



# SHOP(안경 쇼핑몰)

안경조 : 강진혁, 김도아, 김정빈

우리인재개발원

01. 프로젝트 개요

02. 프로젝트 팀 구성 및 역할

03. 프로젝트 수행 절차 및 방법

04. 프로젝트 수행 결과

05. 자체 평가 의견

06. 개인 추가

A photograph of a person wearing a white button-down shirt, pointing their right index finger towards a screen. A white rectangular box with a blue top border is overlaid on the image, containing the Korean word '목차' (Table of Contents) in large black font.

목차

# 01 프로젝트 개요

---

## 프로젝트 주제 및 선정 배경

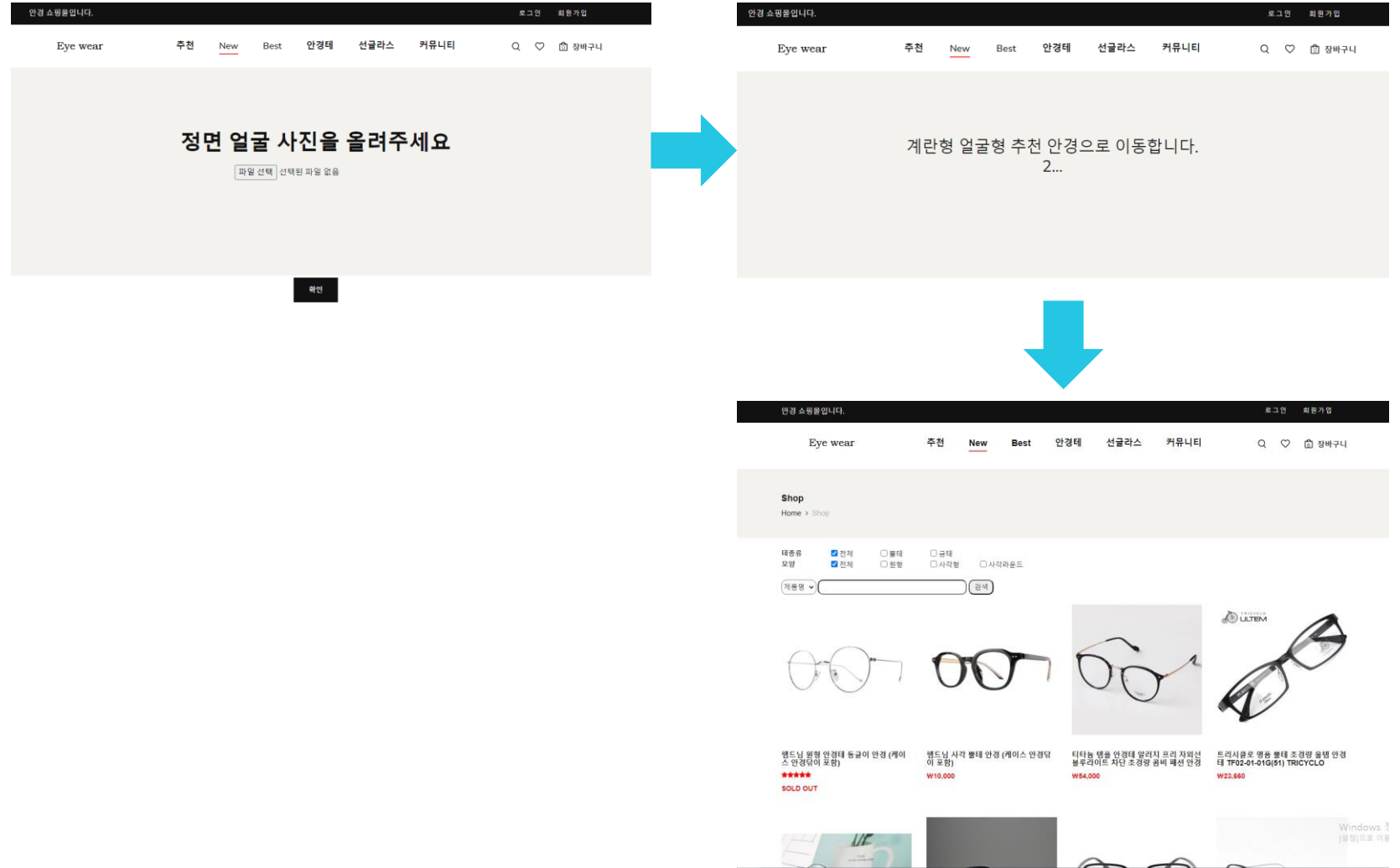
---

AI기능에 중점을 두어 기본적인 웹 페이지 구현이 소홀해 질 수 있기 때문에  
기본적인 기능에 충실한 쇼핑몰 프로젝트를 선정, AI기능은 어디까지나 추가기능으로,  
얼굴형에 어울리는 안경을 추천하는 안경 쇼핑몰로 주제를 선정

# 01 프로젝트 개요

## 구현 내용 및 기대효과

- AI기술을 이용해 얼굴형을 인식하여 얼굴형에 따라 어울리는 안경을 추천해주는 기능
- 이를 이용해 사용자의 흥미를 유발하여 이용자 수 증가를 기대할 수 있다.



# 01 프로젝트 개요

---

## 개발 환경

---



## 사용 기술

---



## 02 프로젝트 팀 구성 및 역할

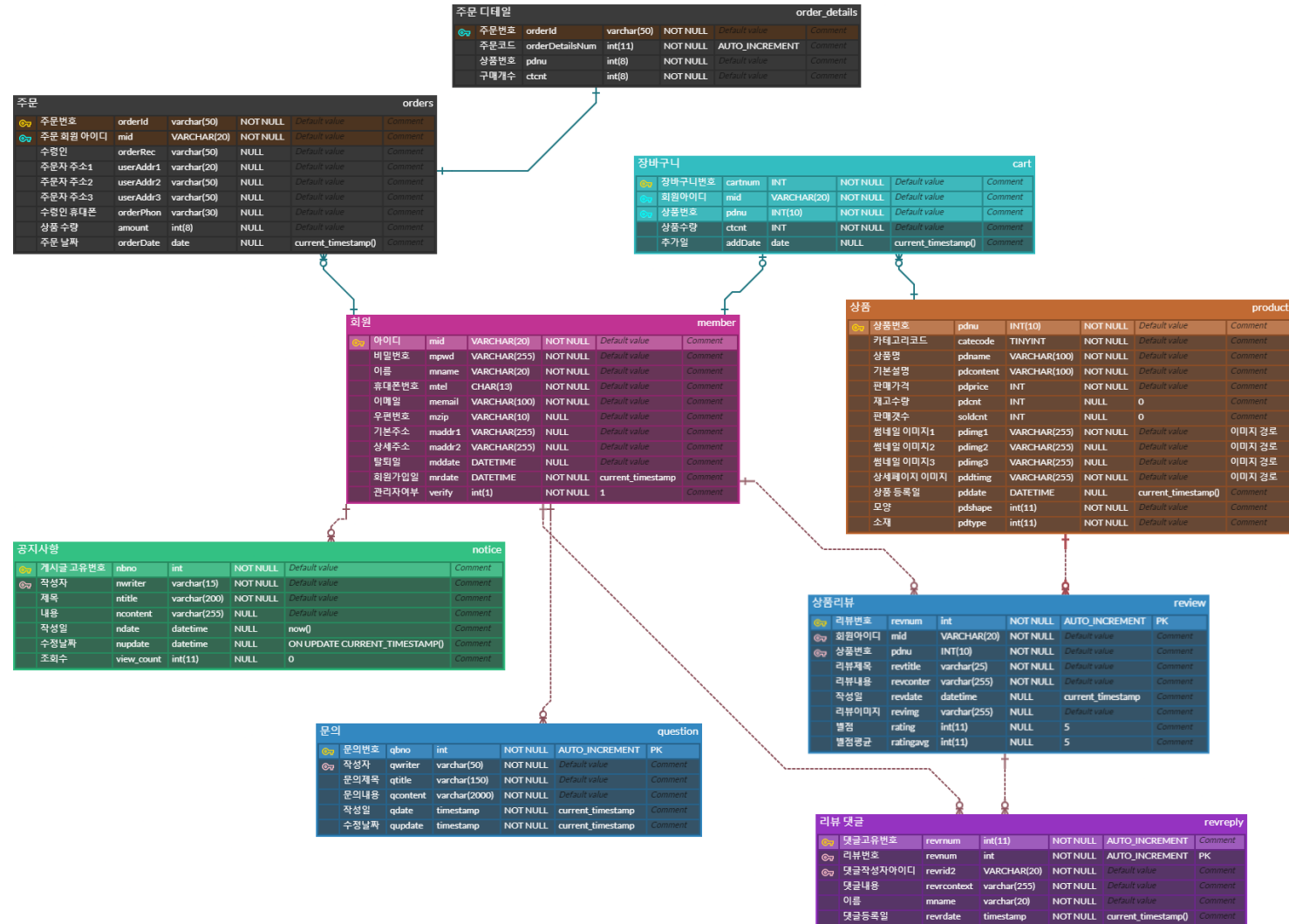
훈련생	역할	담당 업무
강진혁	팀장	▶ 공지사항 기능 구현 ▶ Q&A 기능 구현
김도아	팀원	▶ 로그인 기능 구현 ▶ 관리자 페이지 구현 ▶ 구매관련 기능 구현
김정빈	팀원	▶ 상품 관련 기능 구현 ▶ 리뷰 기능 구현 ▶ AI 기능 구현 ▶ AWS 구축

## 03 프로젝트 수행 절차 및 방법

구분	기간	활동	비고
사전 기획	▶ 5/31(수)~6/8(목)	▶ 프로젝트 기획 및 주제 선정 ▶ 기획안 작성	▶ 아이디어 선정
구조 설계	▶ 6/9(금)~6/12(월)	▶ 필요 기능 정의 ▶ 데이터베이스 ERD 설계	▶ 기본 구조 설계
기본 기능 구현	▶ 6/13(화)~7/10(월)	▶ 쇼핑몰 기본 기능 구현	▶ 기본 기능 구현
AI 기능 구현	▶ 6/27(화)~6/30(금)	▶ 데이터셋 작성 및 기능 구현	▶ AI 기능 구현
AWS 구축	▶ 7/3(월)~7/7(금)	▶ AWS 구축 및 연결	▶ EC2, RDS, S3
총 개발기간	▶ 5/31(수)~7/10(월)(총 6주)	-	-

