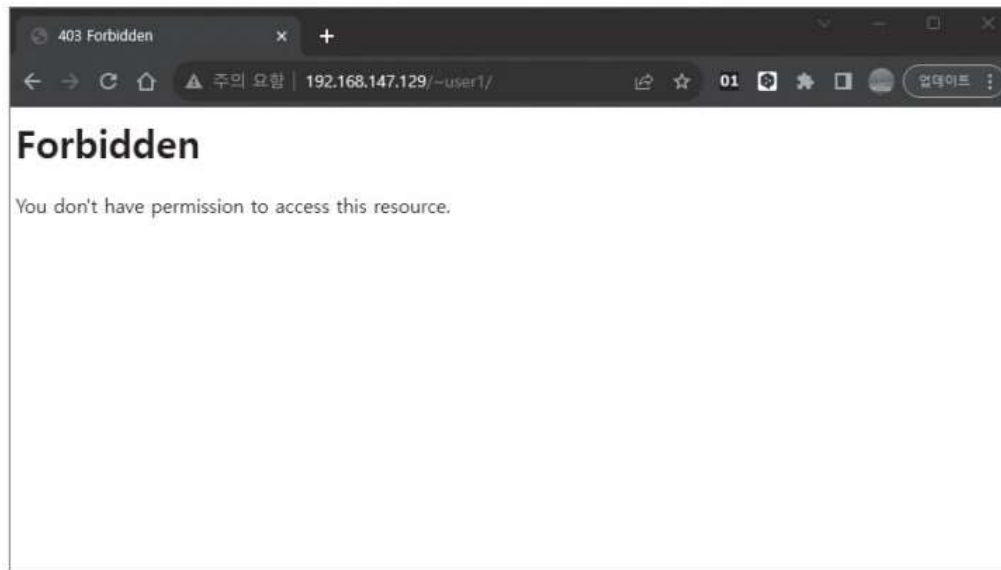


그림 13-4 Forbidden 오류 화면

03 웹 서버 설치와 사용

■ 일반 사용자 계정에서 웹 페이지 작성하기

- ③ 권한을 다시 확인한 뒤 SELinux의 보안을 해제하고 확인
SELinux 기능을 정지하고 웹 페이지에 접속

```
[root@localhost ~]# setenforce 0
```

- ④ 브라우저에서 html 문서의 내용을 다시 확인. [그림 13-5]와 같이 보이면 정상적으로 처리된 것

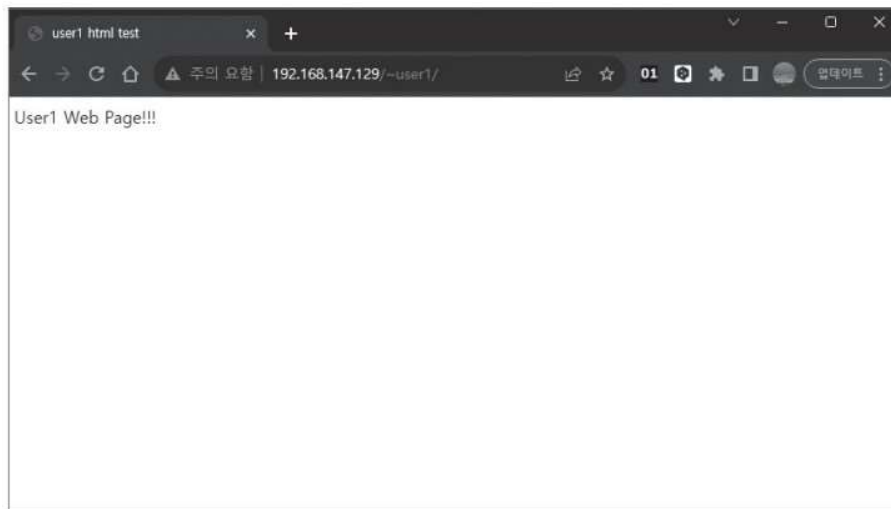


그림 13-5 사용자 웹 페이지 출력 화면

03 웹 서버 설치와 사용

■ APM과 게시판 설치

- 리눅스의 아파치 웹 서버와 웹 프로그래밍 언어인 PHP, 데이터베이스인 MySQL을 묶어서 APM이라고 부름. 그런데 MySQL이 MariaDB로 대체되었으므로 이제 APM의 M은 MySQL이 아니라 MariaDB를 의미
- 일반적으로 아파치와 PHP, 데이터베이스가 연동되도록 설치하고 이를 이용한 웹 응용 프로그램을 사용함. 우리가 웹 페이지에서 흔히 사용하는 게시판도 APM을 이용하여 개발된 것이 많음
- PHP를 설치하여 APM을 완성하고, 공개 게시판을 설치하여 APM이 제대로 동작하는지 확인해 보자

03 웹 서버 설치와 사용

■ APM 설치하기

- APM을 연동하기 위해 설치해야 할 PHP 패키지는 php, php-gd, php-mysqlnd인데 dnf로 한 번에 설치함. 의존성이 있는 패키지를 포함하여 총 여덟 개의 패키지가 설치됨

```
[root@localhost ~]# dnf install php php-gd php-mysqlnd
마지막 메타자료 만료확인(1:25:52 이전): 2023년 10월 03일 (화) 오후 05시 13분 53초.
종속성이 해결되었습니다.
(생략)
설치되었습니다:
  nginx-filessystem-1:1.20.1-14.el9.noarch      php-8.0.27-1.el9_1.x86_64
php-cli-8.0.27-1.el9_1.x86_64
  php-common-8.0.27-1.el9_1.x86_64             php-fpm-8.0.27-1.el9_1.x86_64
php-gd-8.0.27-1.el9_1.x86_64
  php-mbstring-8.0.27-1.el9_1.x86_64           php-mysqlnd-8.0.27-1.el9_1.x86_64
php-opcache-8.0.27-1.el9_1.x86_64
  php-pdo-8.0.27-1.el9_1.x86_64                php-xml-8.0.27-1.el9_1.x86_64

완료되었습니다!
```

