

웹서버보안프로그래밍

2. 웹개발환경 구축

중부대학교 정보보호학과

이병천 교수

sultan@joongbu.ac.kr

전체 목차

1. 강의 개요
- 2. 웹개발환경 구축**
3. Next.js 프레임워크 소개
4. 예제 프로젝트 1 : 쇼핑몰 구축
5. 예제 프로젝트 2
6. PBL 팀프로젝트 발표

2. 웹개발환경 구축



2. 개발 환경 구축

- 2.1 크롬 브라우저
- 2.2 Node.js 설치
- 2.3 VSCode 설치
 - 환경 설정,
 - 유용한 확장 패키지 설치
- 2.4 Tailwind CSS
- 2.5 Git, Github, Vercel
- 2.6 MongoDB

2.1 크롬 브라우저

- 메인 브라우저
 - 크롬을 메인 브라우저로 통일하여 사용
 - 확장 프로그램 설치 사용 필요
- 서브 브라우저
 - 엣지, 브레이브, 웨일 등
 - 개발 결과가 다른 브라우저에서 동일하게 동작하는지 점검

2.2 VSCode

- 웹에디터 VSCode 설치
 - 마이크로소프트에서 개발한 통합개발환경
 - <https://code.visualstudio.com/>
 - 현재 버전: 1.80.2 (2023.7)
 - 자동으로 업데이트
 - 웹개발자들이 가장 사랑하고 널리 사용되는 에디터
 - 쉽게 설치하여 사용할 수 있는 다양한 확장 프로그램들



VSCode

- 유용한 확장기능 설치 활용
 - Korean language pack: 한국어 UI 환경
 - Live Server: 프론트엔드 개발용 로컬 웹서버
 - Prettier: 코드 포맷 관리, 저장시 포맷 정렬
 - Auto close tag: HTML 태그 자동 완성
 - Auto Rename Tag: HTML 태그의 쌍을 자동 완성
 - Indent-rainbow: 들여쓰기를 색깔로 구별 표시
 - ES7+React/Redux/React-native snippets: React 사용시 필수
 - Tailwind: Tailwind CSS 사용시 클래스 정보 표시
 - :emojisense: 이모티콘 사용
 - vscode-icons: 파일별 아이콘 사용
- * VSCode 유용한 확장기능 검색

VSCode


- 유용한 설정
 - Default formatter: Prettier 적용
 - Format on save: 파일 저장시 자동 포맷 정렬
 - Word wrap: 한 줄이 넘는 문장은 다음 줄로 넘김
 - Tab size: 2
 - Prettier: single quote 체크 (자바스크립트 문장에서는 단일 따옴표 사용)
 - Prettier: Jsx single quote 체크 안함 (Jsx 문장에서는 이중 따옴표 사용)
 - Prettier: Semi 체크 안함 (자바스크립트 문장의 끝에 세미콜론을 뺌)
- 내장 명령창 활용
 - Ctrl+`
 - 여러 개의 명령창 열기

2.3 Node.js

- Node.js 설치
 - 자바스크립트 프로그램 별도 실행 환경
 - 강의 실습에서 사용하는 다양한 패키지들을 설치하여 사용하는데 필요
 - <https://nodejs.org/ko>
 - NPM (노드 패키지 관리자)가 함께 설치됨
 - LTS (long-term stable) 버전으로 설치.
 - 기본 설치 위치: C:\Program Files\nodejs
- 설치 후 버전 확인
 - > node -v (Node.js 프로그램 실행 명령어)
 - > npm -v (노드 패키지 관리자)

Run JavaScript Everywhere

Node.js® is a free, open-source, cross-platform JavaScript runtime environment that lets developers create servers, web apps, command line tools and scripts.

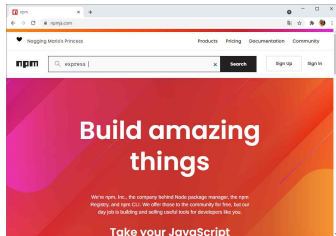
Download Node.js (LTS) 

Downloads Node.js **v20.17.0**¹ with long-term support.
Node.js can also be installed via [package managers](#).

