



web

programming

team

project

웹 프로그래밍 팀 프로젝트

web programming team project__

<p style="text-align:left;">Left</p>

<p style="text-align:center;">Center</p>

<p style="text-align:right;">Right</p>

-

-

-

프로그래머스 데브코스 개발 4팀

팀장 오병일



web

programming

team

project

프로젝트 개요 ; Project outline ____

설계 파트 ; Architect Design ____

팀원 분담 ; Division of team members ____

프로젝트기능; Project Function__

시연 영상 ; Develop result ____

KPT 회고; Keep Problem Try ____



web

programming

team

project

Project outline ____

프로젝트 명

카페 메뉴 관리 시스템

개발 일정

24.11.29 ~ 24.12.09

특징

MariaDB와 Mybatis 이용

Spring Web Project

협업 경험을 위한 프로젝트

Git 사용으로 관리



web

programming

team

project

Development Schedule & Architect Design

└ DFD 설계

└ 기능 통합

└ ERD 설계

└ API 명세서 작성

└ dir & branch 구조

└ 개발 구현

└ 개발 통합

2024.11.29

2024.12.02

2024.12.06

2024.12.09



web

programming

team

project

Division of team members ____

Team Leader

오병일

- └ 주문 상태 갱신 기능
- └ 프로젝트 기능 설계
- └ 프로젝트 진행
- └ 프로젝트 merge

Developer

강동윤

- └ 주문 추가 기능
- └ 프로젝트 기능 설계
- └ Validation 처리
- └ Summary UI

고진영

- └ 상품 등록 기능
- └ 프로젝트 기능 설계
- └ pagination 구현
- └ Main UI



web

programming

team

project

Division of team members ____

Team Leader

오병일

- └ 주문 상태 갱신 기능
- └ 프로젝트 기능 설계
- └ 프로젝트 진행
- └ 프로젝트 merge

Developer

송혜원

- └ 주문 갱신 기능
- └ 프로젝트 기능 설계
- └ choiced_order UI

조희지

- └ 주문 삭제 기능
- └ 프로젝트 기능 설계
- └ orderlist_search UI
- └ 로그인



web

programming

team

project

project function __add_item

장바구니 상품 추가



Summary UI에 상품 데이터 정리



JSON 데이터 처리

AJAX 통신



project function __orders

orders DB 등록



JSON데이터 인계



orders DB에 주문 내용 등록



web

programming

team

project

```
// POST 요청으로 JSON 데이터 전송
fetch("/add_item", {
  method: "POST",
  headers: {
    "Content-Type": "application/json"
  },
  body: JSON.stringify(orderData)
})

.then(response => response.json())
.then(flag => {
  if (flag === 1) {
    alert("정상 등록되었습니다.");
    location.reload(); // 결제 후 페이지 새로고침
  } else {
    alert("등록 실패하였습니다.");
  }
})

.catch(error => {
  console.error("에러 발생:", error);
  alert("오류가 발생하였습니다.");
});

console.log(orderData);
```




web

programming

team

project

project function __list_up

목록 불러오기



DB를 통해 Item 테이블 접근



상품 요소 list-up

.....>

project function __pagination

페이지네이션



불러온 list-up 가공



offset, limit으로 pagination



web

programming

team

project

```
9
10
11
12 @RequestMapping(🌐"/main")  👤 Kang +1
13 public String main(
14     @RequestParam(defaultValue = "1") int page, // 현재 페이지 (기본값: 1)
15     @RequestParam(defaultValue = "5") int pageSize, // 페이지당 항목 수 (기본값: 5)
16     Model model) {
17
18     int offset = (page - 1) * pageSize; // SQL OFFSET 계산
19     List<ItemT0> lists = addItemDAO.cafeList(pageSize, offset); // 현재 페이지의 항목 수량, db에서 잘라야 할 offset정의
20     List<ItemT0> allLists = addItemDAO.itemList(); // 전체 데이터 가져오기
21     int totalItemsCount = addItemDAO.getTotalItemsCount(); // 전체 데이터 수 조회
22     int totalPages = (int) Math.ceil((double) totalItemsCount / pageSize); // 총 페이지 수 계산
23
24     model.addAttribute(attributeName: "page", page);
25     model.addAttribute(attributeName: "totalPages", totalPages);
26     model.addAttribute(attributeName: "lists", lists);
27     model.addAttribute(attributeName: "allLists", allLists);
28
29     return "main";
30 }
```