03 웹 서버 설치와 사용

■ PHP의 동작 확인하기

- PHP가 정상적으로 동작하는지 확인해 보자
- 아파치에 PHP 모듈이 추가된 것이므로 httpd. service를 다시 동작 시켜야 함.
MariaDB도 다시 동작 시킴

```
[root@localhost ~]# systemctl restart httpd  
[root@localhost ~]# systemctl restart mariadb
```

- PHP의 동작을 확인하기 위해 /var/www/html에 다음과 같이 파일을 만듦

```
[root@localhost html]# vi phpinfo.php  
<?php phpinfo(); ?>  
~  
:wq
```

03 웹 서버 설치와 사용

■ PHP의 동작 확인하기

- 웹 브라우저에서 `http://IP주소/phpinfo.php`로 접속했을 때 [그림 13-6]과 같이 출력되면 아파치와 PHP가 정상적으로 연동된 것



그림 13-6 PHP 동작 확인 화면

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ① 웹 브라우저(파이어폭스)를 실행하여 그누보드 사이트(sir.kr)에 접속해서 다운로드. 2023년 10월 현재 그누보드5 5.5.8.3.4 버전(gnuboard-gnuboard5-v5.5.8.3.4-0-g4b9b1af.tar.gz)이 최신 버전

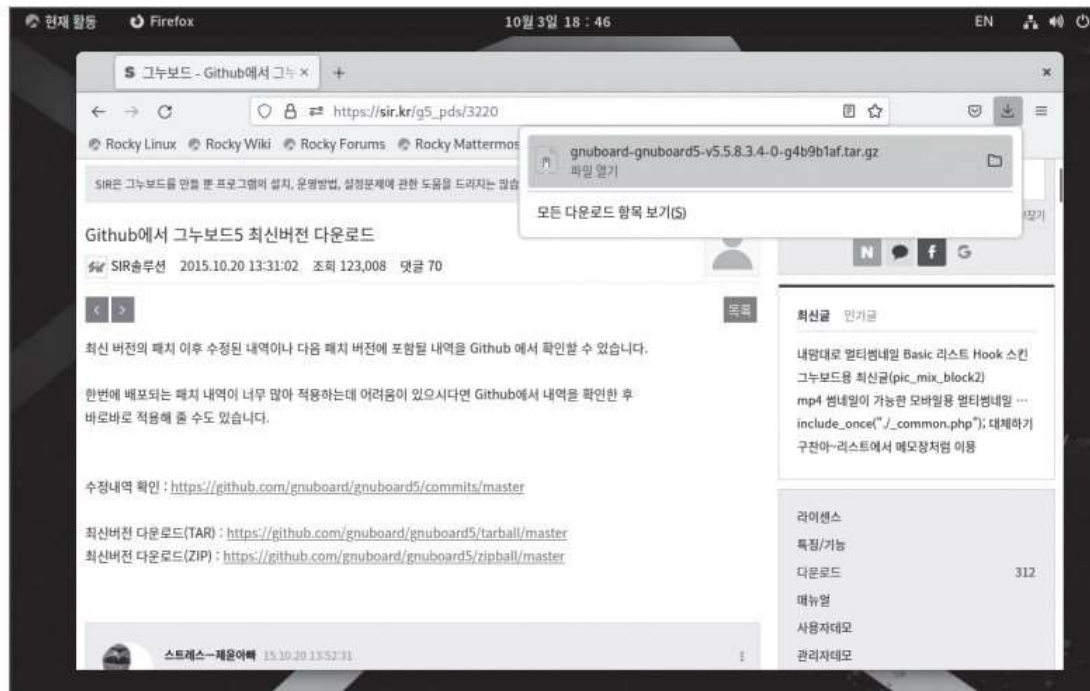


그림 13-7 그누보드 최신 버전 다운로드

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ② 다운로드한 파일을 /var/www/html로 복사. 다음과 같이 /var/www/html 디렉터리로 복사

```
[root@localhost html]# cp ~user1/다운로드/gnuboard-gnuboard5-v5.5.8.3.4-0-g4b9b1af.tar.gz .  
[root@localhost html]# ls gnuboard*  
gnuboard-gnuboard5-v5.5.8.3.4-0-g4b9b1af.tar.gz
```

- ③ tar 명령으로 묶인 압축 파일을 모두 풀고, 디렉터리 이름을 gnuboard5로 변경

```
[root@localhost html]# tar xzvf gnuboard-gnuboard5-v5.5.8.3.4-0-g4b9b1af.tar.gz  
(생략)  
gnuboard-gnuboard5-4b9b1af/version.php  
gnuboard-gnuboard5-4b9b1af/yc4_import.php  
gnuboard-gnuboard5-4b9b1af/yc4_import_run.php  
[root@localhost html]# mv gnuboard-gnuboard5-4b9b1af gnuboard5
```


03 웹 서버 설치와 사용

■ 게시판 설치하기

④ 그누보드가 사용할 데이터베이스를 구축

```
[root@localhost html]# mariadb
Welcome to the MariaDB monitor.  Commands end with ; or \g.
(생략)

MariaDB [(none)]> create database gnu;
Query OK, 1 row affected (0.002 sec)
```

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ⑤ 윈도우에서 웹 브라우저로 접속하여 설정. 접속할 주소는 `http://IP주소/ gnuboard5`. 다운로드한 그누보드의 tar 파일을 풀어놓은 디렉터리에 따라 웹 브라우저로 접속할 때 경로명도 달라져야 함. 웹 브라우저로 접속하면 [그림 13-8]과 같은 메시지 창이 뜬. DB 설정이 없다는 내용임. ④ 번에서 생성한 DB를 아직 설정하지 않은 것이므로 [그누보드5 설치하기]를 클릭함