# 04 프로젝트 수행 결과

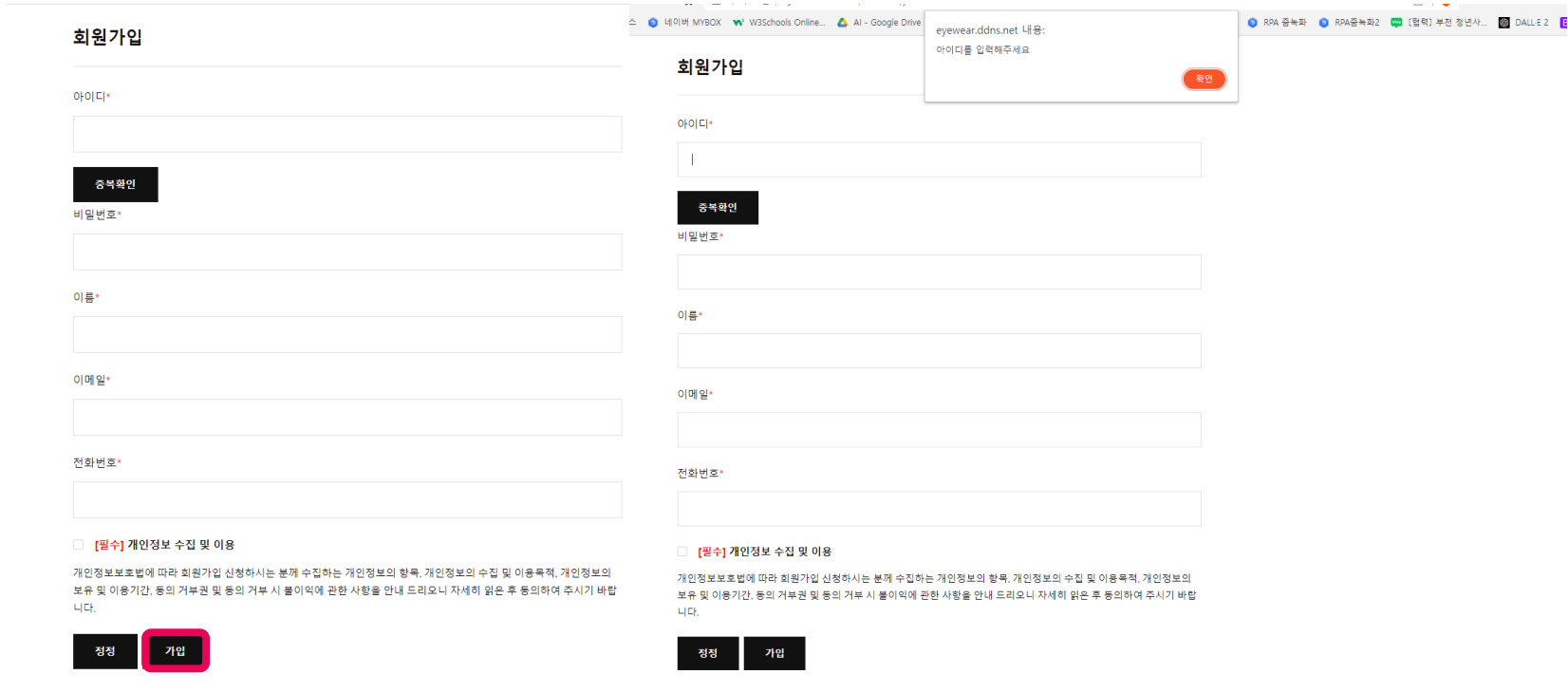
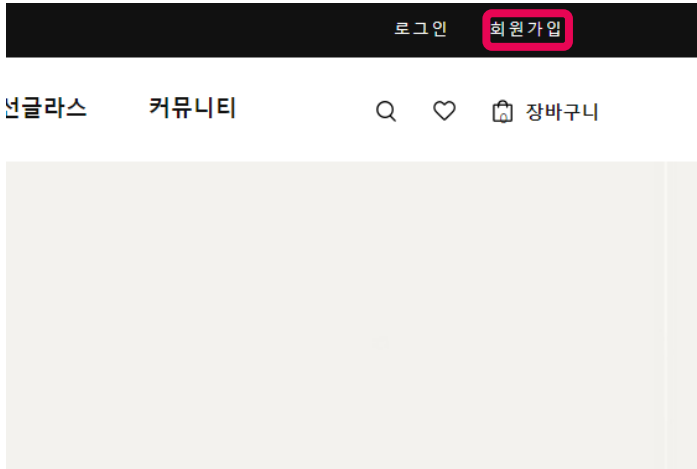
## 데이터베이스 ERD





# 04 프로젝트 수행 결과

## 회원 가입



# 04 프로젝트 수행 결과

## 회원 가입

### 회원가입

아이디\*

test1

중복된 아이디입니다.

중복확인

### 회원가입

아이디\*

test123

사용 가능한 아이디입니다.

중복확인

### 회원가입

아이디\*

test123

사용 가능한 아이디입니다.

중복확인

비밀번호\*

....

이름\*

테스트

이메일\*

test123@naver.com

전화번호\*

0105555555

☒ [필수] 개인정보 수집 및 이용

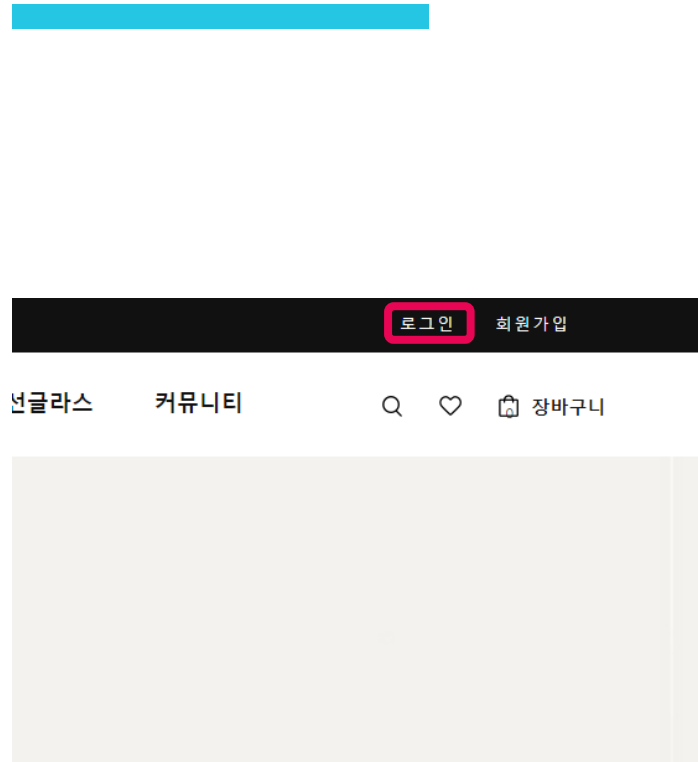
개인정보보호법에 따라 회원가입 신청하시는 분께 수집하는 개인정보의 항목, 개인정보의 수집 및 이용목적, 개인정보의 보유 및 이용기간, 동의 거부권 및 동의 거부 시 불이익에 관한 사항을 안내 드리오니 자세히 읽은 후 동의하여 주시기 바랍니다.

정정

가입

# 04 프로젝트 수행 결과

## 로그인



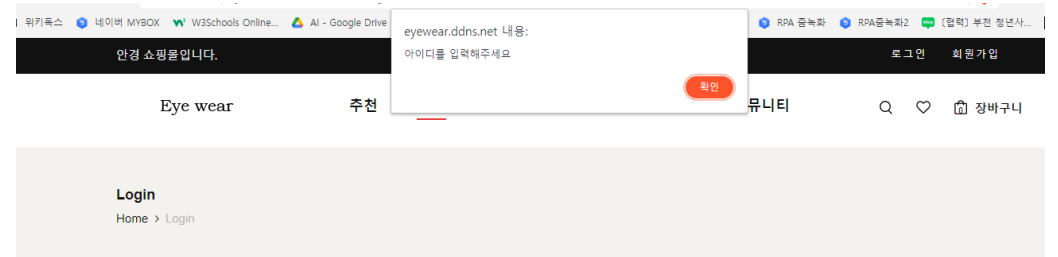
### 로그인

아이디\*

비밀번호\*

로그인

회원가입



### 로그인

아이디\*

비밀번호\*

로그인

회원가입

# 04 프로젝트 수행 결과

## 로그인

```
// 로그인 데이터 사용자 -> 서버로
@RequestMapping(value = "member/login", method = RequestMethod.POST)
public String loginPOST(MemberVO vo, HttpServletRequest req) throws Exception {
    logger.info("로그인 데이터 전송 완료");
    HttpSession session = req.getSession();
    MemberVO login = memberService.login(vo);

    if(login == null) {
        session.setAttribute("member", null);
    } else {
        session.setAttribute("member", login);
    }
}
```

입력한 정보가 DB와 일치할 경우 회원 정보를 session값에 저장,  
Verify값이 1인지 2인지에 따라서 일반 회원과 관리자 구분

### 로그인

아이디\*

test123

비밀번호\*

....

로그인

회원가입

안경 쇼핑물입니다.

로그아웃 회원가입 마이페이지

Eye wear

추천

New

Best

안경테

선글라스

커뮤니티

장바구니

test123님 환영합니다.

회원정보 수정

로그아웃

# 04 프로젝트 수행 결과

## 회원 정보 수정

안경 소핑을입니다.

Eye wear 추천 **New** Best 안경테 선글라스

test123님 환영합니다.