web

programming

team

project

project function __update_item

주문 수정



DB를 통해 orders 테이블 접근



이메일과 주문번호



project function __database

DB 수정 과정



데이터 일치여부 확인



데이터 수정 및 UI 출력



web

programming

team

project

<!-- 주문 ID로 주문 내역 조회 -->

<select id="findOrdersByOrderId" resultType="OrderItem">

SELECT

oi.order_item_id,

oi.order_id,

i.item_id,

i.name AS item_name,

oi.orderCount,

oi.orderPrice

FROM orders o

JOIN orderItem oi ON o.order_id = oi.order_id

JOIN item i ON oi.item_id = i.item_id

WHERE o.order_id = #{order_id} -- and oi.item_id=#{item_id}

</select>

<!-- 주문 정보 업데이트 -->

<update id="updateOrders" parameterType="Orders">

UPDATE orders

SET address = #{address}, zip_code = #{zip_code}

WHERE order_id = #{order_id}

</update>



web

programming

team

project

project function __delete_item

주문 삭제



DB를 통해 orders 테이블 접근



주문 번호를 통한 대상 검색



project function __database

DB 수정 과정



데이터 일치여부 확인



데이터 삭제 및 main return



web

programming

team

project

```
// TODO DB 자동 백업 구현 후 14시에 배송상태 업데이트 했을때 주식 풀어서 사용
// 현재 시스템 시간대의 ZonedDateTime을 가져옴
ZoneId systemZoneId = ZoneId.systemDefault(); // 현재 시스템 시간대
ZonedDateTime now = ZonedDateTime.now(systemZoneId); // 현재 시간
ZonedDateTime today14 = now.withHour(14).withMinute(0).withSecond(0).withNano(nanoOfSecond: 0);
if(now.isAfter(today14)){
    boolean isBefore14 = now.isBefore(today14); // 14시 이후인지 확인

    String orderStatus = ordersT0.getOrderStatus();
    if (orderStatus.equals("beforeDelivery")) {
        // orderStatus before 인데 금일 14시 이후에 삭제를 눌렀을 때
        // (14시 이전에 페이지를 넘겼으나, 페이지에 머무르다 14시 이후가 된 경우)
        if (isBefore14) {
            flag = 1;
            return flag;
        }
    }
}
```




web

programming

team

project

```
MariaDB [NBE2_3_1_Team4]> select * from orders;
```

order_id	email	address	order_date	orderStatus	zip_code	password
1	dhquddlf5@gmail.com	서울시 강남구 테헤란로 1	2024-12-05 15:06:01	afterDelivery	12345	*668425423DB5193AF921380129F465A6425216D0
2	user2@example.com	부산시 해운대구 해운대로 2	2024-12-05 15:06:01	afterDelivery	54321	*DC52755F3C09F5923046BD42AFA76BD1D80DF2E9
3	user3@example.com	대구시 중구 중앙대로 3	2024-12-05 15:06:01	afterDelivery	11111	*40C3E7D386A2FADBDF69ACEBE7AA4DC3C723D798
4	user4@example.com	인천시 연수구 송도대로 4	2024-12-05 15:06:01	afterDelivery	22222	*F97AEB38B3275C06D822FC9341A2151642C81988
8	dhquddlf5@gmail.com	부산 해운대구	2024-12-08 12:59:28	beforeDelivery	24583	*FCB7DBC53D42BA2DCC99C534E10260615E030EF4
9	dhquddlf5@gmail.com	부천시 원미구	2024-12-08 13:51:43	beforeDelivery	12222	*C456C4ED017C0A70C5F25F9F14962AAAF2A3CF52
10	dhquddlf5@gmail.com	경기도 부천시 원미구	2024-12-08 18:24:03	beforeDelivery	12222	*F6DD0C0AC75395CB5BFC12C46B8880CD156B4799
11	dhquddlf5@gmail.com	경기도 부천시	2024-12-09 13:56:47	beforeDelivery	11111	*F6DD0C0AC75395CB5BFC12C46B8880CD156B4799
12	dhquddlf5@gmail.com	경기도 부천시	2024-12-09 20:39:32	beforeDelivery	11111	*F6DD0C0AC75395CB5BFC12C46B8880CD156B4799