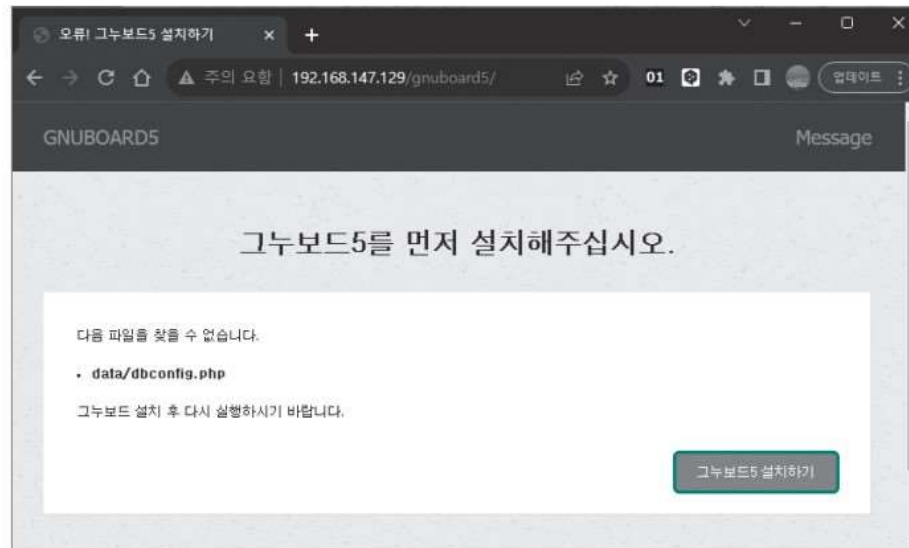


그림 13-8 그누보드 설치-DB 설정 오류 화면

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ⑥ [그림 13-9]와 같은 화면이 출력. 지시에 따라 data 디렉터리를 만들고 접근 권한을 707로 수정

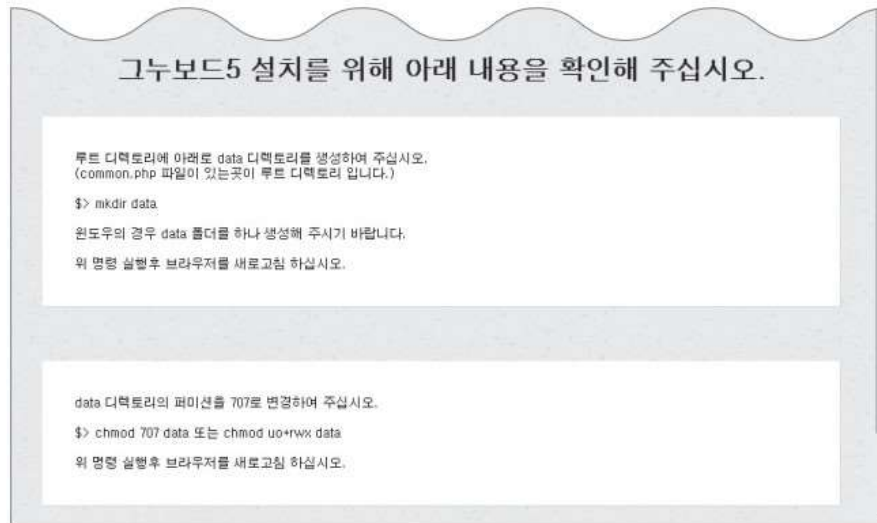


그림 13-9 그누보드 설치-data 디렉터리 생성

```
[root@localhost html]# cd gnuboard5
[root@localhost gnuboard5]# mkdir data
[root@localhost gnuboard5]# chmod 707 data
[root@localhost gnuboard5]# ls -ld data
drwx---rwx. 2 root root 6 10월  3 20:03 data
```

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ⑦ 접근 권한을 조정한 후 브라우저에서 새로 고침을 하면 [그림 13-10]과 같이 라이선스에 동의하는 창이 뜬다. 라이선스에 동의해야 설치를 계속 진행할 수 있음

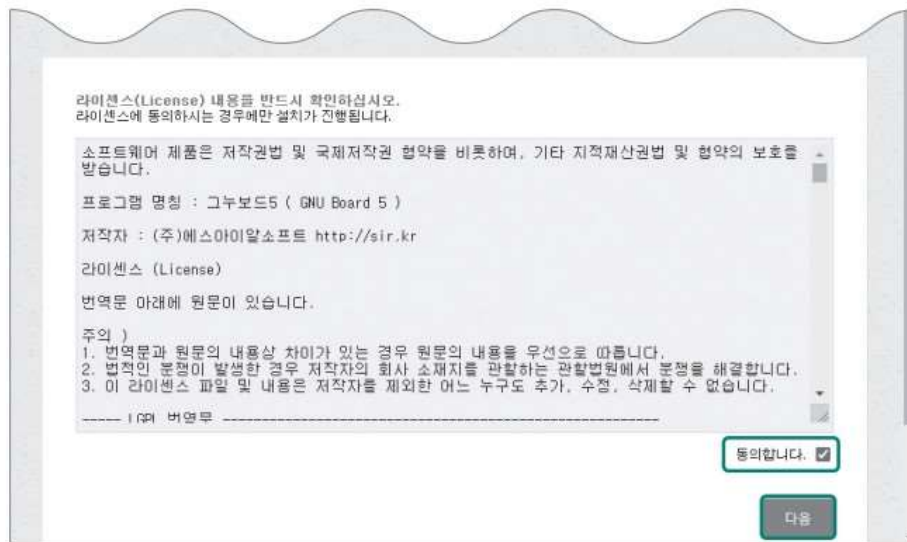


그림 13-10 그누보드 설치-라이선스 확인

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ⑧ MariaDB에 설정한 DB 정보와 그누보드 관리자 정보를 입력. MariaDB 정보에서 사용자 정보는 다음과 같이 일반 사용자에게 해당 데이터베이스에 대한 권한을 부여하고 지정함. MariaDB의 root 계정을 그냥 사용할 경우 보안에 문제가 생길 수 있음

```
MariaDB [(none)]> grant all privileges on gnu.* to gnuser@localhost identified by '123456';  
Query OK, 0 rows affected (0.006 sec)
```

03 웹 서버 설치와 사용

■ 게시판 설치하기

- ⑧ 여기서 지정한 gnuser 사용자는 리눅스 시스템에 로그인할 때 필요한 계정이 아닌 MariaDB 사용자 계정.
- 그누보드의 최고관리자admin 암호도 지정.
- 실습에서 암호는 편의상 123456으로 했으나 실무에서는 이보다 강력한 암호를 사용해야 함

The screenshot shows the 'GNUBOARD5 INSTALLATION' web page. It contains two main sections: 'MySQL 정보입력' (MySQL Information Input) and '최고관리자 정보입력' (Administrator Information Input).