**회원정보 수정**

로그아웃

### 회원 정보 수정

아이디\*

test123

비밀번호\*

이름\*

테스트

이메일\*

test123@naver.com

전화번호\*

0105555555

정정 수정

eyewear.ddns.net 내용:  
비밀번호를 입력해주세요

확인

### 회원 정보 수정

아이디\*

test123

비밀번호\*

이름\*

테스트

이메일\*

test123@naver.com

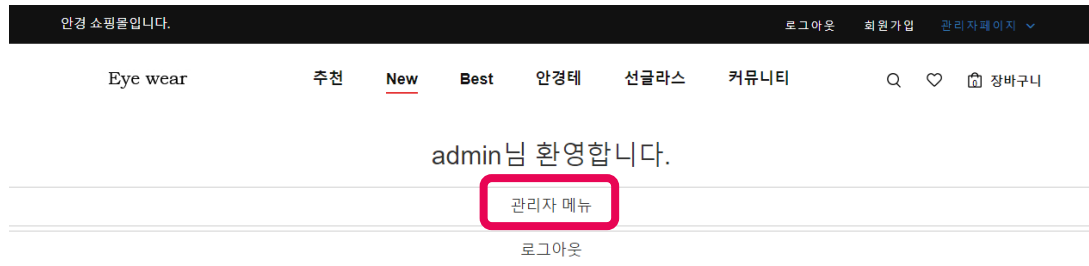
전화번호\*

0105555555

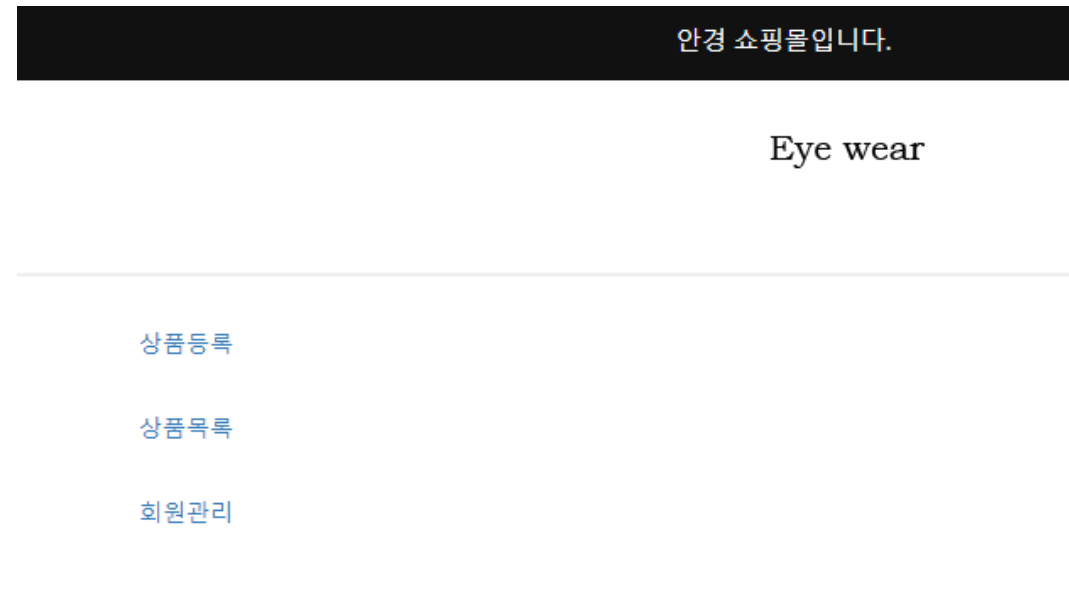
정정 수정

# 04 프로젝트 수행 결과

## 관리자 페이지

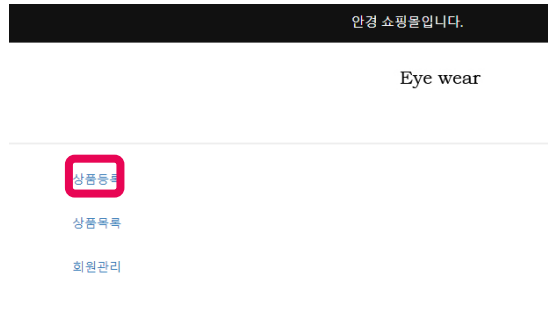


Session에 저장된  
Verify값이 1인가 2인가에 따라서 일반 회원과 관리자 구분



# 04 프로젝트 수행 결과

## 관리자 페이지(상품 등록)



### 제품상세

본문 이미지  
[파일선택] 선택한 파일 업로드

Windows 정보  
[설정]으로 이동하여 Win

등록

이미지는 MutipartFile로 받아서 s3에 저장

```
@PostMapping("product/productInsert")
public String productInsert(@ModelAttribute ProductVO vo,
    @RequestParam("image1") MultipartFile image1,
    @RequestParam("image2") MultipartFile image2,
    @RequestParam("image3") MultipartFile image3,
    @RequestParam("image4") MultipartFile image4,
    HttpServletRequest request) {

    AWSCredentials credentials = new BasicAWSCredentials(akey, pkey);
    // Create an S3 client with default credential provider chain
    AmazonS3 s3Client = AmazonS3ClientBuilder.standard()
        .withCredentials(new AWSSessionCredentialsProvider(credentials))
        .withRegion(Regions.AP_NORTHEAST_2)
        .build();

    if (!image1.isEmpty()) {
        String fileRealName = image1.getOriginalFilename();
        String imageName = generateUniqueImageName(fileRealName);
        long size = image1.getSize();

        System.out.println("File name: " + imageName);
        System.out.println("Capacity size (byte): " + size);


        // Specify the bucket name and key (file path) for the image in the S3 bucket
        String bucketName = "버킷이름";
        String key = "images/" + imageName;

        try {
            InputStream inputStream = image1.getInputStream();
            // Upload the image file to the S3 bucket
            PutObjectRequest putObjectRequest = new PutObjectRequest(bucketName, key, inputStream, new ObjectMetadata());
            s3Client.putObject(putObjectRequest);

            // Set the image URL to the S3 bucket URL for accessing the uploaded image
            vo.setPding1(key);
        } catch (Exception e) {
            e.printStackTrace();
            // Handle exception
        }
    }
}
```

Windows 정보  
[설정]으로 이동하여 Win

## 관리자 페이지(상품 목록)



재고수량 10

판매가수 0

상품코드 11583583

카테고리 ○ 안경대 ● 신강라스

종류 ○ 플렉스 ● 금대

모양 ● 원형 ○ 사각형 ○ 사각라운드 ○ 캣스형 ○ 스포츠형

## 제품상세


본문 이미지


과일 선택

선택한 과일 없음

### Size display

Fashion trend 1986 trend glasses new fashion glasses are your fashion





WEIGHT: 39.5g

Windows


[물류]온도 1




# 04 프로젝트 수행 결과

## 관리자 페이지(상품 목록)

관리자 페이지(상품 목록)

상품번호	샘플사진	상품명	상품종류	상품가격	재고	등록일	수정	삭제
519611		테스트	안경테	₩1,234	10	2023-07-11 03:28:32	수정	삭제

관리자 페이지(상품 목록)

상품번호	샘플사진	상품명	상품종류	상품가격	재고	등록일	수정	삭제
17944148		스타일호른 가빈 스포츠 선글라스 G90 얼굴을 딱 잡아주는 안정적인 선글라스 (도수클리프 포함)	선글라스	₩24,900	10	2023-07-04 04:48:35	수정	삭제
12816348		라피덴트 라이딩 고글 주야간 겸용 편광 자외선 차단 스포츠 선글라스	선글라스	₩29,900	10	2023-07-04	수정	삭제

# 04 프로젝트 수행 결과

## 관리자 페이지(회원 관리)

안경 쇼핑물입니다.

Eye wear

상품등록

상품목록

회원관리

안경 쇼핑물입니다.로그아웃회원가입관리자페이지

Eye wear추천NewBest안경테선글라스커뮤니티

회원목록

회원관리 페이지입니다.

아이디	비밀번호	이름	전화번호	이메일	우편번호	주소	상세주소	회원등록일	관리자여부	정보수정	강제탈퇴
admin	1	1	124	1@342					2	수정	강제탈퇴
test	1234	테스트	0105555553	tet@naver.com					1	수정	강제탈퇴
test1	1234	테스트1	0105555555	test1@naver.com					1	수정	강제탈퇴
test10	1234	테스트10	010-555-5555	test10@naver.com					1	수정	강제탈퇴
test11	1234	테스트11	010-555-5555	test11@naver.com					1	수정	강제탈퇴
test12	1234	테스트12	0107777777	test12@naver.com					1	수정	강제탈퇴
test123	1234	테스트	0105555555	test123@naver.com					1	수정	강제탈퇴
test2	1234	테스트2	0105555555	test2@naver.com					1	수정	강제탈퇴
test3	1234	테스트3	0105555555	test3@naver.com					1	수정	강제탈퇴

# 04 프로젝트 수행 결과

## 관리자 페이지(회원 관리)

안경 쇼핑물입니다.

로그아웃회원가입관리자페이지

Eye wear

추천

New

Best

안경테 커뮤니티

선글라스

Q

♡

장바구니

회원목록

회원관리 페이지입니다.

아이디	비밀번호	이름	전화번호	이메일	우편번호	주소	상세주소	회원등록일	관리자여부	정보수정	강제탈퇴
admin	1	1	124	1@342					2	수정	강제탈퇴
test	1234	테스트	0105555553	tet@naver.com					1	수정	강제탈퇴
test1	1234	테스트1	0105555555	test1@naver.com					1	수정	강제탈퇴
test10	1234	테스트10	010-555-5555	test10@naver.com					1	수정	강제탈퇴
test11	1234	테스트11	010-555-5555	test11@naver.com					1	수정	강제탈퇴
test12	1234	테스트12	0107777777	test12@naver.com					1	수정	강제탈퇴
test123	1234	테스트	0105555555	test123@naver.com					1	수정	강제탈퇴
test2	1234	테스트2	0105555555	test2@naver.com					1	수정	강제탈퇴
test3	1234	테스트3	0105555555	test3@naver.com					1	수정	강제탈퇴

### 회원 정보 수정

아이디\*

test11

비밀번호\*

이름\*

테스트11

이메일\*

test11@naver.com

전화번호\*

010-555-5555

정정

수정

# 04 프로젝트 수행 결과

## 관리자 페이지(회원 관리)

안경 쇼핑을합니다.

로그아웃 회원가입 관리자페이지

Eye wear

추천

New

Best

안경테

선글라스

커뮤니티

회원목록

회원관리 페이지입니다.

아이디	비밀번호	이름	전화번호	이메일	우편번호	주소	상세주소	회원등록일	관리자여부	정보수정	강제탈퇴
admin	1	1	124	1@342					2	수정	강제탈퇴
test	1234	테스트	0105555553	tet@naver.com					1	수정	강제탈퇴
test1	1234	테스트1	0105555555	test1@naver.com					1	수정	강제탈퇴
test10	1234	테스트10	010-555-5555	test10@naver.com					1	수정	강제탈퇴
test11	1234	테스트11	010-555-5555	test11@naver.com					1	수정	강제탈퇴
test12	1234	테스트12	0107777777	test12@naver.com					1	수정	강제탈퇴
test123	1234	테스트	0105555555	test123@naver.com					1	수정	강제탈퇴
test2	1234	테스트2	0105555555	test2@naver.com					1	수정	강제탈퇴
test3	1234	테스트3	0105555555	test3@naver.com					1	수정	강제탈퇴

안경 쇼핑을합니다.