Want new features sooner? Get **Node.js v22.7.0**¹ instead.

Node.js와 NPM

- NPM (Node Package Manager)
 - Node.js에서는 많은 확장 패키지들을 설치하여 사용할 수 있음.
 - NPM은 노드 패키지 관리자로서 패키지 관리가 매우 편리함. 노드를 유명하게 만든 강력한 기능.
 - Node.js 설치시 NPM도 기본 설치됨
 - 자신이 프로그래밍한 코드를 패키지 형태로 만들어 재활용할 수 있음
 - <https://www.npmjs.com/> 참조



2.4 Tailwind CSS 활용

- Tailwind CSS



- <https://tailwindcss.com/>
- A utility-first CSS framework
- 미리 정의된 클래스 이름을 HTML 파일에서 지정하기만 하면 디자인이 바로 적용됨
- 사용 매뉴얼 <https://tailwindcss.com/docs/installation>
- VSCode에서 [Tailwind CSS IntelliSense](#) plugin 설치

**Rapidly build modern websites
without ever leaving your HTML.**

A utility-first CSS framework packed with classes like `flex`, `pt-4`, `text-center` and `rotate-90` that can be composed to build any design, directly in your markup.

Get started



Quick search...

Ctrl K

Tailwind CSS

- **Install Tailwind CSS with Next.js**
 - <https://tailwindcss.com/docs/guides/nextjs>
- **Install Tailwind CSS with Create React App**
 - <https://tailwindcss.com/docs/guides/create-react-app>
- Next.js 앱을 생성할때 Tailwind CSS를 사용할 것인지 질문함

2.5 Git and Github

- Git
 - source 관리를 위한 분산 버전 관리 시스템
 - 최초로 리눅스 토발즈가 리눅스 커널 개발에 이용하려고 개발
 - 프로그램 다운로드 및 설치 <https://git-scm.com/downloads>
 - 현재 버전: 2.46.0 (2024.9.2)
 - Git GUI clients



GIT 작업 흐름과 명령어

도움말: git "명령어" --help

Git의 글로벌 설정은 \$HOME/.gitconfig에 저장 (git config --help)

생성

새 저장소 생성하기

```
cd ~/projects/myproject
git init
git add .
```

기존 저장소 Clone하기

```
git clone ~/existing/repo ~/new/repo
git clone git://host.org/project.git
git clone you@host.org/project.git
```

보기

워킹 디렉터리의 파일 상태 보기

```
git status
```

파일의 변경사항 보기

```
git diff
```

\$ID1과 \$ID2 사이와 변경사항 보기

```
git diff $id1 $id2
```

커밋 히스토리 보기

```
git log
```

특정 파일의 커밋 히스토리를 변경사항 보기

```
git log -p $file $dir/ec/tory/
```

특정 파일을 누가 언제 고쳤는지 보기

```
git blame $file
```

\$ID 커밋 보기

```
git show id
```

\$ID 버전의 파일 보기

```
git show $id $file
```

로컬 브랜치들 보기

```
git branch
/* 표시는 현재 브랜치를 나타냄
```

범례

\$id - 커밋 ID, 브랜치 이름, 태그 이름을 나타냄
\$file - 파일 이름
\$branch - 브랜치 이름

개념

Git 기본

master : 기본 브랜치
origin : 기본 리모트 저장소
HEAD : 현재 브랜치
HEAD^ : HEAD의 부모
HEAD~4 : HEAD의 부모의 부모의 부모의 부모