MySQL 정보입력

Host	localhost
User	gnuser
Password	123456
DB	gnu
TABLE명 접두사	g5_
쇼핑몰TABLE명 접두사	g5_shop_
그누보드5 재설치	<input type="checkbox"/> 재설치
쇼핑몰설치	<input checked="" type="checkbox"/> 설치

최고관리자 정보입력

회원 ID	admin
비밀번호	123456
이름	최고관리자
E-mail	admin@domain.com

주의! 이미 그누보드5가 존재한다면 DB 자료가 망실되므로 주의하십시오.
주의사항을 이해했으며, 그누보드 설치를 계속 진행하시려면 다음을 누르십시오.

다음

그림 13-11 그누보드 설치-DB 정보와 관리자 암호 설정

03 웹 서버 설치와 사용

■ 게시판 설치하기

⑨ 그누보드 설치가 완료.

[새로운 그누보드5로 이동]을 클릭

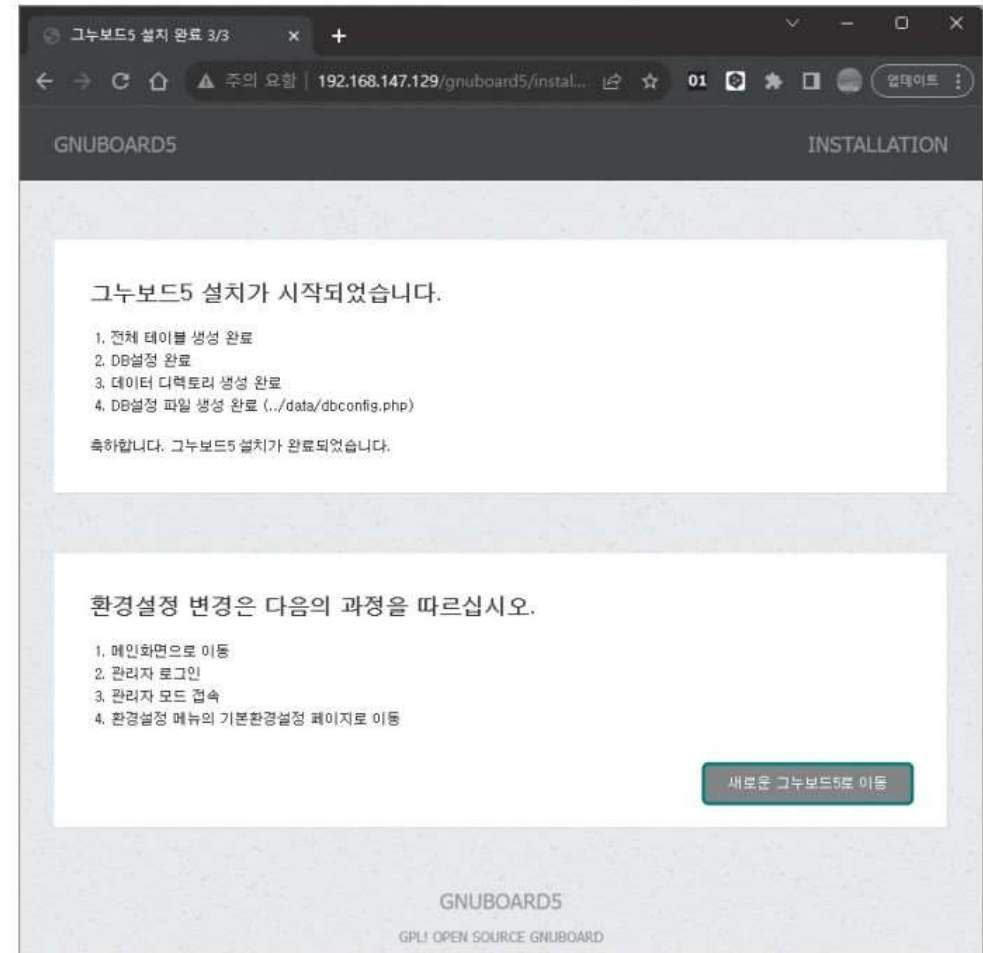
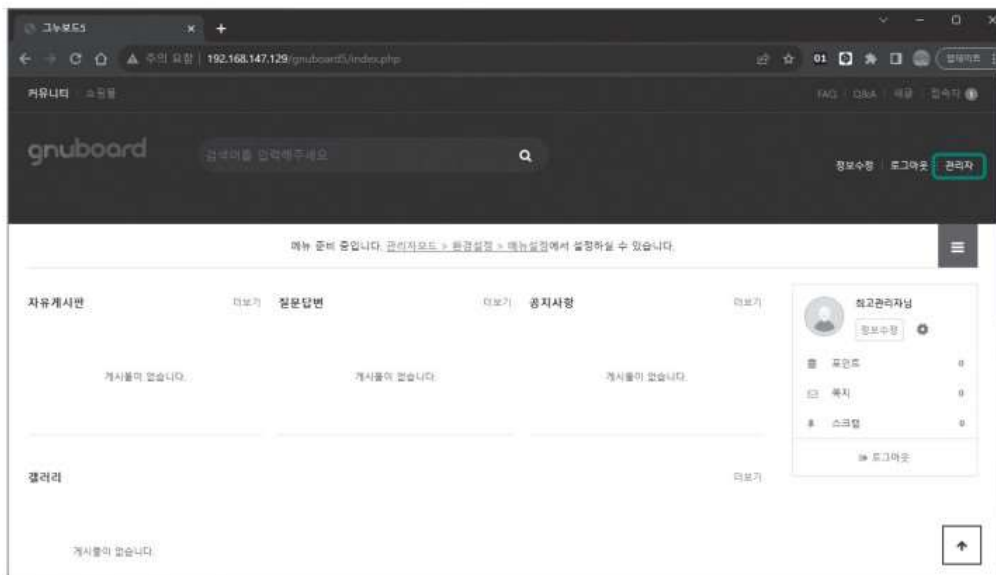