로그아웃 회원가입 관리자페이지

Eye wear

추천

New

Best

안경테

선글라스

커뮤니티

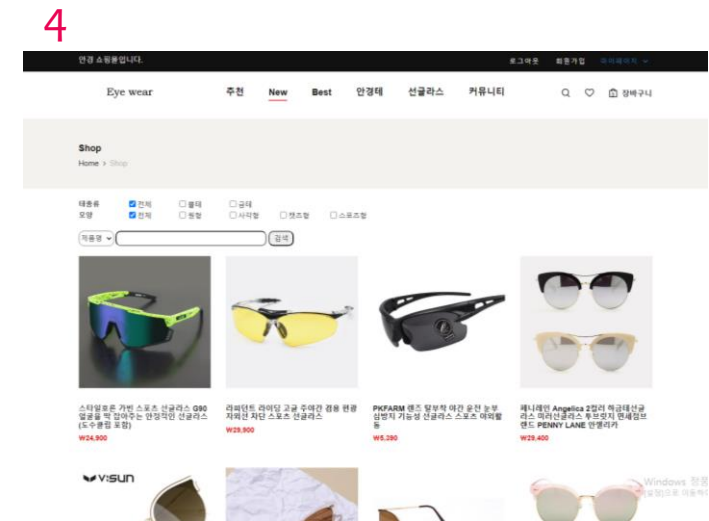
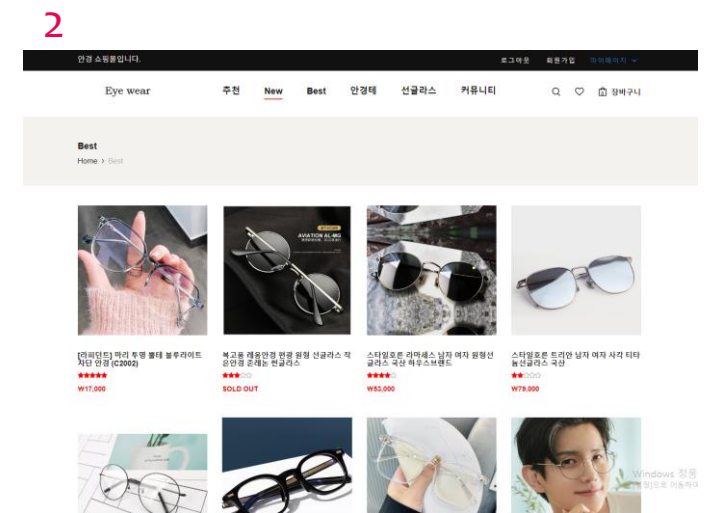
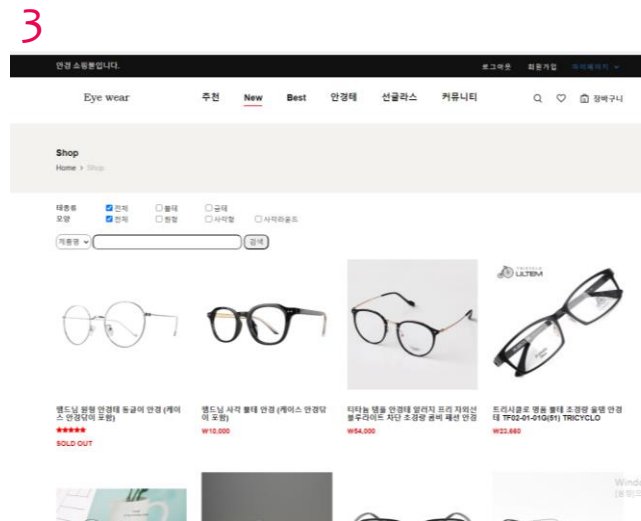
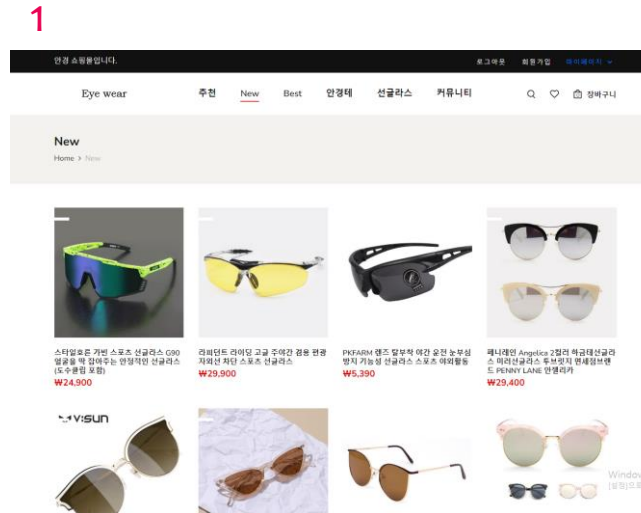
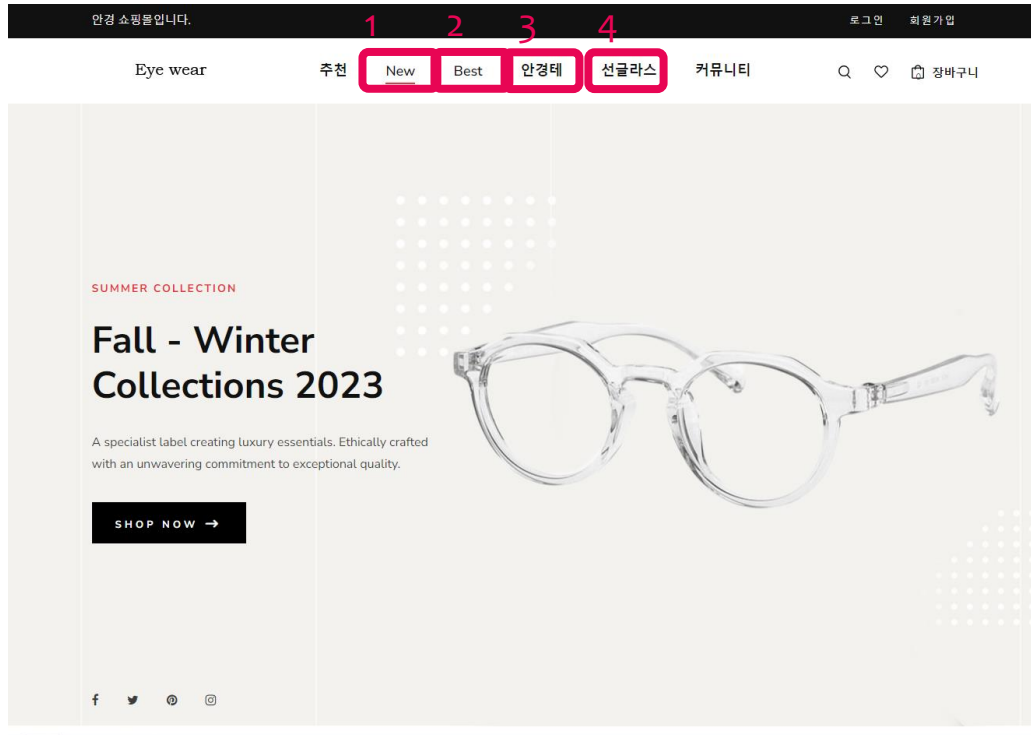
회원목록

회원관리 페이지입니다.

아이디	비밀번호	이름	전화번호	이메일	우편번호	주소	상세주소	회원등록일	관리자여부	정보수정	강제탈퇴
admin	1	1	124	1@342					2	수정	강제탈퇴
test	1234	테스트	0105555553	tet@naver.com					1	수정	강제탈퇴
test1	1234	테스트1	0105555555	test1@naver.com					1	수정	강제탈퇴
test10	1234	테스트10	010-555-5555	test10@naver.com					1	수정	강제탈퇴
test11	1234	테스트11	010-555-5555	test11@naver.com					1	수정	강제탈퇴
test12	1234	테스트12	0107777777	test12@naver.com					1	수정	강제탈퇴
test123	1234	테스트	0105555555	test123@naver.com					1	수정	강제탈퇴
test2	1234	테스트2	0105555555	test2@naver.com					1	수정	강제탈퇴

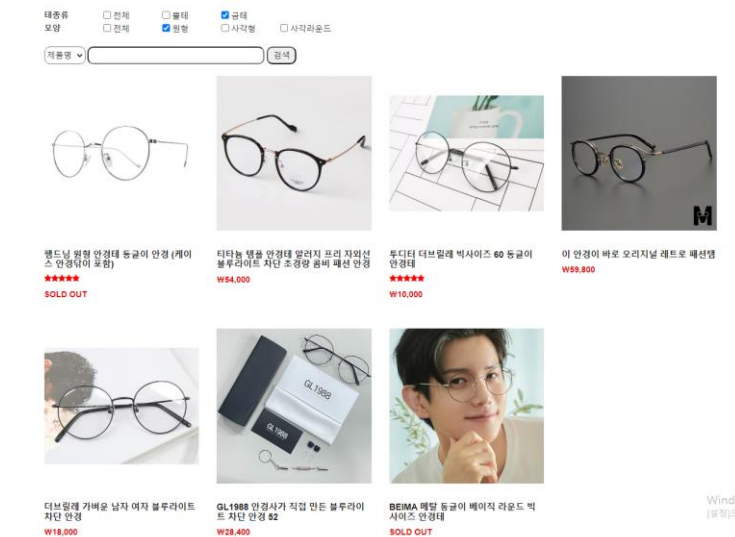
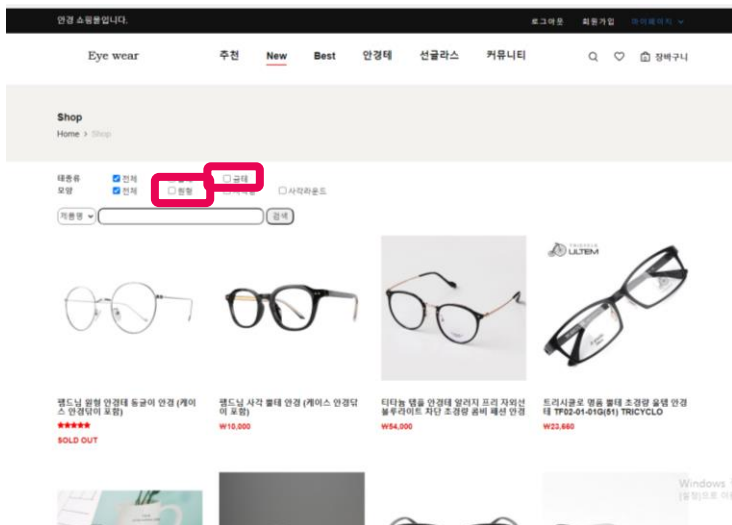
# 04 프로젝트 수행 결과

