

[지난주 수업예제의 디비화]

1. board-app.zip 다운로드 후 압축 풀기

2. Docker로 운영

이미지 빌드

```
docker build -t board-app .
```

컨테이너 실행 (DB 영속화를 위해 로컬 ./db ↔ /app/db 마운트)

```
docker run -d --name board-app -p 3000:3000 -v $(pwd)/db:/app/db board-app
```

확인

<http://localhost:3000>

[홈페이지 만들고 데이터베이스 연동하여 배포하기]

1. 준비

웹서버: Apache2

DB: MySQL Server

서버 사이드: PHP 7.4+ (php-mysql)

프로젝트 경로: /var/www/html/my_website

2. 기본 명령

```
sudo apt update
```

```
sudo apt install -y apache2
```

```
sudo systemctl status apache2
```

```
sudo mkdir -p /var/www/html/my_website
```

```
sudo chown -R $USER:$USER /var/www/html/my_website
```

```
cd /var/www/html/my_website
```

3. html과 css 작성

(1) index.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <title>홈 | My Website</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <header>
        <h1>My Simple Homepage</h1>
        <nav>
            <a href="index.html">홈</a>
            <a href="about.html">자기소개</a>
            <a href="projects.html">프로젝트</a>
            <a href="board.html">게시판</a>
        </nav>
    </header>

    <main>
        <h2>환영합니다!</h2>
        <p>이곳은 HTML과 CSS로 만든 간단한 홈페이지입니다.</p>
    </main>

    <footer>
        <p>&copy; 2025 My Website</p>
    </footer>
</body>
</html>
```

(2) about.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>자기소개 | My Website</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    <h1>자기소개</h1>
    <nav>
      <a href="index.html">홈</a>
      <a href="about.html">자기소개</a>
      <a href="projects.html">프로젝트</a>
      <a href="board.html">게시판</a>
    </nav>
  </header>

  <main>
    <h2>안녕하세요!</h2>
    <p>저는 웹 개발과 교육을 좋아하는 사람입니다. HTML, CSS, JS를 활용해 다양한 실습용 페이지를 제작하고 있습니다.</p>
  </main>

  <footer>
    <p>&copy; 2025 My Website</p>
  </footer>
</body>
</html>
```

(3) projects.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>프로젝트 | My Website</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    <h1>프로젝트</h1>
    <nav>
      <a href="index.html">홈</a>
      <a href="about.html">자기소개</a>
      <a href="projects.html">프로젝트</a>
      <a href="board.html">게시판</a>
    </nav>
  </header>

  <main>
    <h2>진행한 프로젝트 목록</h2>
    <ul>
      <li> 시스템 프로그래밍 강의 자료 웹화</li>
      <li> AI PPT 자동 생성기</li>
      <li> 포트폴리오 사이트 제작</li>
    </ul>
  </main>

  <footer>
    <p>&copy; 2025 My Website</p>
  </footer>
</body>
</html>
```

(4) board.html

```
<!DOCTYPE html>
<html lang="ko">
<head>
  <meta charset="UTF-8">
  <title>게시판 | My Website</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <header>
    <h1>게시판</h1>
    <nav>
      <a href="index.html">홈</a>
      <a href="about.html">자기소개</a>
      <a href="projects.html">프로젝트</a>
      <a href="board.html">게시판</a>
    </nav>
  </header>

  <main>
    <h2>간단한 게시판</h2>
    <p>이곳에 방문 소감을 남겨주세요.</p>

    <form>
      <label>이름: <input type="text" name="name"></label><br>
      <label>내용: <textarea name="content"></textarea></label><br>
      <button type="submit">등록</button>
    </form>
  </main>

  <footer>
    <p>&copy; 2025 My Website</p>
  </footer>
</body>
</html>
```

(5) style.css