그림 13-12 그누보드 설치-설치 완료

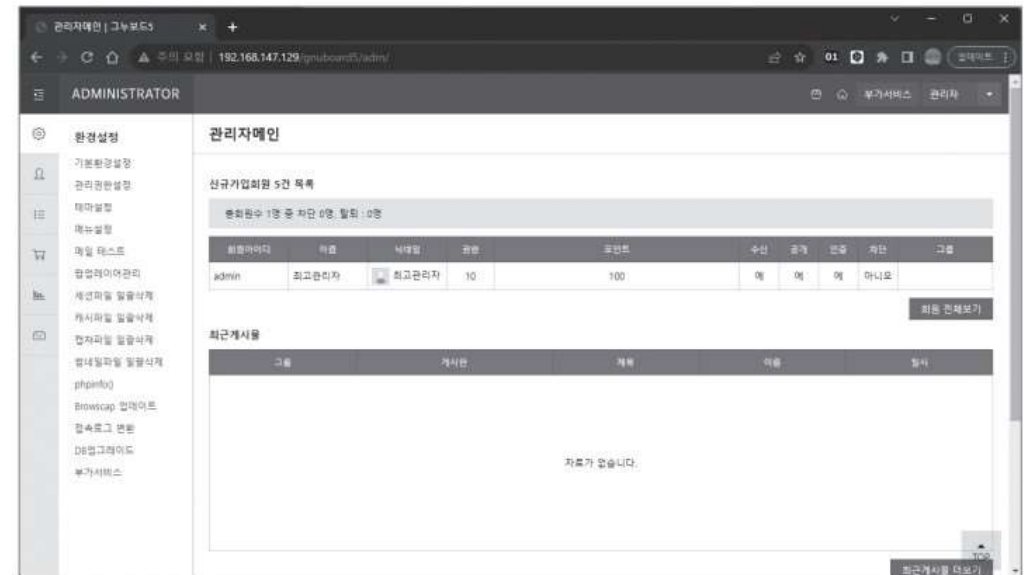
03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

- ① [그림 13-13]에서 admin 계정으로 로그인한 다음 우측 상단의 [관리자]를 클릭하면 관리자메인 화면으로 전환



(a) admin 로그인 화면



(b) 관리자메인 화면

그림 13-14 그누보드 사용-관리자메인 화면으로 전환

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

② 게시판을 만들려면 먼저 게시판 그룹을 생성해야 함.

관리자메인 화면에서 [게시판관리]-[게시판그룹관리]를 선택하면 게시판관리 화면으로 바뀜.
여기서 오른쪽 상단의 [게시판그룹 추가]를 클릭

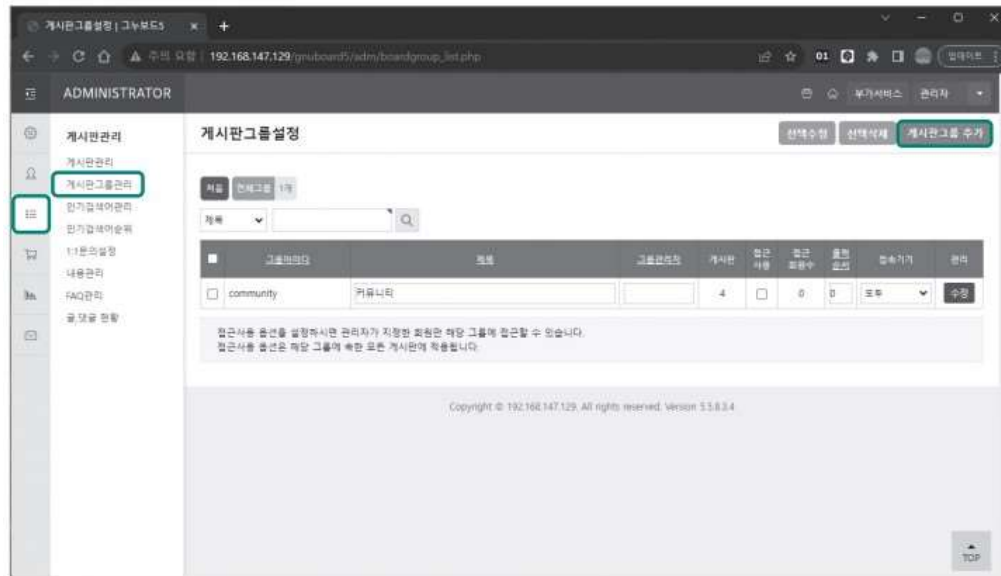


그림 13-15 그누보드 사용-게시판그룹관리 선택

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

③ 게시판그룹 생성 화면이 나타남.

여기서 그룹 ID와 그룹 제목만 입력하고 [확인]을 클릭하면 게시판 그룹이 생성됨

The screenshot shows the '게시판그룹 생성' (Board Group Creation) form in the GnuBoard administrator interface. The form is titled '게시판그룹 생성' and has a '확인' (Confirm) button at the top right. The form fields are as follows:

- 그룹 ID: bbud01 (with a note: 영문자, 숫자, _ 만 가능 (공백없이))
- 그룹 제목: 게시판테스트
- 참조기: PC와 모바일 사용을 구분합니다. (PC와 모바일에서 모두 사용)
- 그룹 관리자: (empty field)
- 참조사항: 사용자 체크하시면 이 그룹에 속한 게시판은 접근가능한 권한을 접근할 수 있습니다. (checkbox checked)
- 참조사항수: 0
- 여분필드1: 여분필드 1 제목 (empty) / 여분필드 1 내용 (empty)
- 여분필드2: 여분필드 2 제목 (empty) / 여분필드 2 내용 (empty)
- 여분필드3: 여분필드 3 제목 (empty) / 여분필드 3 내용 (empty)

A red box highlights the '그룹 ID' and '그룹 제목' fields. A 'TOP' button is located at the bottom right of the form.