되돌림

마지막 커밋 사항으로 되돌리기

```
git reset --hard
/* Hard Reset은 되돌릴 수 없음
```

마지막 커밋 내용을 되돌리고 커밋하기

```
git revert HEAD
/* 새로운 커밋 생성
```

특정 커밋 내용을 되돌리고 커밋하기

```
git revert $id
/* 새로운 커밋 생성
```

마지막 커밋 수정하기

```
git commit -a --amend
/* 앞의 커밋에서 수정하고 싶을 때
```

\$id 시점의 파일을 꺼내기

```
git checkout $id $file
```

브랜치

브랜치를 Checkout하기

```
git checkout $id
```

\$branch1을 \$branch2에 Merge하기

```
git checkout $branch2
git merge $branch1
```

새 브랜치 만들기

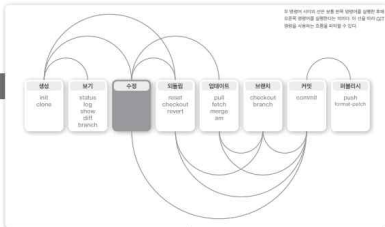
```
git branch $branch
```

\$other와 같은 커밋을 가리키는 브랜치를 새로 만들고 바로 Checkout하기

```
git checkout -b $new_branch $other
```

\$branch를 삭제하기

```
git branch -d $branch
```



도움말에 서두지 않고 보통 한두 단락을 읽어야 할 때
이전 버전의 명령어를 실행한다는 의미로, 이 단락을 주의 깊게
읽어야 하는 이유는 여러가지가 있다

Git 명령어 익히기

업데이트

origin에서 최신 데이터를 가져오기

```
git fetch
/* Merge하지는 않음
```

origin에서 최신 데이터를 가져와 Merge하기

```
git pull
/* fetch하고 Merge까지 함
```

누군가 보낸 패치를 Merge하기

```
git am -3 patch-mbox
/* 패치가 포함된 mbox 후 git am --resolved
```

퍼블리시

현재 모든 수정사항을 커밋하기

```
git commit -a
```

다른 개발자에게 보낼 패치를 작성하기

```
git format-patch origin
```

origin으로 업데이트를 Push하기

```
git push
```

버전이나 마일스톤을 생성하기

```
git tag v1.0
```

유용한 명령어

문제가 발생한 커밋 이전 상태로 되돌리기

```
git bisect start
/* git bisect good $id (이전 패치 있는 상태로 표시)
git bisect bad $id (이전 패치 있는 상태로 표시)
```

```
git bisect bad/good (현재 상태가 문제가 있는지 있는지 확인)
git bisect visualize (이전 상태를 시각화하여 확인)
git bisect reset (이전 상태로 Checkout 함)
```

저장소의 무결성 검사 및 저장소 청소하기

```
git fsck
git gc --prune
```

워킹 디렉터리에서 'foo!'라는 문자열 검색하기

```
git grep "foo!"
```

Merge 충돌 해결

Merge 충돌 내용 보기

```
git diff (충돌 내용 전체 보기)
git diff --base $file (Merge Base를 기준으로)
git diff --ours $file (본 브랜치를 기준으로)
git diff --theirs $file (Merge 대상 브랜치를 기준으로)
```

충돌이 생기는 패치 버리기


```
git reset --hard
git rebase --skip
```

충돌 해결 후 Merge 진행하기

```
git add $conflicting_file
git rebase --continue
```

Github

- Github
 - 소스 관리, 버전 관리, 프로젝트 협업 포털 서비스
 - <https://github.com/>
 - 회원가입 (sign up)
 - 로그인 (sign in)
 - 자신의 포트폴리오 페이지로 관리



Sign in to GitHub

Username or email address

Password [Forgot password?](#)

[Sign in](#)

New to GitHub? [Create an account](#).

