

K-Digital Training KDT 풀스택 웹 개발자 양성 부트캠프 3기

Full Stack Mini Project

지금까지 웹에 대해 학습하느라 고생하셨습니다! 🙌🙌🙌

😎 그동안 배운 것을 토대로 **나만의 Todo App**을 만들어 볼까요?

WITH 팀 리쳐드

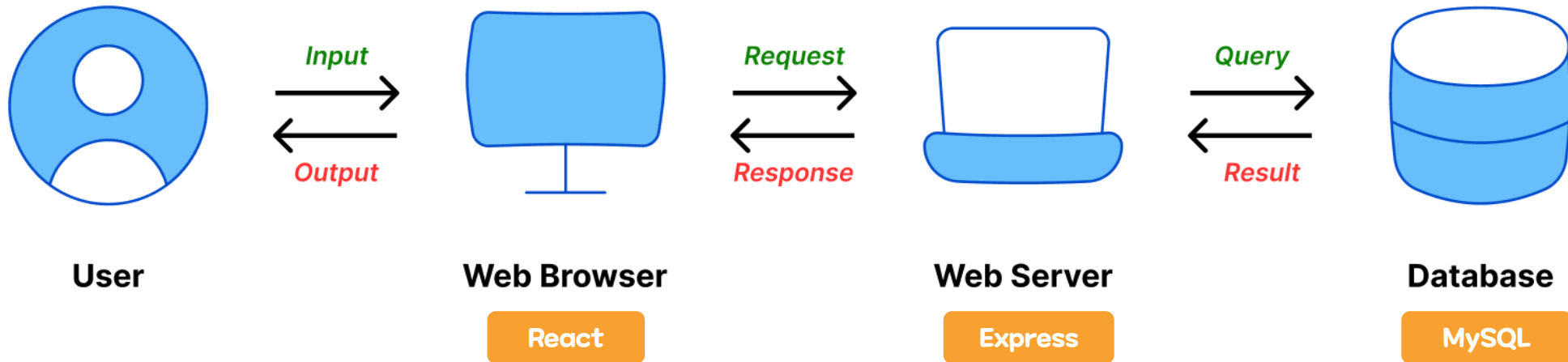


프로젝트 소개

- Node.js express와 React 로 만드는 Full Stack Mini Project
- 나만의 **Todo Application**을 만들어보자!

Frontend

Backend



프로젝트 개발 순서

- ① 백엔드 개발

- Node.js express 프레임워크와 MySQL 사용해 백엔드 애플리케이션 개발

- ② 프론트엔드 개발

- React 라이브러리를 이용해 프론트엔드 애플리케이션 개발

- ③ 서비스 통합

- 독립적으로 동작하는 백엔드 애플리케이션과 프론트엔드 애플리케이션을 하나의 서비스로 통합

프로젝트 생성

- todo-app 프로젝트 폴더 구조

```
client/ # react app  
server/ # express app  
.gitignore
```

백엔드 개발

Part1.

Node.js **express** 프레임워크와 MySQL 사용해 백엔드 애플리케이션을 만들어보자!

1. 데이터베이스 테이블 생성

- 테이블 구조

```
mysql> desc todo;
+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+
| id     | int           | NO   | PRI | NULL    | auto_increment |
| title  | varchar(100)  | NO   |     | NULL    |                |
| done   | tinyint(1)    | NO   |     | 0       |                |
+-----+-----+-----+-----+-----+
3 rows in set (0.01 sec)
```

2. REST API 구조

- REST API

- `GET /todos` - show all todos
- `POST /todo` - create a new todo
- `PATCH /todo/:todoId` - edit a specific todo
- `DELETE /todo/:todoId` - remove a specific todo

Postman

- REST API 테스트 툴
 - 브라우저 만으로 REST API 테스트 하는 데에는 한계가 있음
 - REST API 테스트를 위해 임시 프론트엔드 UI 만드는 것은 좋지 않음
 - => GUI를 제공하는 테스트 툴을 이용하자!