## 상품 페이지



# 04 프로젝트 수행 결과

## 상품 페이지

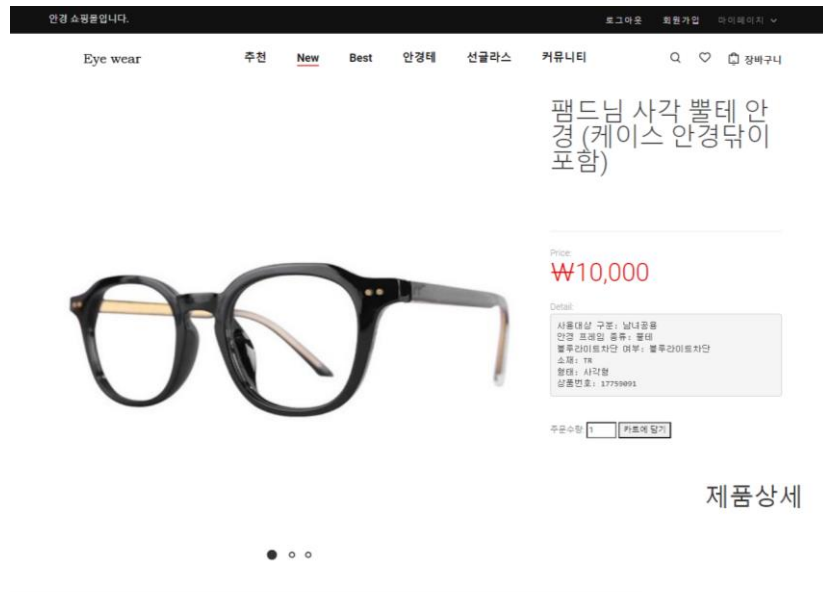
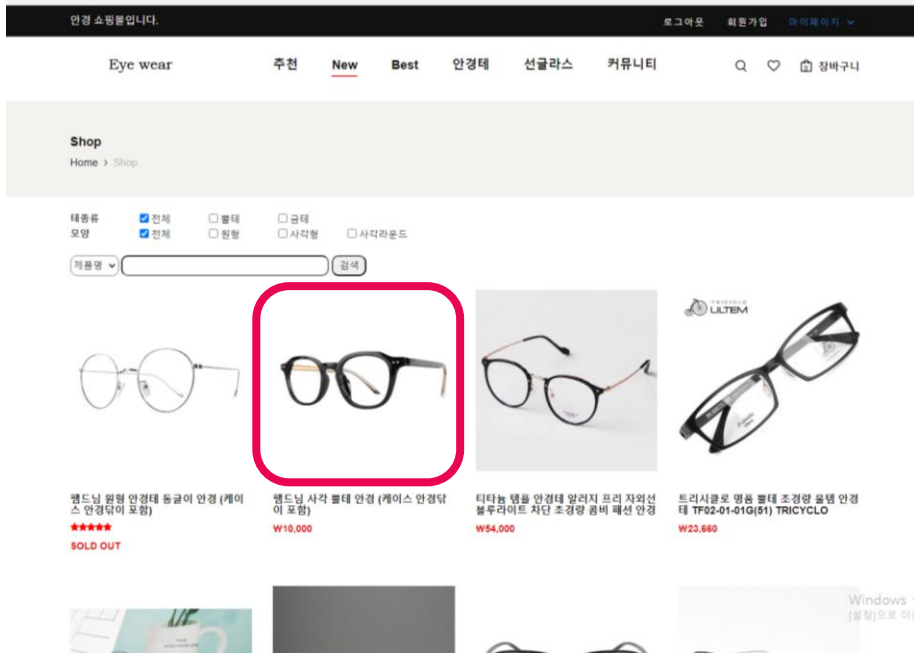


필터링 기능은 List로 받은 값을 반복문을 통해서 처리

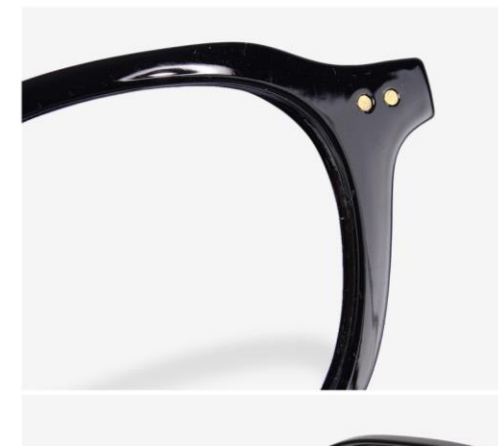
```
<sql id="typelist">
  pdtype in
  <choose>
    <when test="pdtypes.contains(0)">
      <!-- Lookup Everywhere -->
      (1, 2)
    </when>
    <otherwise>
      <!-- Search only for the relevant part -->
      <foreach collection="pdtypes" item="pdtype" open="(" close=")" separator=",">
        #{pdtype}
      </foreach>
    </otherwise>
  </choose>
</sql>
<sql id="shapelist">
  pdshape in
  <choose>
    <when test="pdshapes.contains(0)">
      <!-- Lookup Everywhere -->
      (1, 2, 3, 4, 5)
    </when>
    <otherwise>
      <!-- Search only for the relevant part -->
      <foreach collection="pdshapes" item="pdshape" open="(" close=")" separator=",">
        #{pdshape}
      </foreach>
    </otherwise>
  </choose>
</sql>
```

# 04 프로젝트 수행 결과

## 상품 상세 페이지

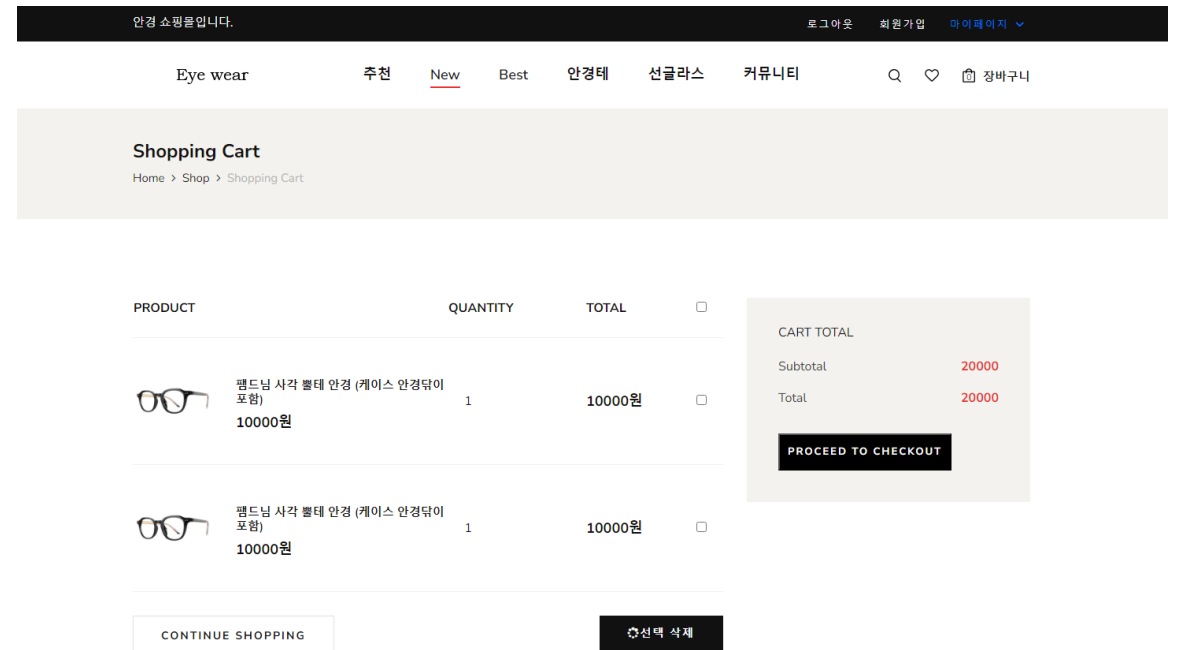
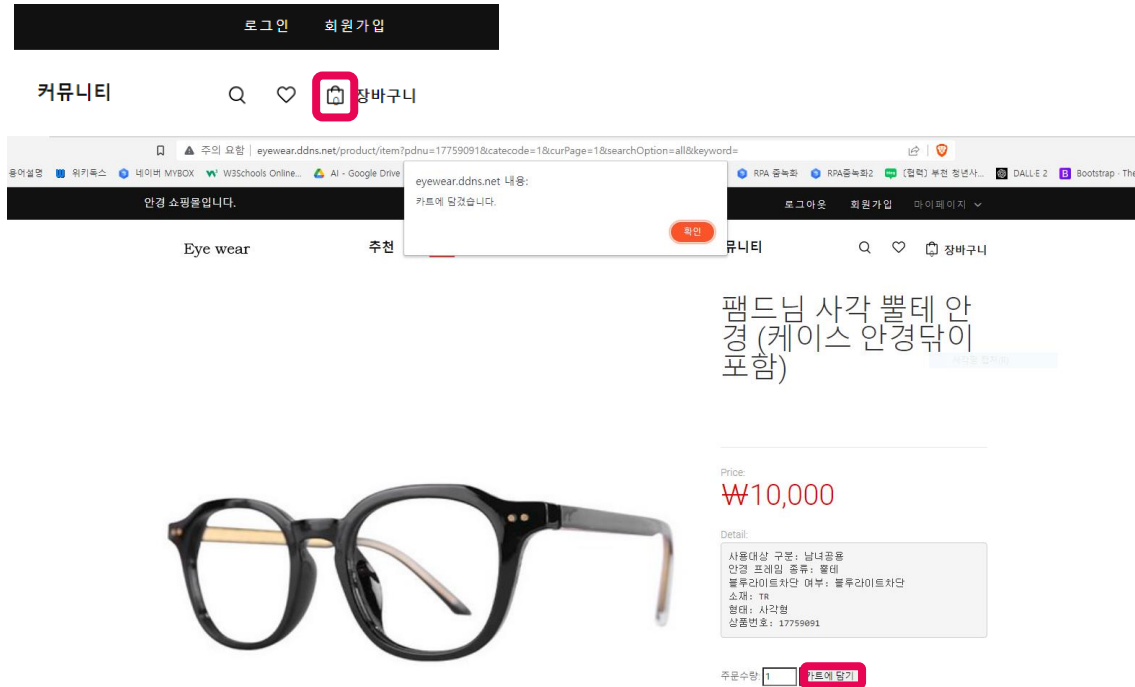