```
body {  
    font-family: "Noto Sans KR", sans-serif;  
    margin: 0;  
    padding: 0;  
    background-color: #f9f9f9;  
}  
  
header {  
    background-color: #004c99;  
    color: white;  
    padding: 20px;  
    text-align: center;  
}  
  
nav {  
    margin-top: 10px;  
}  
  
nav a {  
    color: white;  
    margin: 0 15px;  
    text-decoration: none;  
    font-weight: bold;  
}  
  
nav a:hover {  
    text-decoration: underline;  
}  
  
main {  
    padding: 40px;  
    text-align: center;  
}  
  
footer {  
    background-color: #eee;  
    text-align: center;  
    padding: 10px;  
    position: fixed;  
    width: 100%;  
    bottom: 0;  
}
```

4. 확인

http://localhost



5. MySQL 서버

MySQL 서버 설치

```
sudo apt install -y mysql-server
```

PHP + MySQL 연동 모듈

```
sudo apt install -y php libapache2-mod-php php-mysql
```

방화벽 (UFW 사용 중이라면)

```
sudo ufw allow 'Apache Full'
```

<확인>

```
systemctl status apache2      # active (running) 이어야 정상  
php -v                      # PHP 버전 확인
```

6. DB 만들기

(1) 보안스크립트

```
sudo mysql_secure_installation
```

- 질문의 답

VALIDATE PASSWORD? → 개발용이면 N(선택)

새 root 패스워드 설정

익명 사용자 제거(Y), 원격 root 차단(Y), 테스트 DB 제거(Y), 권한 새로고침(Y)

(2) MySQL 접속

```
sudo mysql -u root -p
```

비번 : 1234

(3) DB 계정과 테이블

```
-- DB와 전용 계정 만들기 (원하는 이름으로 변경 가능)
CREATE DATABASE mysite CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
CREATE USER 'mysiteuser'@'localhost' IDENTIFIED BY 'StrongPassw0rd!';
GRANT ALL PRIVILEGES ON mysite.* TO 'mysiteuser'@'localhost';
FLUSH PRIVILEGES;

-- 게시판 테이블
USE mysite;
CREATE TABLE posts (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    title VARCHAR(200) NOT NULL,
    content TEXT NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

exit

(4) PHP 타임존/문자셋 기본 세팅(권장)

sudo nano /etc/php/*apache2/php.ini

date.timezone = Asia/Seoul로 설정

저장 후 Apache 재시작:

sudo systemctl restart apache2

(5) 프로젝트 폴더 만들기 & 권한

sudo mkdir -p /var/www/html/my_website

sudo chown -R \$USER:www-data /var/www/html/my_website

find /var/www/html/my_website -type d -exec chmod 775 {} \;

find /var/www/html/my_website -type f -exec chmod 664 {} \;

(6) 정적 페이지 수정

- 모든 html의 메뉴

<nav>

홈

자기소개

프로젝트

게시판 <!-- 여기만 변경 -->

</nav>

(7) DB 연결 파일 작성

7-1) /var/www/html/my_website/db.php

```
<?php
// db.php : 모든 PHP 페이지에서 include하여 DB 커넥션 재사용
$host = 'localhost';
$db   = 'mysite';
$user = 'mysiteuser';
$pass = 'StrongPassw0rd!';
$charset = 'utf8mb4';

$dsn = "mysql:host=$host;dbname=$db;charset=$charset";
$options = [
    PDO::ATTR_ERRMODE            => PDO::ERRMODE_EXCEPTION,
    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
    PDO::ATTR_EMULATE_PREPARES   => false,
];

try {
    $pdo = new PDO($dsn, $user, $pass, $options);
} catch (PDOException $e) {
    exit('DB 연결 실패: ' . htmlspecialchars($e->getMessage(), ENT_QUOTES, 'UTF-8'));
}
```