POSTMAN

<https://www.postman.com/downloads/>

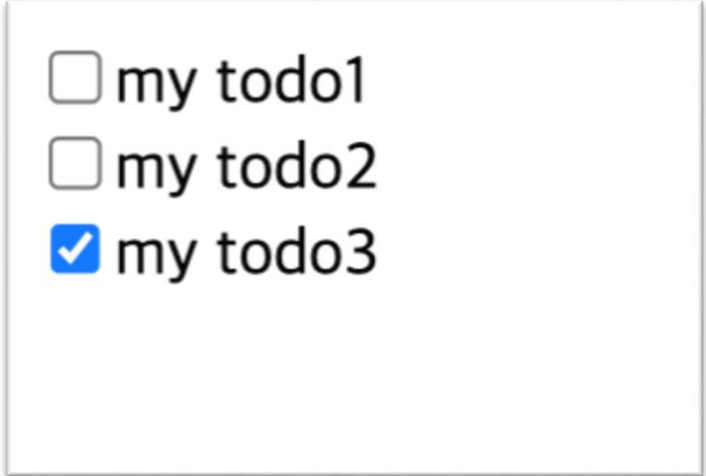
프론트엔드 개발

Part2.

React 라이브러리를 이용해 프론트엔드 애플리케이션을 만들어보자!

1. Todo 리스트

- **Todo 컴포넌트**를 만들어보자!
- checkbox 와 label을 렌더링하는 컴포넌트!
- 결과물 미리보기



- ☐ my todo1
- ☐ my todo2
- ☒ my todo3

2. Todo 추가

- Todo 추가할 UI를 만들어보자!
- input에 title 입력 후 버튼 클릭 시 할 일 추가하기
- input에 title 입력 후 Enter 키 누르면 할 일 추가하기
- 결과물 미리보기



Add your new Todo

- ☐ my todo1
- ☐ my todo2
- ☒ my todo3
- ☐ hello
- ☐ Yeah 🤘

3. Todo 삭제

- 각 Todo에 **삭제 버튼 추가**하자!
- 삭제 버튼 클릭시 아이템 삭제하기
- 결과물 미리보기

☐ my todo1

☒ my todo3

☐ awesome!

4. Todo 수정

- Todo가 수정되는 경우는?
 - 체크 박스에 체크 → done 값 변경
 - 타이틀을 변경 → title 부분 클릭시 자동으로 수정할 수 있는 상태가 되고 Enter 키 입력 시 내용 저장