제품상세



# 04 프로젝트 수행 결과

## 장바구니

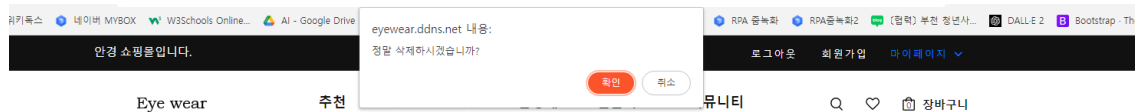




Windows 정품 인증  
[설정]으로 이동하여 Windows



# 04 프로젝트 수행 결과

## 장바구니




PRODUCT	QUANTITY	TOTAL	
 캠드님 사각 볼테 안경 (케이스 안경담이 포함) 10000원	1	10000원	<input type="checkbox"/>
 캠드님 사각 볼테 안경 (케이스 안경담이 포함) 10000원	1	10000원	<input checked="" type="checkbox"/>
<a href="#">CONTINUE SHOPPING</a>		<a href="#">선택 삭제</a>	

CART TOTAL

Subtotal 20000

Total 20000

[PROCEED TO CHECKOUT](#)

PRODUCT	QUANTITY	TOTAL	
 캠드님 사각 볼테 안경 (케이스 안경담이 포함) 10000원	1	10000원	<input type="checkbox"/>
<a href="#">CONTINUE SHOPPING</a>		<a href="#">선택 삭제</a>	

CART TOTAL

Subtotal 10000

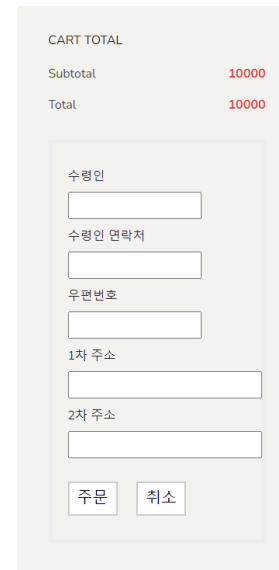
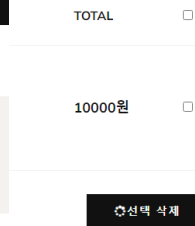
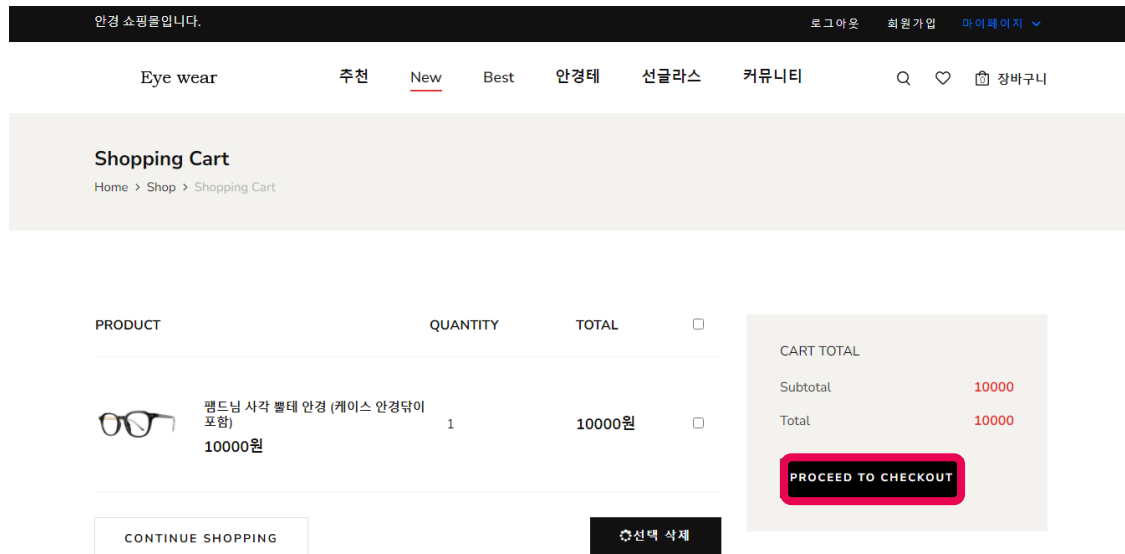
Total 10000

[PROCEED TO CHECKOUT](#)

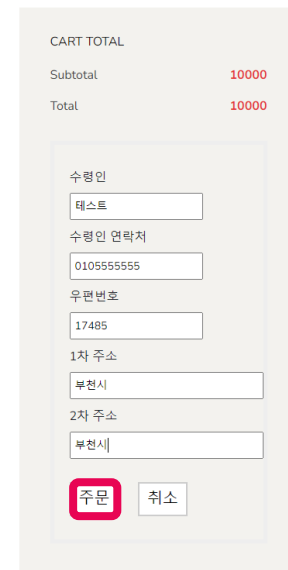
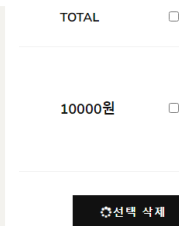
Windows 정품 인증

# 04 프로젝트 수행 결과

## 장바구니(구매)



Windows 정



Windows 정  
[설정]으로 이동

# 04 프로젝트 수행 결과

## 구매 정보

CART TOTAL

Subtotal 10000

Total 10000

수령인

테스트

수령인 연락처

0105555555

우편번호

17485

1차 주소

부천시

2차 주소

부천시

주문 취소

로그아웃 회원가입 마이페이지

니티

장바구니

안경 소팔물입니다.		로그아웃	회원가입	마이페이지
Eye wear		주천	New	Best
안경테		선글라스	커뮤니티	
OrderList		Home > Shop > OrderList		
주문번호	20230627_096239			
수령인	test			
주소	(test) test test			
가격	113231원			
상태				
주문번호	20230627_803710			
수령인	test			
주소	(test) test test			
가격	414362원			
상태				
주문번호	20230627_975747			
수령인	test			
주소	(test) test test			

안경 소팔물입니다.