그림 13-16 그누보드 사용-게시판그룹 생성 화면

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

- ④ 게시판그룹 생성이 완료되면 [게시판 생성] 버튼이 나타나는데 이 버튼을 클릭하면 게시판 생성 화면이 출력

The screenshot shows a web browser window with the URL `192.168.147.129/gnuboard5/admin/boardgroup_form.php?list=0&code=0&sf=0&tab=0&page=0&w=0&gpr_id=0&...`. The page title is '게시판그룹 수정' (Board Group Edit). The form contains the following fields:

- 그룹 ID: bbcf1
- 그룹 제목: 게시판테스트
- 첨속기: PC와 모바일 사용을 구분합니다. PC와 모바일에서 모두 사용
- 그룹 관리자:
- 참조화면사용: ☐ 사용
- 참조화면수: 0
- 여분필드1: 여분필드 1 제목, 여분필드 1 내용
- 여분필드2: 여분필드 2 제목, 여분필드 2 내용
- 여분필드3: 여분필드 3 제목, 여분필드 3 내용

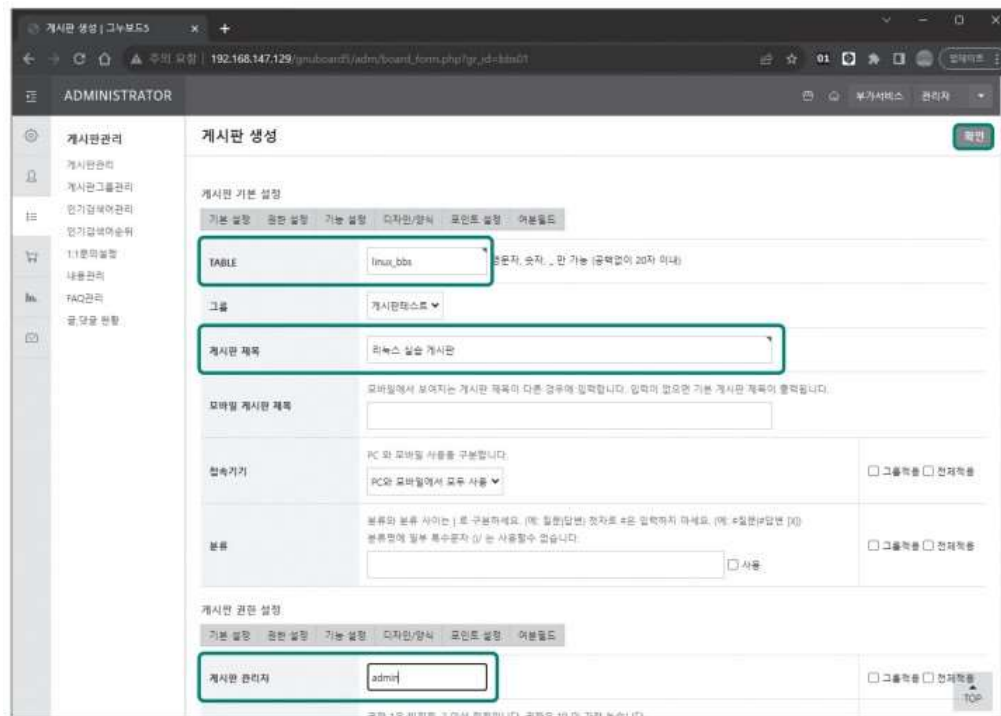
A green box highlights the '게시판생성' button next to the '그룹 제목' field.

그림 13-17 그누보드 사용-게시판생성 버튼 클릭

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

- ⑤ [그림 13-18]과 같은 게시판 생성 화면에는 게시판의 여러 속성을 설정하는 항목이 있으나 여기서는 간략하게 기본 게시판만 생성. TABLE 항목에 게시판 이름을 입력하고 게시판 제목을 입력.



게시판 권한 설정에서 게시판 관리자를 지정하고 [확인]을 클릭하면 게시판이 생성

그림 13-18 그누보드 사용 - 게시판 생성 화면

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

- ⑥ [그림 13-18]에서 [확인]을 클릭하면 게시판이 생성되고 [그림 13-19]와 같은 게시판 수정 화면이 출력됨. 게시판과 관련한 설정 중 수정할 것이 있으면 수정함.
[게시판 바로가기] 버튼을 클릭하면 해당 게시판 화면으로 바로 이동. 게시판 관리 화면에서 게시판 이름을 클릭해도 같은 화면으로 이동함

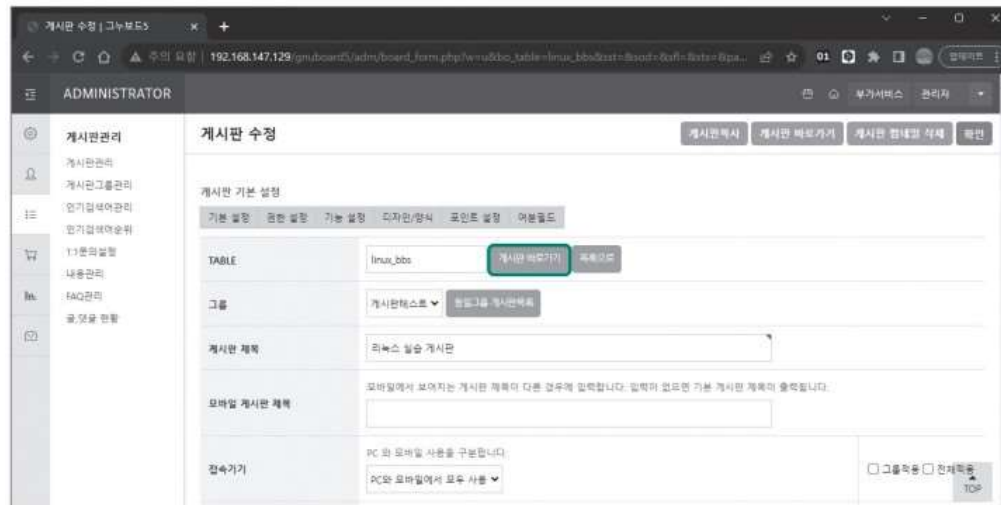


그림 13-19 그누보드 사용-게시판 수정 화면

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

⑦ [그림 13-20]과 같은 게시판 화면에서  을 클릭하여 게시물을 작성할 수 있음.