Github

- 새로운 저장소(repository) 생성하기
 - Repository name
 - Public / Private
 - 사용 안내문이 나옴

```
echo "# test1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/lbcsultan/test1.git
git push -u origin main
```


Git 명령어 흐름

- 0. 프로젝트 로컬 폴더로 이동
- 1. 저장소 초기화
 - > git init
- 2. 현재 폴더의 모든 파일들을 git 저장소에 추가
 - > git add .
 - .gitignore에 선언된 폴더, 파일들은 제외
- 3. 커밋, 전송 준비
 - > git commit -am 'first'
 - 구별할 수 있는 메시지 추가
- 4. 원격 저장소 지정
 - > git remote
- 5. 푸시, 전송
 - > git push, 실제 코드를 github로 업로드

Vscode에서 github 계정정보 설정

```
> git config --global user.email "이메일주소"
```

```
> git config --global user.name "계정이름"
```

한번 계정정보를 설정하고 로그인하면

윈도우의 자격 증명 관리자에

인증정보가 저장되며 인증이 유지됨

계정정보 설정을 다시 입력할 필요 없음

2.6 MongoDB 활용

- **클라우드 DB 서비스 활용**

- Atlas: MongoDB as a Service
- <https://www.mongodb.com/>
- 회원가입 및 활용 방법 안내
- 팀프로젝트 협업을 위해서는 온라인 클라우드 DB 사용이 필요



mongoDB®

- **DB 서버 로컬 운영**

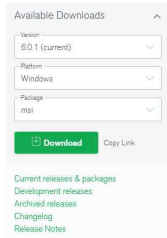
- MongoDB community server 설치
- <https://www.mongodb.com/try/download/community>

- **GUI 클라이언트 프로그램 활용**

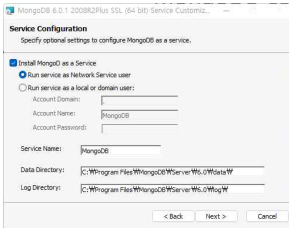
- MongoDB Compass 설치
- <https://www.mongodb.com/products/compass>
- MongoDB 활용을 위한 GUI 프로그램

MongoDB community server

- 서버 다운로드 설치
 - <https://www.mongodb.com/try/download/community>

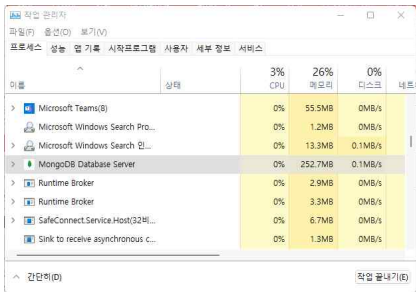


서버 설치 위치

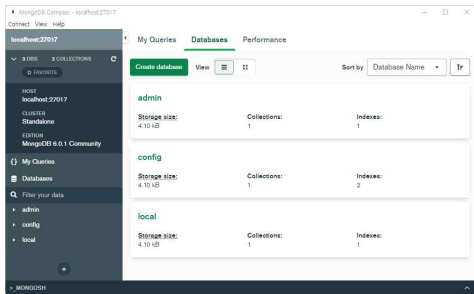


Compass 설치 옵션

Compass 로 서버 연결



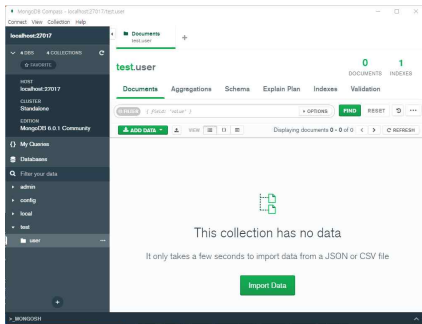
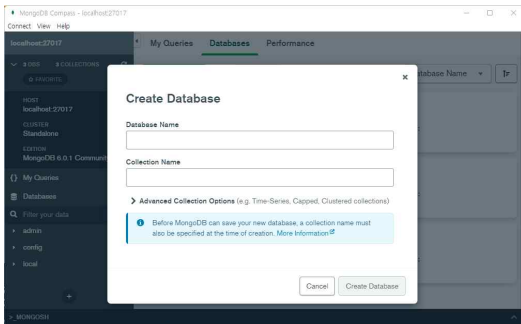
백그라운드로 서버 동작중
부팅시 서버 자동 실행