로그아웃 회원가입 마이페이지

Eye wear

주천

New

Best

안경테

선글라스

커뮤니티

장바구니

OrderView

Home > Shop > OrderView

수령인  
테스트

주소  
(17485) 부천시 부천시

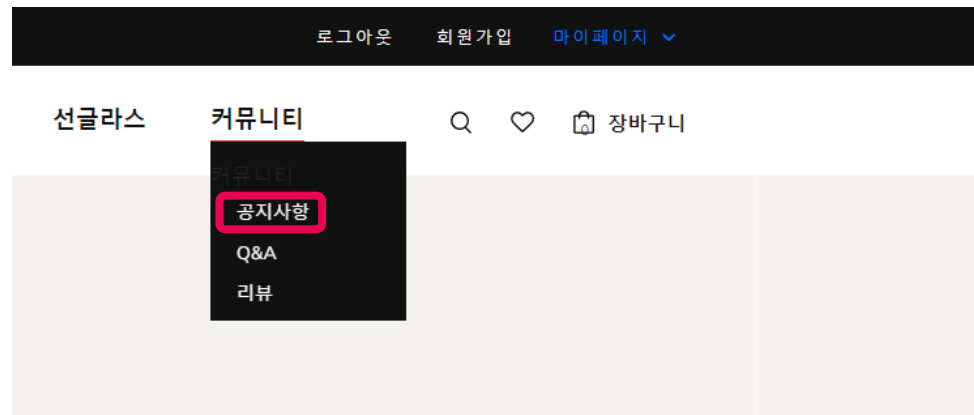
가격  
10000원



캠드님 사각 볼테 안경 (케이스 안경닦이 포함)  
개당 가격: 10000원  
구입 수량: 1 개  
최종 가격: 10000원

# 04 프로젝트 수행 결과

## 공지사항



1

## 게시판 등록

제목

내용

작성자

2

게시물

게시판 번호

게시판 제목

게시판 내용

게시판 작성자

게시판 등록일

게시판 수정일

3

## 조회 페이지

게시판 번호

게시판 제목

게시판 내용

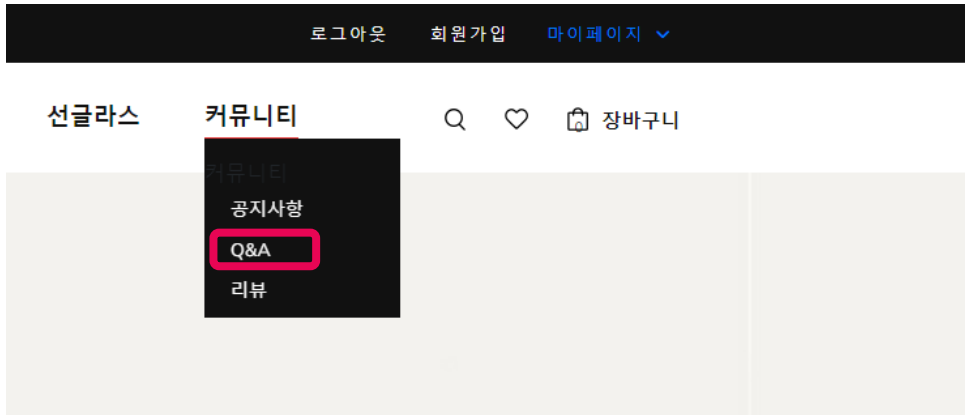
게시판 작성자

게시판 등록일

게시판 수정일

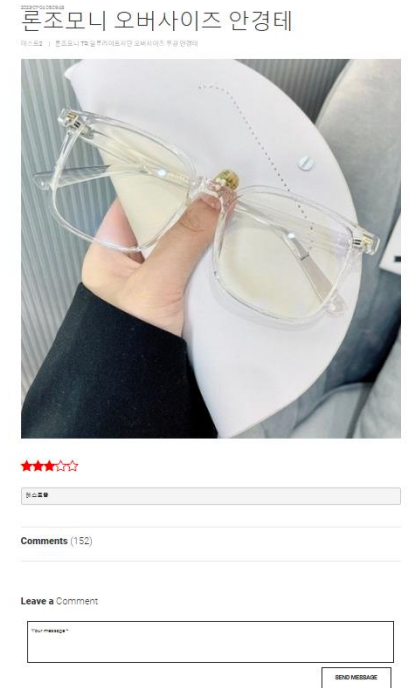
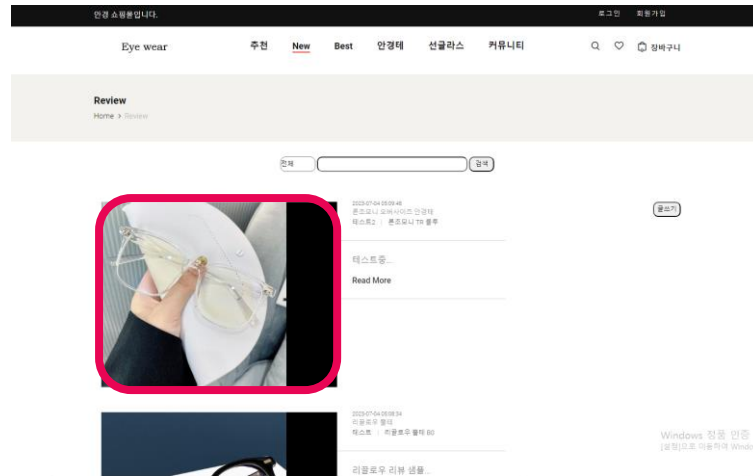
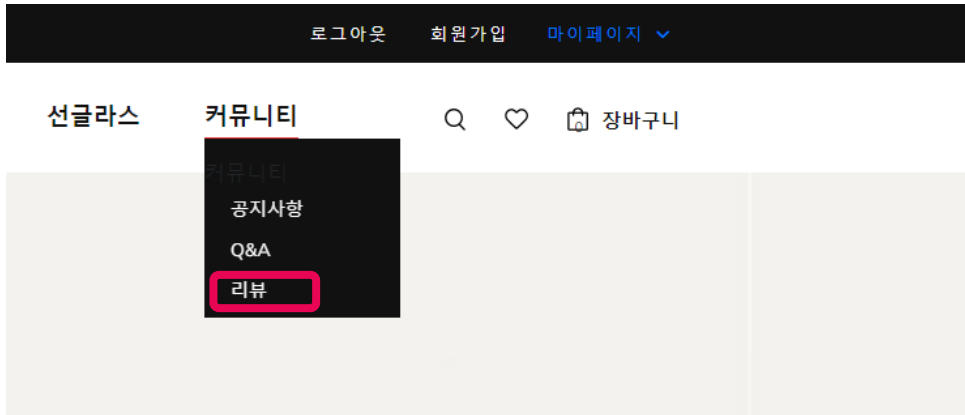
# 04 프로젝트 수행 결과

## Q&A



# 04 프로젝트 수행 결과

## 리뷰



# 04 프로젝트 수행 결과

## 리뷰

리글로우 볼데

리글로우 볼데



★★★★★

리글로우 볼데

리글로우 볼데

Comments (150)

Leave a Comment

Your message\*

SEND MESSAGE

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

리글로우 볼데

```
<c:if test="${dto.mid eq sessionScope.member.mid or sessionScope.member.verify==2}">
  <input type="button" id="btnUpdate" value="수정" style="height:30px; border-radius:10px">
  <input type="button" id="btnDelete" value="삭제" style="height:30px; border-radius:10px">
</c:if>
```

본인이 작성한 글이거나 관리자일 경우에만  
수정, 삭제 버튼이 보이도록 설계

# 04 프로젝트 수행 결과

## 리뷰

```
@RequestMapping(value = "review/reviewWrite", method = RequestMethod.GET)
public void getReviewWrite(HttpSession session, Model model) {

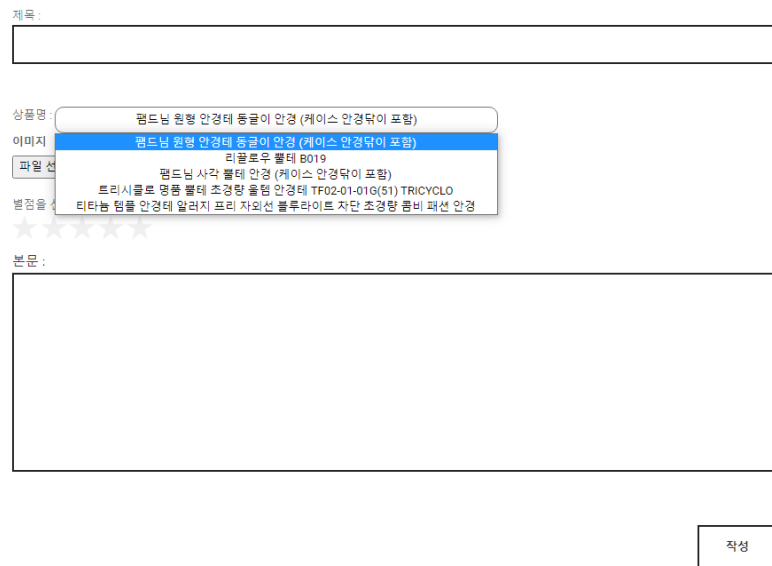
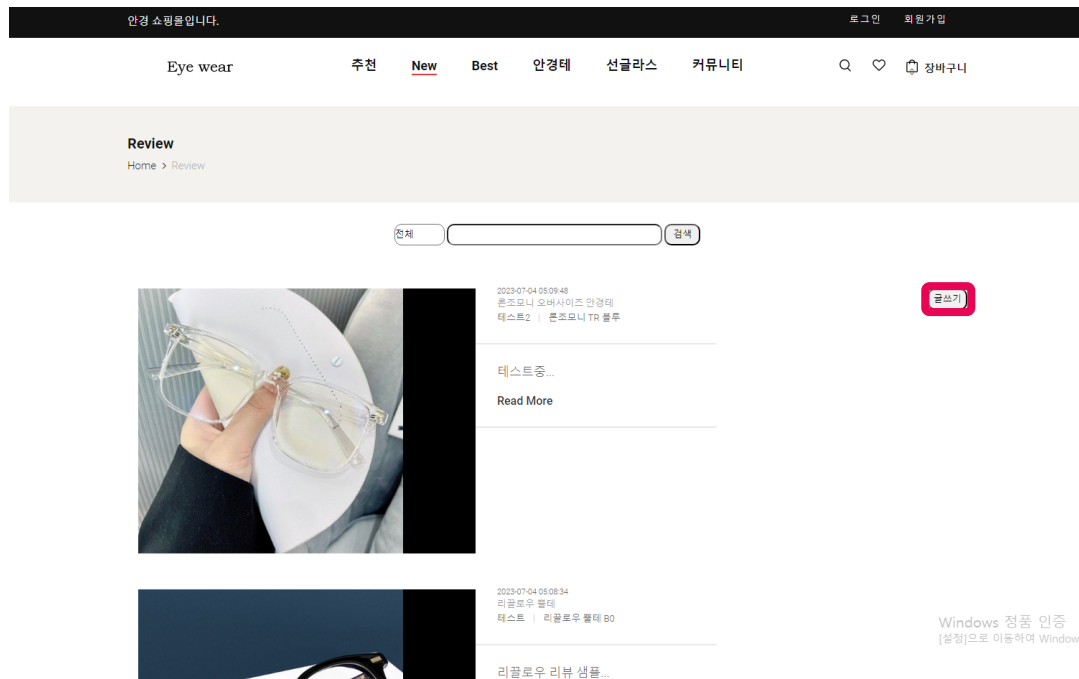
    MemberVO member = (MemberVO)session.getAttribute("member");

    Logger.info("상품등록 목록 페이지 진입");
    List<ProductVO> list = productService.searchpd(member.getMid());
    model.addAttribute("list", list);

}
```

```
상품명 :
<select name="pdnu" style="height:30px; width:500px; border-radius:10px">
    <c:forEach var="row" items="${list }">
        <option value="${row.pdnu }" style="text-align:center">${row.pdname }</option>
    </c:forEach>
</select>
```

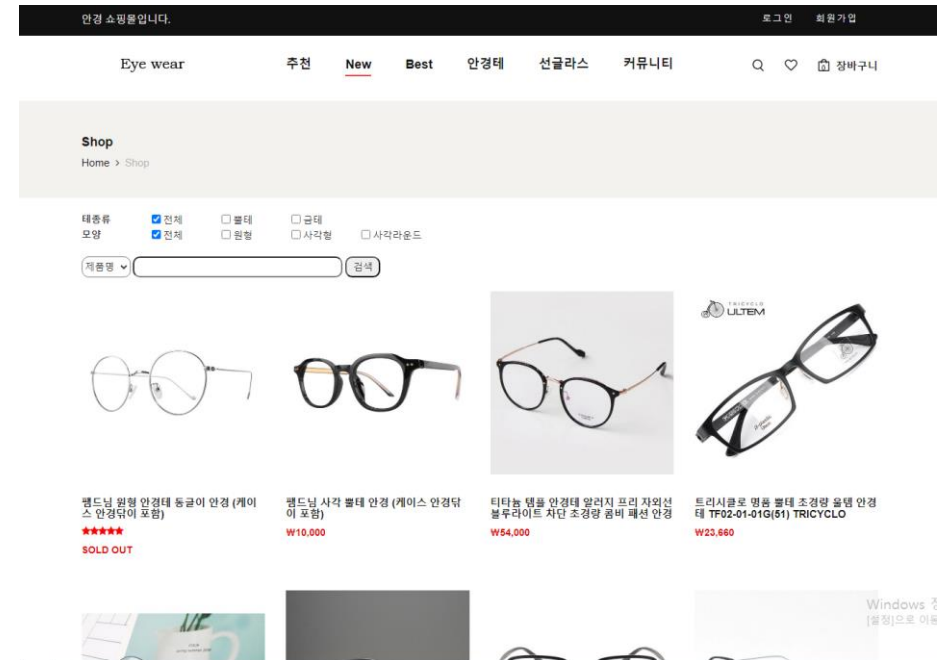
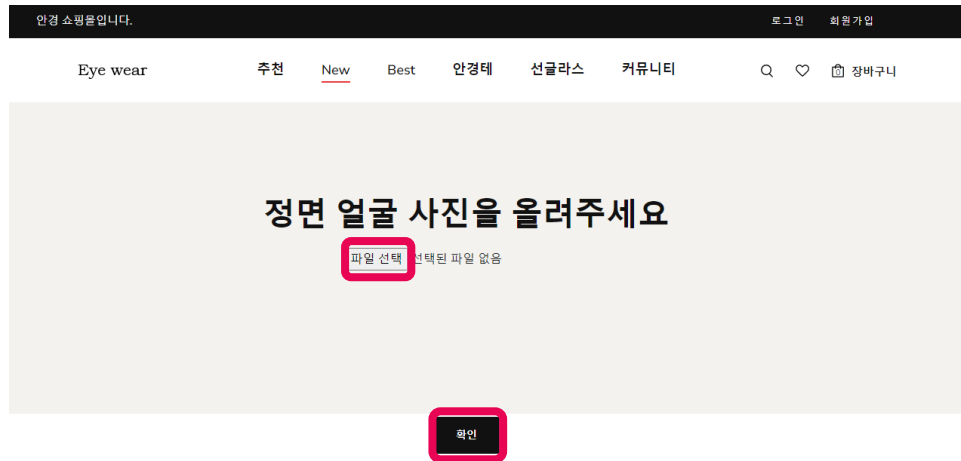
구매한 상품에 대해서만 리뷰를 작성할 수 있게 설계