7-2) 목록 + 글쓰기 버튼 (board.php)

```
<?php require_once __DIR__ . '/db.php'; ?>
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <title>게시판 | My Website</title>
    <link rel="stylesheet" href="style.css">
    <style>
        .board { max-width: 900px; margin: 40px auto; }
        table { width: 100%; border-collapse: collapse; }
        th, td { padding: 10px; border-bottom: 1px solid #ddd; text-align: left; }
        .actions { text-align: right; margin: 20px 0; }
        .btn { padding: 8px 14px; border: 1px solid #004c99; background:#004c99; color:#fff; text-decoration:none; border-radius:8px; }
        .btn-outline { background:#fff; color:#004c99; }
    </style>
</head>
```

```

<body>
<header>
    <h1>게시판</h1>
    <nav>
        <a href="index.html">홈</a>
        <a href="about.html">자기소개</a>
        <a href="projects.html">프로젝트</a>
        <a href="board.php">게시판</a>
    </nav>
</header>
<main class="board">
    <div class="actions">
        <a class="btn" href="write.php">글쓰기</a>
    </div>
    <table>
        <thead>
            <tr><th>번호</th><th>제목</th><th>작성자</th><th>작성일</th></tr>
        </thead>
        <tbody>
            <?php
                $stmt = $pdo->query("SELECT id, title, name, created_at FROM posts ORDER BY id DESC");
                foreach ($stmt as $row):
            ?>
            <tr>
                <td><?= (int)$row['id'] ?></td>
                <td><a href="view.php?id=<?= (int)$row['id'] ?>"><?= htmlspecialchars($row['title']) ?></a></td>
                <td><?= htmlspecialchars($row['name']) ?></td>
                <td><?= htmlspecialchars($row['created_at']) ?></td>
            </tr>
            <?php endforeach; ?>
        </tbody>
    </table>
</main>
<footer>
    <p>&copy; 2025 My Website</p>
</footer>
</body>
</html>

```

7-3) 글쓰기 폼 (write.php)

```
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <title>글쓰기 | My Website</title>
    <link rel="stylesheet" href="style.css">
    <style>
        .board { max-width: 700px; margin: 40px auto; }
        label { display: block; margin: 12px 0 6px; }
        input[type="text"], textarea { width: 100%; padding: 10px; border: 1px solid #ccc; border-radius: 8px; }
        .actions { margin-top: 16px; }
        .btn { padding: 8px 14px; border: 1px solid #004c99; background: #004c99; color: #fff; text-decoration: none; border-radius: 8px; }
        .btn-outline { background: #fff; color: #004c99; }
    </style>
</head>
<body>
<header>
    <h1>글쓰기</h1>
    <nav>
        <a href="index.html">홈</a>
        <a href="about.html">자기소개</a>
        <a href="projects.html">프로젝트</a>
        <a href="board.php">게시판</a>
    </nav>
</header>
<main class="board">
    <form action="save.php" method="post">
        <label for="name">이름</label>
        <input id="name" name="name" type="text" required maxlength="50">
        <label for="title">제목</label>
        <input id="title" name="title" type="text" required maxlength="200">
        <label for="content">내용</label>
        <textarea id="content" name="content" rows="10" required></textarea>

        <div class="actions">
            <button class="btn" type="submit">등록</button>
            <a class="btn btn-outline" href="board.php">목록</a>
        </div>
    </form>
</main>
<footer>
    <p>© 2025 My Website</p>
</footer>
</body>
</html>
```

7-4) 글 저장 (save.php)

```
<?php
require_once __DIR__ . '/db.php';

$name    = trim($_POST['name'] ?? '');
$title   = trim($_POST['title'] ?? '');
$content = trim($_POST['content'] ?? '');

if ($name === '' || $title === '' || $content === '') {
    exit('필수 항목이 비었습니다. <a href="write.php">뒤로</a>');
}

$stmt = $pdo->prepare("INSERT INTO posts (name, title, content) VALUES (?, ?, ?)");
$stmt->execute([$name, $title, $content]);

header('Location: board.php');
exit;
```