• 결과물 미리보기

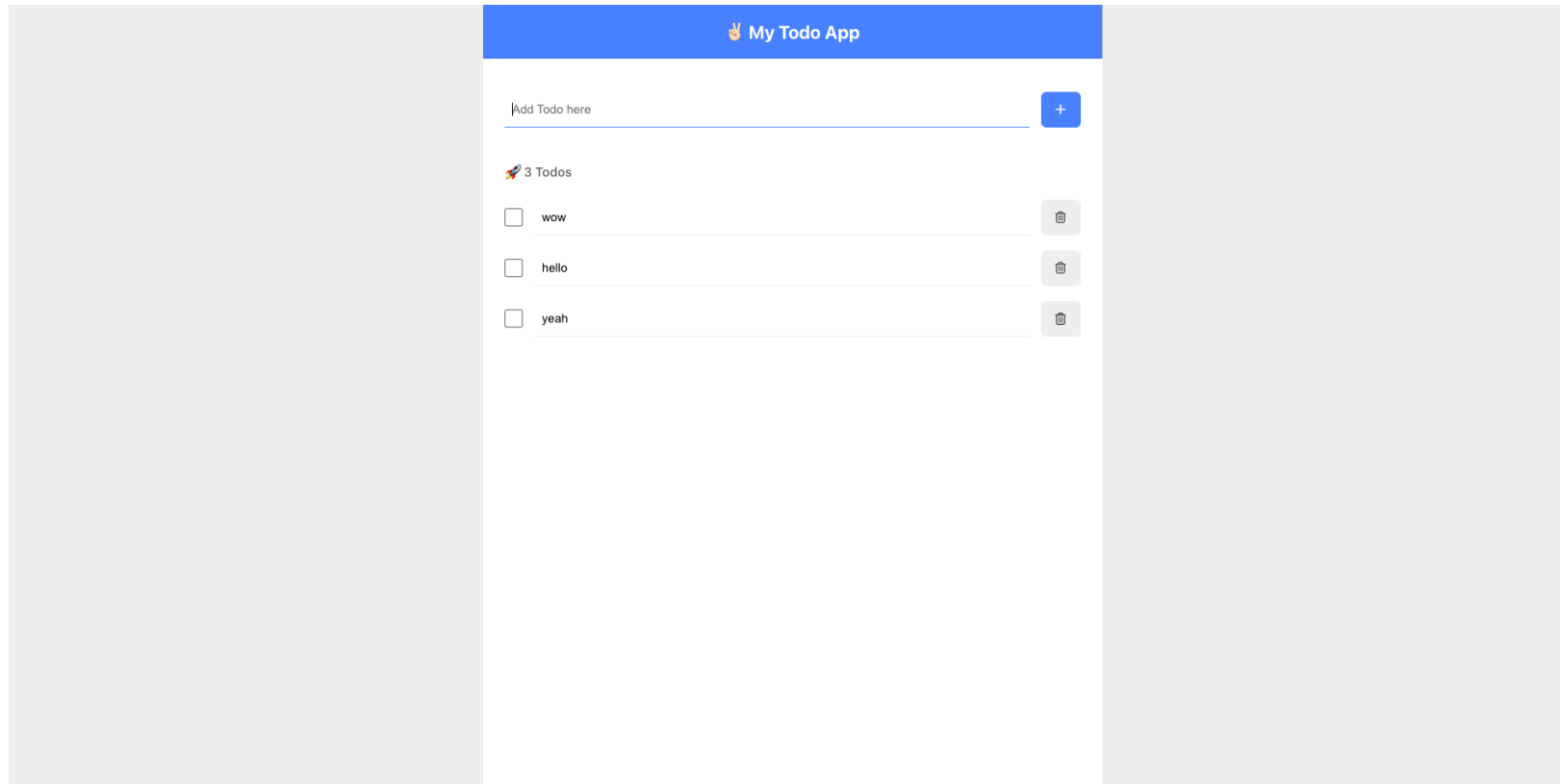
Add your new Todo		ADD
<input type="checkbox"/>	my todo1	DELETE
<input type="checkbox"/>	my todo2	DELETE
<input checked="" type="checkbox"/>	my todo3 - 수정	DELETE

4. Todo 수정 (보충 설명)

- Todo 컴포넌트에 **readOnly** 플래그 만들기
 - `readOnly = true`; 아이템 수정 불가능
 - `readOnly = false`; 아이템 수정 가능
- 사용자가 특정 **title** 클릭 시 `readOnly` 상태 **false**로 변경
- 사용자가 **Enter** 키 또는 **Return** 키 누르면 `readOnly` 상태 **true**로 변경
- 체크박스 클릭시 **done** 값 전환
 - `true` → `false`
 - `false` → `true`

4. CSS 꾸미기

🌸☁️🌸🌹 예쁘게 꾸며보자! 🌿🌸☀️🌱



서비스 통합

Part3.

독립적으로 동작하는 백엔드 애플리케이션과 프론트엔드 애플리케이션을 하나의 서비스로 통합하자!

서비스 통합?