# 04 프로젝트 수행 결과

## AI추천



# 04 프로젝트 수행 결과

## AI추천

### Controller

```
@RequestMapping("/product/recommend")
public void recommend(@RequestParam(value = "face", required = false) String face, Model model) {
    model.addAttribute("face", face);
}
```

```
@PostMapping("/product/recommend")
public String getRecommend(@RequestParam("images") MultipartFile images, Model model) {
    RestTemplate restTemplate = new RestTemplate();

    try {
        // Encode the image file in base64 format
        String base64Image = encodeImage(images);

        // Create the request body
        MultiValueMap<String, Object> requestBody = new LinkedMultiValueMap<>();
        requestBody.add("img_path", images.getOriginalFilename());
        requestBody.add("data", base64Image);
        requestBody.add("uuid", generateUUID());

        // Create the request headers
        org.springframework.http.HttpHeaders headers = new org.springframework.http.HttpHeaders();
        headers.setContentType(MediaType.APPLICATION_FORM_URLENCODED);

        // Create the HTTP entity with headers and body
        HttpEntity<MultiValueMap<String, Object>> requestEntity = new HttpEntity<>(requestBody, headers);

        // Send the POST request to the Flask server
        ResponseEntity<String> response = restTemplate.postForEntity(flaskServerUrl, requestEntity, String.class);
        // Handle the response if needed
        // ...

        String face = response.getBody();
        System.out.println(face);
        model.addAttribute("face", face);
    } catch (IOException e) {
        e.printStackTrace();
        // Handle exception
        return "Error uploading images.";
    }

    return "redirect:/product/recommend";
}
```

1. Base64로 인코딩한 이미지 값을 flask서버에 전송

3. Response라는 변수로 받은 return값을 읽어서 recommend로 redirect  
return 값은 0, 1, 2, 3으로  
0: 긴 얼굴형  
1: 계란형 얼굴  
2: 둥근 얼굴  
3: 각진 얼굴  
이다.

### detect.py

```
def detect(image, txt, crop, exp):
    run(source=image, save_txt=txt, save_crop=crop, name=exp)
```

### detector\_win.py

```
prev_id = ''
app = Flask(__name__)

@app.route('/detect', methods=['POST'])
def detect_target():
    global prev_id

    file_name = request.form.get('img_path')
    img = request.form.get('data')

    # Check if both file_name and img are not None
    ret_img_string = img
    img = base64.b64decode(ret_img_string)

    # Save the image file to disk
    base_directory = "/home/ubuntu" # Replace with the desired directory
    file_path = os.path.join(base_directory, file_name)
    with open(file_path, 'wb') as file:
        file.write(img)

    uuid = request.form.get('uuid')
    print(file_name)

    # Perform the necessary processing on the image file
    # (e.g., call the Detect.detect() function)
    Detect.detect(file_path, True, True, uuid)

    prev_id = uuid

    image_url = '/home/ubuntu/yolov5/runs/detect/' + uuid
    image_path = os.path.join(image_url, file_name)

    filename = image_url + '/labels/' + (file_name.split('/')[-1])[0:-4] + '.txt'
    f = open(filename, 'r')
    lines = f.readlines()

    for line in lines:
        face = line.split(' ')[0]
        break

    if os.path.isfile(image_path):
        os.remove(image_path)

    if os.path.exists(image_url):
        shutil.rmtree(image_url)

    return face

if __name__ == '__main__':
    app.run(host='0.0.0.0')
```

2. Base64형태의 이미지 값을 다시 디코딩해서 이미지 분석, label폴더 안에 클래스와 크롭한 좌표값을 저장한 txt파일의 클래스값을 읽어서 return값으로 전송

## 05 자체 평가 의견

---

소셜 로그인 등의 API기능이 없고, 주문 기능은 실제 결제가 되는 것이 아니며, 공지사항, Q&A 기능은 미완성이다.

리뷰 기능도 한 상품에 여러 번 리뷰를 작성 할 수 있다는 문제점이 있고, AI기능은 각진 얼굴의 데이터 비율이 커서 대부분의 경우 각진 얼굴로 인식하는 문제가 있다.

여러모로 부족한 부분이 있는 프로젝트였지만 프로그램의 전체 흐름과 파일 저장, 다른 서버로의 데이터 전송, AWS 연결 등 모르던 기능들을 공부할 수 있는 기회가 되었다.

# 06 개인 추가

## 네이버 로그인

식별 아이디를 받아서 mid를 조회



Verify값을 3으로 구분해서 일반로그인이 안되게 처리

```
// 로그인 데이터 사용자 -> 서버로
@RequestMapping(value = "member/login", method = RequestMethod.POST)
public String loginPOST(MemberVO vo, HttpServletRequest req) throws Exception {
    logger.info("로그인 데이터 전송 완료");
    HttpSession session = req.getSession();
    MemberVO login = memberService.login(vo);

    if(login == null) {
        session.setAttribute("member", null);
    } else {
        if(login.getVerify() != 3) {
            session.setAttribute("member", login);
        }
    }
    return null;
}
```

# 06 개인 추가

## 네이버 로그인

한국어 ▾

NAVER

Eye wear 추천 **New** Best 안경테 선글라스 커뮤니티 🔍 ❤️ 🏠 장바구니

Login  
Home > Login

로그인

아이디\*

비밀번호\*

로그인 회원가입

**N** 네이버 아이디로 로그인

NAVER

ID 로그인 [1] 일회용 번호 QR코드

네이버에 로그인하여 **eyewear** 서비스를  
이용하실 수 있습니다.

공용 PC에서 사용하시는 경우 보안을 위해 서비스 이용 후  
네이버에서도 반드시 로그아웃해 주세요.

👤 아이디

🔑 비밀번호

✔ 로그인 상태 유지 IP보안 ☒

로그인

비밀번호 찾기 | 아이디 찾기 | 회원가입

# 06 개인 추가

## 네이버 로그인



Join

Home > Join

### 회원가입

아이디\*

중복확인

전화번호\*

☐ **[필수]** 개인정보 수집 및 이용

개인정보보호법에 따라 회원가입 신청하시는 분께 수집하는 개인정보의 항목, 개인정보의 수집 및 이용목적, 개인정보의 보유 및 이용기간, 동의 거부권 및 동의 거부 시 불이익에 관한 사항을 안내 드리오니 자세히 읽은 후 동의하여 주시기 바랍니다.

정정

가입

첫 로그인 시 가입 절차 후 바로 로그인

Eye wear

추천

New

Best

안경테

선글라스

커뮤니티

Q

♡

장바구니

jbk0607님 환영합니다.

마이페이지



회원정보 수정

관심상품

주문조회

내가쓴글

탈퇴하기

# 06 개인 추가

## 네이버 로그인

1. Callback에서 정보를 받은 후 naverLogin으로 바로 이동
2. DB에 정보가 없을 경우 가입화면, 아닐경우 로그인

```
1
<script type="text/javascript">
var naver_id_login = new naver_id_login("clientId", "http://localhost:8080/member/callback");
// 접근 토큰 값 출력
var access_token = naver_id_login.oauthParams.access_token;
// 네이버 사용자 프로필 조회
naver_id_login.get_naver_userprofile("naverSignInCallback()");
// 네이버 사용자 프로필 조회 이후 프로필 정보를 처리할 callback function
function naverSignInCallback() {
    var id = naver_id_login.getProfileData('id');
    var email = naver_id_login.getProfileData('email');
    var name = naver_id_login.getProfileData('name');
    var mobile = naver_id_login.getProfileData('mobile');
    document.write("<form id='form1' action='/member/naverLogin' method='post'><input type='hidden' name='access_token' value='" + access_token + "'");
    document.write("<input type='hidden' name='email' value='" + email + "'>");
    document.write("<input type='hidden' name='id' value='" + id + "'>");
    document.write("<input type='hidden' name='name' value='" + name + "'></form>");
    document.getElementById("form1").submit();
}
</script> 2

@RequestMapping(value = "/member/naverLogin", method = RequestMethod.POST)
public String naverLogin(naverVO naver, RedirectAttributes redirectAttributes, HttpServletRequest req) {
    int access = memberService.naverChk(naver.getId());
    // Perform any necessary logic here
    if(access==0) {
        redirectAttributes.addFlashAttribute("naver", naver);
        return "redirect:/member/naverJoin";
    } else {
        MemberVO login = memberService.naverLogin(naver);
        HttpSession session = req.getSession();
        if(login == null) {
            session.setAttribute("member", null);
        } else {
            session.setAttribute("member", login);
        }
        return "redirect:/member/login";
    }
}
```

# 06 개인 추가

## 네이버 로그아웃

네이버 로그아웃 팝업을 띄우는 것으로 네이버 로그아웃 후 logout 컨트롤러로 이동해 session 삭제

```
<script>
var naverLogin = new naver.LoginWithNaverId(
    {
        clientId: "clientId", //내 애플리케이션 정보에 cliendId를 입력해줍니다.
        callbackUrl: "http://localhost:8080/member/callback", // 내 애플리케이션 API설정의 Callback URL 을 입력해줍니다.
        isPopup: false,
        callbackHandle: true
    }
);
var naverPopUp;
function openPopUp() {
    naverPopUp= window.open("https://nid.naver.com/nidlogin.logout", "_blank", "toolbar=yes,scrollbars=yes,resizable=yes,width=1,height=1");
}
function closePopUp(){
    naverPopUp.close();
}

function naverLogout() {
    openPopUp();
    setTimeout(function() {
        closePopUp();
        location.href = "/member/logout";
    }, 100);
}
</script>
```





# 감사합니다!

김정빈



+82 10 9268 1882



tw1300@naver.com



우리인재개발원