7-5) 글 상세 (view.php)

```
<?php
require_once __DIR__ . '/db.php';
$id = (int)($_GET['id'] ?? 0);

$stmt = $pdo->prepare("SELECT * FROM posts WHERE id = ?");
$stmt->execute([$id]);
$post = $stmt->fetch();

if (!$post) { exit('글을 찾을 수 없습니다. <a href="board.php">목록</a>'); }
?>
<!DOCTYPE html>
<html lang="ko">
<head>
    <meta charset="UTF-8">
    <title><?= htmlspecialchars($post['title']) ?> | My Website</title>
    <link rel="stylesheet" href="style.css">
    <style>
        .board { max-width: 900px; margin: 40px auto; }
        .meta { color:#666; margin-bottom:10px; }
        pre { white-space: pre-wrap; word-break: break-word; }
        .actions { margin-top: 16px; }
        .btn { padding: 8px 14px; border:1px solid #004c99; background:#004c99; color:#fff; text-decoration:none; border-radius:8px;}
        .btn-outline { background:#fff; color:#004c99; }
    </style>
</head>
```

```

<body>
<header>
    <h1>게시글 보기</h1>
    <nav>
        <a href="index.html">홈</a>
        <a href="about.html">자기소개</a>
        <a href="projects.html">프로젝트</a>
        <a href="board.php">게시판</a>
    </nav>
</header>
<main class="board">
    <h2><?= htmlspecialchars($post['title']) ?></h2>
    <div class="meta">
        작성자: <?= htmlspecialchars($post['name']) ?> | 작성일: <?=
        htmlspecialchars($post['created_at']) ?>
    </div>
    <pre><?= htmlspecialchars($post['content']) ?></pre>

    <div class="actions">
        <a class="btn btn-outline" href="board.php">목록</a>
        <a class="btn" href="delete.php?id=<?= (int)$post['id'] ?>" onclick="return confirm('삭제하시겠습니까?');">삭제</a>
    </div>
</main>
<footer>
    <p>© 2025 My Website</p>
</footer>
</body>
</html>

```

7-6) 글 삭제 (delete.php)

```

<?php
require_once __DIR__ . '/db.php';
$id = (int)($_GET['id'] ?? 0);

if ($id > 0) {
    $stmt = $pdo->prepare("DELETE FROM posts WHERE id = ?");
    $stmt->execute([$id]);
}

header('Location: board.php');
exit;

```

(8) 동작 테스트

브라우저에서 http://서버IP/my_website/board.php 열기

“글쓰기”로 글 등록 → 목록/상세에 노출되면 성공

문제 발생 시 Apache 에러로그 확인:

`sudo tail -f /var/log/apache2/error.log`

(9) Docker 이미지 만들기

[전체구조]

```
my_website_docker/
├── docker-compose.yml
├── .env
├── web/
│   ├── Dockerfile
│   └── php.ini
└── db/
    └── init.sql
└── app/           ← 실제 웹 소스(HTML/CSS/PHP)
    ├── index.html
    ├── about.html
    ├── projects.html
    ├── style.css
    ├── board.php
    ├── write.php
    ├── save.php
    ├── view.php
    ├── delete.php
    └── db.php
```

```

9-0) 사전 준비
# Docker Engine
sudo apt update
sudo apt install -y ca-certificates curl gnupg
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
echo \
"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] \
https://download.docker.com/linux/ubuntu $(. /etc/os-release && echo $UBUNTU_CODENAME)
stable" \
| sudo tee /etc/apt/sources.list.d/docker.list >/dev/null
sudo apt update
sudo apt install -y docker-ce docker-ce-cli containerd.io docker-buildx-plugin
docker-compose-plugin

# docker 명령 sudo 없이 사용(로그아웃/재로그인 필요)
sudo usermod -aG docker $USER

# 프로젝트 디렉터리 만들기
# 작업 루트(원하는 곳)로 이동
cd ~
mkdir -p my_website_docker/{app,web,db}
cd my_website_docker

# 기존 소스 복사
# 기존 소스 위치: /var/www/html/my_website
# 새 위치: ~/my_website_docker/app
cp -a /var/www/html/my_website/* ./app/

# 도커 디렉터리 이동 및 확인
cd ~/my_website_docker

```

9-1) docker-compose.yml

```
version: "3.9"
services:
  web:
    build:
      context: ./web
      dockerfile: Dockerfile
    container_name: mysite-web
    ports:
      - "8080:80"          # http://localhost:8080
    env_file: .env
    volumes:
      - ./app:/var/www/html    # 코드 실시간 반영
    depends_on:
      db:
        condition: service_healthy
  db:
    image: mysql:8.0
    container_name: mysite-db
    env_file: .env
    command: --character-set-server=utf8mb4 --collation-server=utf8mb4_general_ci
    volumes:
      - db_data:/var/lib/mysql
      - ./db/init.sql:/docker-entrypoint-initdb.d/01_init.sql:ro
    healthcheck:
      test: ["CMD-SHELL", "mysqladmin ping -h localhost -u $$MYSQL_USER -p$$MYSQL_PASSWORD --silent"]
      interval: 5s
      timeout: 3s
      retries: 20
    ports:
      - "3306:3306"          # 필요 시 외부에서 DB 접속
volumes:
  db_data:
```