Compass로 서버 연결

로컬서버 연결 주소
mongodb://localhost:27017

Database, collection 만들기



MongoDB 클라우드 서비스 활용

- MongoDB Atlas
 - <https://www.mongodb.com/atlas>
 - Sign Up (회원가입)
 - Google, Github, Email
 - Sign In (로그인)
 - <https://account.mongodb.com/account/login>



Log in to your account



or

Email Address

Next

Don't have an account? [Sign Up](#)

MongoDB 클라우드 서비스 활용

- 초기 설정
 - Organization명, project명 - 자동 생성됨, 신규 생성 가능
 - Cluster명 선택: 클라우드 서비스, 지역 등, Free tier 선택
 - Free tier는 cluster명을 하나만 선택 가능
 - 관리자 계정 생성 필요
 - 초기 클러스터 생성 및 설정에 시간이 걸림. 약간 기다린 후 대시보드가 로드됨.
 - 최초의 DB, collection 생성
 - 완성된 대시보드 로드
 - Database Access: 생성한 DB를 사용할 수 있는 계정을 생성, 여러 개 생성 가능. 읽기/쓰기 권한 부여 필요
 - Network Access: 네트워크 접근제어 설정. 무선 네트워크를 이용하는 노트북의 경우 어디서나 접근 가능하도록 0.0.0.0/0 으로 설정

초기 클러스터 생성

EC'S ORD - 2022-09-07 > PROJECT 0

Database Deployments

Find a database deployment...



Create a database

Choose your cloud provider, region, and specs.

Build a Database

Once your database is up and running, live migrate an existing MongoDB database into Atlas with our [Live Migration Service](#).



Deploy a cloud database

Experience the best of MongoDB on AWS, Azure, and Google Cloud. Choose a deployment option to get started.

NEW

Serverless

For application development and testing, in workloads with variable traffic, minimal configuration required.

- ✓ Pay only for the operations you run
- ✓ Resources scale seamlessly to meet your workload
- ✓ Always-on security and backup

Create

Starting at
\$0.10/1M reads

ADVANCED

Dedicated

For production applications with sophisticated workload requirements. Advanced configuration controls.

- ✓ Network isolation and fine-grained access controls
- ✓ On-demand performance advice
- ✓ Multi-region and multi-cloud options available

Create

Minimum of
\$0.08/hr*
*Estimated cost of \$0.04/hr/GB

FREE

Shared

For learning and exploring MongoDB in a cloud environment. Basic configuration options.

- ✓ No credit card required to start
- ✓ Explore with sample datasets
- ✓ Upgrade to dedicated clusters for full functionality

Create

Starting at
FREE

Cluster Tier

M0 Sandbox (Shared RAM, 512 MB Storage)

Encrypted

Additional Settings

MongoDB 5.0, No Backup

Cluster Name

Cluster0

FREE

Free forever! Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

[Back](#)

Create Cluster

관리자 계정 생성, IP 접근제어 설정

How would you like to authenticate your connection?

Your first user will have permission to read and write any data in your project.

Username and Password

Certificate

Create a database user using a username and password. Users will be given the *read and write* to any database *privilege* by default. You can update these permissions and/or create additional users later. Ensure these credentials are different to your MongoDB Cloud username and password. You can manage existing users via the [Database Access Page](#).

Username

Enter username

Password

Enter password

Autogenerate Secure Password

Copy

Create User

관리자 계정 생성
일반 사용자 계정 생성

- 사용하는 특정 DB에 대한 읽기/쓰기 권한 부여 필요

Where would you like to connect from?

Enable access for any network(s) that need to read and write data to your cluster.

My Local Environment

Use this to add network IP addresses to the IP Access List. This can be modified at any time.

Cloud Environment

Use this to configure network access between Atlas and your cloud or on-premise environment. Specifically, set up IP Access Lists, Network Peering, and Private Endpoints.