나중에 웹 페이지에 이 게시판을 연결하기 위해서 브라우저 상단의 주소를 복사해 두어야 함

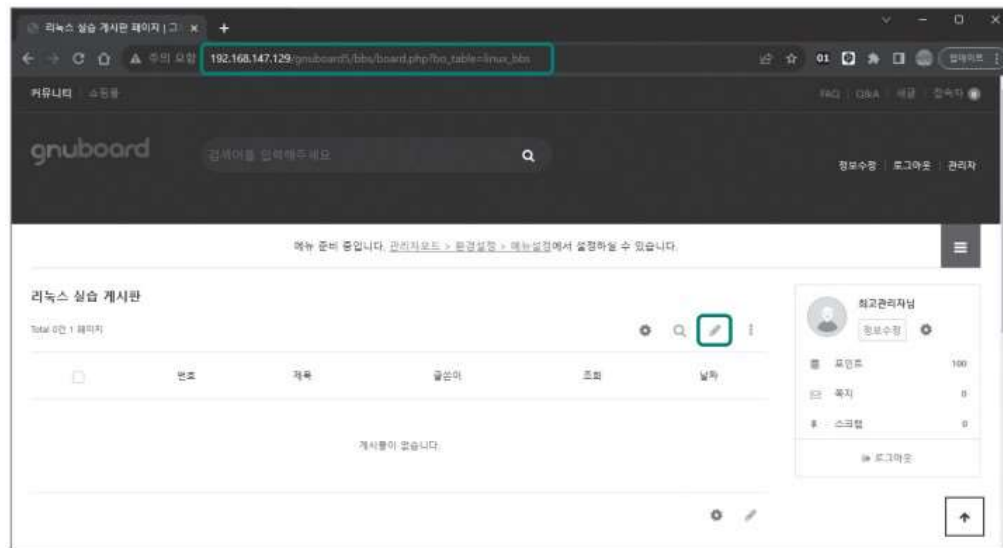


그림 13-20 그누보드 사용-게시판 화면

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

- ⑧ 앞에서 user1 사용자의 홈 디렉터리에 public_html 디렉터를 만들고 index.html 파일을 작성했음.
이 파일을 다음과 같이 수정. 게시판 주소는 [그림 13-20]의 URL 부분을 복사해서 붙이면 됨

```
[user1@localhost public_html]$ vi index.html
<html>
<head>
<title>user1 html test</title>
</head>
<body>
User1 Web Page!!!<br><br>
<a href=http://192.168.147.129/gnuboard5/bbs/board.php?bo_table=linux_bbs>리눅스 실습 게
시판 연결</a>
</body>
</html>
~
:wq
```


03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

- ⑨ 웹 브라우저에서 `http://IP 주소/~user1`에 접속하면 [그림 13-21]과 같이 '리눅스 실습 게시판 연결' 링크가 출력됨. 이 링크를 클릭하면 [그림 13-20]의 게시판이 나옴.
이것으로 게시판을 연결하는 작업이 모두 완료되었음

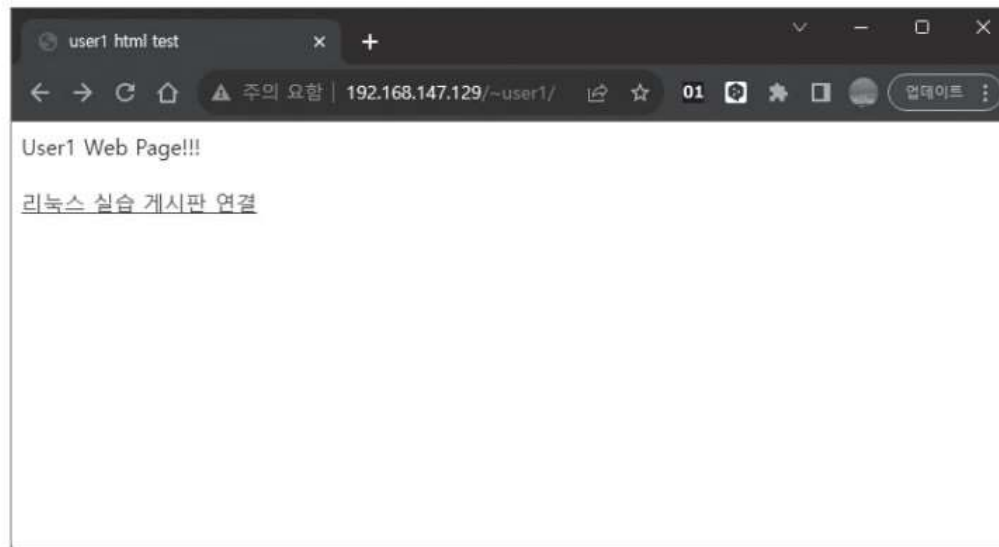


그림 13-21 그누보드 사용-게시판 연결

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

혼자해보기 사용자 웹 페이지 구축하고 게시판 연결하기

새로운 사용자 계정에서 웹 페이지를 구축하고 게시판을 연결해 보자. user2 사용자 계정에서 작업한다.