9-2) .env (루트에 생성)

```
# MySQL 기본값
MYSQL_ROOT_PASSWORD=RootPassw0rd!
MYSQL_DATABASE=mysite
MYSQL_USER=mysiteuser
MYSQL_PASSWORD=StrongPassw0rd!

# PHP에서 읽어 쓸 ENV (db.php가 사용)
APP_DB_HOST=db
APP_DB_NAME=mysite
APP_DB_USER=mysiteuser
APP_DB_PASS=StrongPassw0rd!
APP_TZ=Asia/Seoul
```

9-3) web/Dockerfile

```
FROM php:8.2-apache

# PHP 확장(pdo_mysql) 설치
RUN docker-php-ext-install pdo_mysql

# Apache 유용 모듈
RUN a2enmod rewrite

# 타임존 설정 (php.ini에서 확정)
COPY php.ini /usr/local/etc/php/conf.d/99-custom.ini

# 문서 루트는 기본 /var/www/html (변경 필요 없으면 그대로)
WORKDIR /var/www/html/my_website
```

9-4) web/php.ini

```
date.timezone = ${APP_TZ}
mbstring.language = Korean
default_charset = "UTF-8"
```

9-5) db/init.sql (최초 1회 컨테이너 기동 시 자동 실행)

```
-- 오류가 나면(추가)
CREATE DATABASE IF NOT EXISTS mysite
    CHARACTER SET utf8mb4
    COLLATE utf8mb4_general_ci;

CREATE USER IF NOT EXISTS 'mysiteuser'@'%' IDENTIFIED BY 'StrongPassw0rd!';
GRANT ALL PRIVILEGES ON mysite.* TO 'mysiteuser'@'%';

FLUSH PRIVILEGES;

-- DB/사용자는 .env로 이미 생성되므로 여기서는 테이블만 준비
CREATE TABLE IF NOT EXISTS posts (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(50) NOT NULL,
    title VARCHAR(200) NOT NULL,
    content TEXT NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

10. app/ 소스 (핵심만 변경: db.php + 링크)

10-1) app/db.php (ENV를 읽도록 설정)

```
<?php
$host    = getenv('APP_DB_HOST') ?: 'db';
$db      = getenv('APP_DB_NAME') ?: 'mysite';
$user    = getenv('APP_DB_USER') ?: 'mysiteuser';
$pass    = getenv('APP_DB_PASS') ?: 'StrongPassw0rd!';
$charset = 'utf8mb4';

$dsn = "mysql:host=$host;dbname=$db;charset=$charset";
$options = [
    PDO::ATTR_ERRMODE            => PDO::ERRMODE_EXCEPTION,
    PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
    PDO::ATTR_EMULATE_PREPARES   => false,
];

try {
    $pdo = new PDO($dsn, $user, $pass, $options);
} catch (PDOException $e) {
    exit('DB 연결 실패: ' . htmlspecialchars($e->getMessage(), ENT_QUOTES, 'UTF-8'));
}
```

10-2) 네비게이션의 “게시판” 링크를 board.php로

```
<nav>
    <a href="index.html">홈</a>
    <a href="about.html">자기소개</a>
    <a href="projects.html">프로젝트</a>
    <a href="board.php">게시판</a> <!-- 여기 -->
</nav>
```

● 이미 수정한 경우는 수정 안해도 됨, 경로만 app/

11. 실행

1) 폴더로 이동

```
cd my_website_docker
```

2) (최초) 빌드 + 실행

```
docker compose up -d --build
```

3) 접속

```
# 웹: http://localhost:8080
```

- 재가동하는 경우

- DB 초기화

- docker compose down -v

- 재시작 및 빌드

- docker compose up -d --build

기본 명령 정리

현재 상태 확인 : docker compose ps

로그 확인 : docker compose logs -f

중지 / 재시작 :

docker compose down

docker compose up -d

재 빌드 : docker compose up -d --build

웹만 재시작 : docker compose restart web

DB만 재시작 : docker compose restart db