- **Frontend**

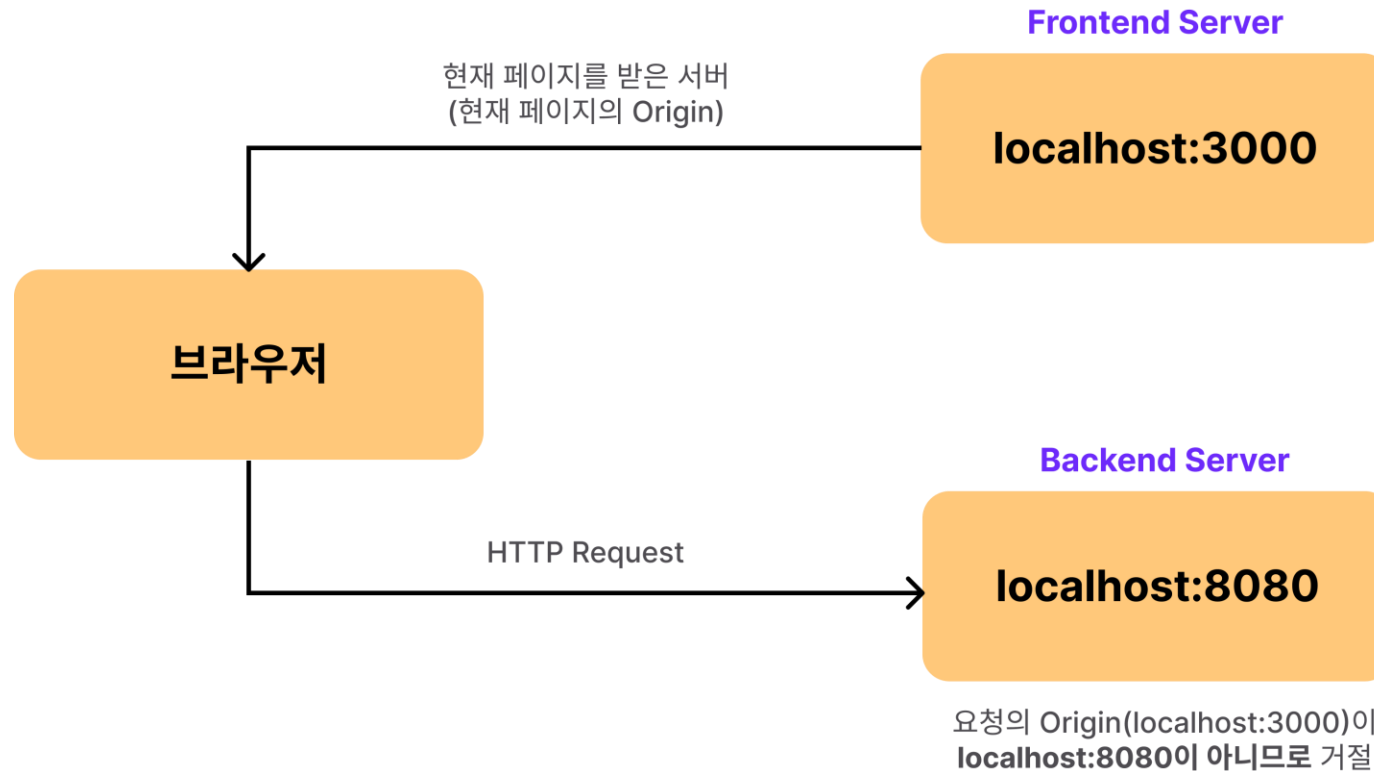
- Backend에 API 요청하는 작업
- Read, Create, Delete, Update 작업 수행

- **Backend**

- CORS 이슈 해결 (보안 관련 이슈)

CORS (Cross-Origin Resource Sharing)

- 처음 리소스를 제공한 도메인(Origin)이 현재 요청하려는 도메인과 다르더라도 요청을 허락해주는 웹 보안 방침



다음 단계는?

Part4.

그동안 배운 것을 바탕으로 여러 기능을 추가해보세요!!