- ① user2 계정에 public_html 디렉터리를 생성하고 접근 권한을 조정한다.
- ② 기본 웹 페이지를 생성한다. public_html 디렉터리에 index.html 파일을 생성한다. html 파일은 다음 예제를 참고한다.

```
<html>
<head>
<title>user2 web page test</title>
</head>
<body>
<h1>User2 Web Page!</h1>
</body>
</html>
```

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

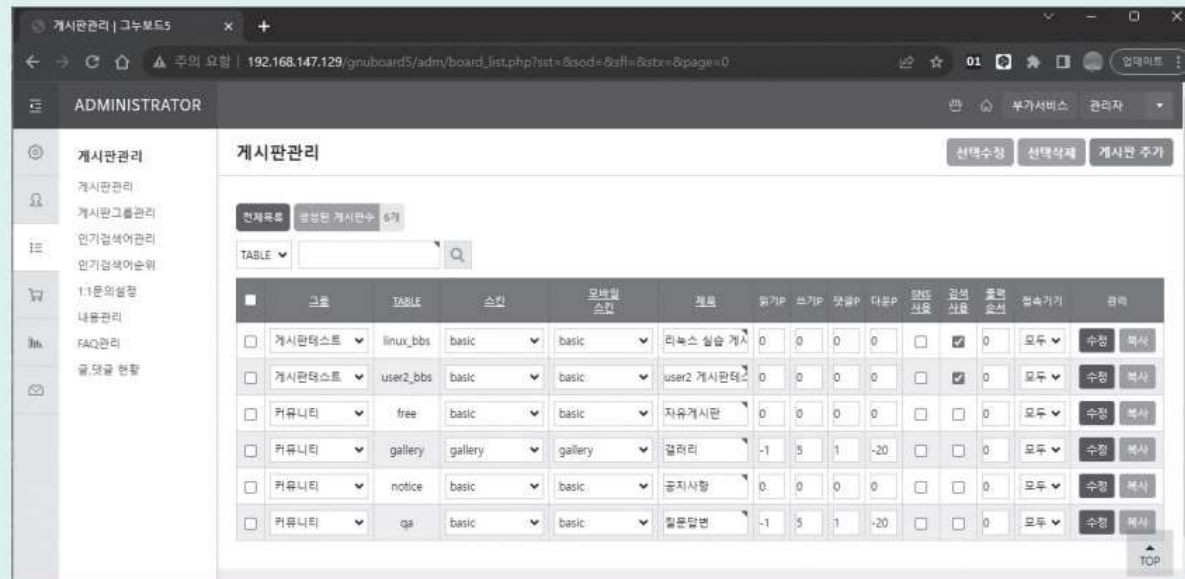
③ 웹 브라우저에서 웹 페이지에 접속하여 확인한다.



03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

4 그누보드에서 user2에게 할당할 게시판을 생성한다.



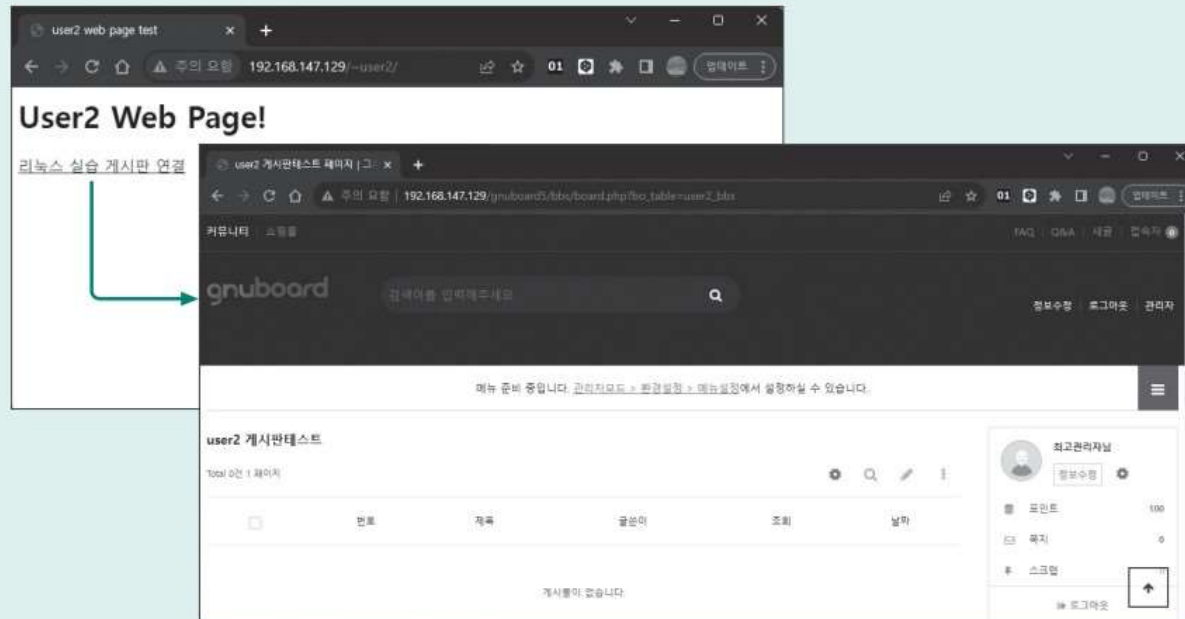
그룹	TABLE	스킨	모바일 스킨	제목	원기P	쓰기P	댓글P	다운P	공백 사용	공백 사용	공백 사용	접속가기	관리	
게시판테스트	linux_bbs	basic	basic	리눅스 실습 게시판	0	0	0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0	모두	수정	복사
게시판테스트	user2_bbs	basic	basic	user2 게시판테스트	0	0	0	0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	0	모두	수정	복사
커뮤니티	free	basic	basic	자유게시판	0	0	0	0	<input type="checkbox"/>	<input type="checkbox"/>	0	모두	수정	복사
커뮤니티	gallery	gallery	gallery	갤러리	-1	5	1	-20	<input type="checkbox"/>	<input type="checkbox"/>	0	모두	수정	복사
커뮤니티	notice	basic	basic	공지사항	0	0	0	0	<input type="checkbox"/>	<input type="checkbox"/>	0	모두	수정	복사
커뮤니티	qa	basic	basic	질문답변	-1	5	1	-20	<input type="checkbox"/>	<input type="checkbox"/>	0	모두	수정	복사

03 웹 서버 설치와 사용

■ 사용자 웹 페이지 구축하고 게시판 연결하기

⑤ 게시판을 연결할 수 있도록 ②번에서 작성한 html 파일을 수정한다.

⑥ user2 웹 페이지에서 게시판이 연결되는지 확인한다.



Thank you!