9 rows in set (0.001 sec)

```
MariaDB [NBE2_3_1_Team4]> CREATE EVENT delivery_update_temp2 ON SCHEDULE EVERY 1 DAY STARTS '2024-12-09 23:54:00' DO UPDATE orders SET orderStatus = 'afterDelivery' WHERE orderStatus = 'beforeDelivery' AND order_date < DATE_ADD(DATE(NOW()), INTERVAL 14 HOUR);
```

Query OK, 0 rows affected (0.006 sec)

```
MariaDB [NBE2_3_1_Team4]> drop event delivery_update_temp;
```

Query OK, 0 rows affected (0.011 sec)

```
MariaDB [NBE2_3_1_Team4]> select * from orders;
```

order_id	email	address	order_date	orderStatus	zip_code	password
1	dhquddlf5@gmail.com	서울시 강남구 테헤란로 1	2024-12-05 15:06:01	afterDelivery	12345	*668425423DB5193AF921380129F465A6425216D0
2	user2@example.com	부산시 해운대구 해운대로 2	2024-12-05 15:06:01	afterDelivery	54321	*DC52755F3C09F5923046BD42AFA76BD1D80DF2E9
3	user3@example.com	대구시 중구 중앙대로 3	2024-12-05 15:06:01	afterDelivery	11111	*40C3E7D386A2FADBDF69ACEBE7AA4DC3C723D798
4	user4@example.com	인천시 연수구 송도대로 4	2024-12-05 15:06:01	afterDelivery	22222	*F97AEB38B3275C06D822FC9341A2151642C81988
8	dhquddlf5@gmail.com	부산 해운대구	2024-12-08 12:59:28	afterDelivery	24583	*FCB7DBC53D42BA2DCC99C534E10260615E030EF4
9	dhquddlf5@gmail.com	부천시 원미구	2024-12-08 13:51:43	afterDelivery	12222	*C456C4ED017C0A70C5F25F9F14962AAAF2A3CF52
10	dhquddlf5@gmail.com	경기도 부천시 원미구	2024-12-08 18:24:03	afterDelivery	12222	*F6DD0C0AC75395CB5BFC12C46B8880CD156B4799
11	dhquddlf5@gmail.com	경기도 부천시	2024-12-09 13:56:47	afterDelivery	11111	*F6DD0C0AC75395CB5BFC12C46B8880CD156B4799

8 rows in set (0.030 sec)



web

programming

team

project

상품 목록

주문조회

상품 목록

	커피콩 샌드위치	5500원	<button>추가</button> <button>삭제</button>
	커피콩 치즈케이크	6000원	<button>추가</button> <button>삭제</button>
	커피콩 카푸치노	4500원	<button>추가</button> <button>삭제</button>
	커피콩 아메리카노	4000원	<button>추가</button> <button>삭제</button>

1

Summary

이메일

비밀번호

주소

우편번호

당일 오후 2시 이후의 주문은 다음날 배송
을 시작합니다.

총금액 0원

결제하기



web

programming

team

project

KPT 회고



Keep

- MyBatis 사용
- validation
- json 데이터를 활용한 AJAX 통신구현
- 효율적인 pagination 구현
- 원활한 의사소통으로 계획

설립 및 문제 해결



Problem

- 시간 부족으로 인한 미완성 부분.
- 복잡한 코드 개선 필요
- 기본 웹 구조에 대한 이해



Try

- 코드리뷰 및 리팩토링
- JSTL과 EL 도입
- Email API
- 세션 쿠키 로그인 방식
- SQL Injection 처리

9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
24
25
26



web

programming

team

project

Q&A _____

자유롭게 질문해주세요!

- ▮ 자세한 스케줄
- ▮ 중간 체크 일정
- ▮ 의사결정프로세스 등등등...