ADVANCED

Add entries to your IP Access List

Only an IP address you add to your Access List will be able to connect to your project's clusters. You can manage existing IP entries via the [Network Access Page](#).

IP Address

Enter IP Address

Description

Enter description

Add Entry

Add My Current IP Address

IP Access List

Description

0.0.0.0/0

REMOVE

IP 접근제어 설정

노트북으로 IP가 자주 바뀌는 경우
0.0.0.0/0 로 어디에서나 접근 가능하도록 설정

최초 데이터베이스 생성



Explore Your Data

- **Find:** run queries and interact with documents
- **Indexes:** build and manage indexes
- **Aggregation:** test aggregation pipelines
- **Search:** build search indexes

Load a Sample Dataset

Add My Own Data

[Learn more in Docs and Tutorials](#)

Create Database

Database name ⓘ

Collection name ⓘ

Additional Preferences

- ☐ Capped Collection ⓘ
- ☐ Time Series Collection ⓘ

Cancel

Create

로그인 후 대시보드

- Cluster
- Organization
- Project

- Database
- Collection

organization
project

database

계정정보 관리
접근제어

The screenshot shows the MongoDB Atlas dashboard. The top navigation bar includes 'Lecture', 'Access Manager', and 'Billing'. The left sidebar has sections for 'DEPLOYMENT' (Database, Data Lake, PREVIEW), 'DATA SERVICES' (Triggers, Data API, Data Federation, Atlas Search), and 'SECURITY' (Quickstart, Database Access, Network Access, Advanced). The main content area is titled 'LECTURE > NEXTJS > DATABASES' and shows 'ClusterO'. Below this, there are tabs for 'Overview', 'Real Time', 'Metrics', 'Collections', 'Search', and 'Prof'. The 'Collections' tab is active, showing 'DATABASES: 9' and 'COLLECTIONS: 16'. A '+ Create Database' button is visible. Below it is a search bar 'Search Namespaces'. A list of namespaces is shown, including 'forge', 'mern-tutorial', 'users', 'mern-tutorial-bt', 'myFirstDatabase', 'next-tailwind-amazona', and 'nextauth'. A blue arrow points from the text '새로운 DB 생성' to the '+ Create Database' button.

new DB creation

클러스터 환경 설정, 클러스터 삭제

BC'S ORG - 2022-09-07 > PROJECT 0

Database Deployments

Find a database deployment...

Cluster0

Connect

View Monitoring

Browse Collections

...

Enhance Your Experience

For production throughput and richer metrics, upgrade to a dedicated cluster now!

Upgrade

R 0

W 0

Last 6 minutes

100.0/s

Edit Configuration

Command Line Tools

Load Sample Dataset

Terminate

MongoDB와 Application의 연결

Connect to Cluster0

✓ Setup connection security Choose a connection method Connect

Choose a connection method [View documentation](#)

Get your pre-formatted connection string by selecting your tool below.



Connect with the MongoDB Shell

Interact with your cluster using MongoDB's Interactive Javascript Interface



Connect your application

Connect your application to your cluster using MongoDB's native drivers



Connect using MongoDB Compass

Explore, modify, and visualize your data with MongoDB's GUI



Connect using VS Code

Connect to a MongoDB host in Visual Studio Code



Go Back

Close

Connect to Cluster0

✓ Setup connection security ✓ Choose a connection method Connect

1 Select your driver and version

DRIVER

Node.js

VERSION

4.1 or later

2 Add your connection string into your application code

☐ Include full driver code example

```
mongodb+srv://<username>:<password>@cluster0.ayx6x.mongodb.net/?  
retryWrites=true&w=majority
```



Replace **<password>** with the password for the **<username>** user. Ensure any option params are [URL encoded](#).

Having trouble connecting? [View our troubleshooting documentation](#)

Go Back

Close

자세한 활용법은 예제 프로젝트 개발 진행시 설명...

HTML, CSS, Javascript 예제

- <https://hcjdemo.netlify.app/>
- <https://github.com/lbcsultan/hcjdemo>

