

A photograph of several runners in motion on a paved path. The runners are blurred, suggesting speed. They are wearing various athletic gear like tank tops, shorts, and sunglasses. The background is a bright, overexposed sky.

React Project ANDAR Web Publishing - 방혜림

Table of Contents



- 01 타임라인 Time Line
- 02 주제선정 Select a Topic
- 03 기획 Project
- 04 구현 Rendering
- 05 마무리 Reflection



Time Line



6월 JUNE

01

기획

25.06.01 - 06.06

■ 주제 선정

■ 페르소나

■ 와이어 프레임

■ 프로토 타입

02

데이터 작성

25.06.09 - 06.10

■ Womens

■ Mens

■ Kiz ■ Best

■ New

03

MAIN 구현

25.06.10 - 06.13

■ 코드 구현

04

SUB 구현

25.06.13 - 06.18

■ 코드 구현

02

주제선정



SNS에 건강과 운동에 대한 긍정적인 피드가 늘면서 다양한 세대층에서 건강과
다이어트 관련한 관심이 늘게 되었고 자연스레 건강과 다이어트 관련한 SNS채널이
다양하게 형성, 다양한 소비자에게 다양한 공급이 늘면서
새로운 브랜드가 많이 창출되고 있어 애슬레저 브랜드로 주제 선정

애슬레저 브랜드

- React
- Redux
- Swiper
- Bootstrap
- Scss

안다르

젝시믹스

뮬라웨어

아디다스

나이키

애슬레저 제품

레깅스

요가복

스포츠 이너웨어

운동복

일상복 겸용

마켓팅 타겟

2030 세대

직장인)

인플루언서

바디프로필

키워드

편안함

기능성 패브릭

심플

자기관리

다양한 핏

일상 복 겸용

세련됨

바디프로필

03

기획

#페르소나

P
r
o
f
i
l
e

배수지 30세 직장인

“ 일상복처럼 입을 수 있고
기능성이 있었으면 좋겠어요 ”

■ Needs

일상복처럼 입을 수 있는 운동복

가볍고 기능성 원단 사용

트렌디한 디자인

■ Story

바쁜 직장 생활 속 운동복을 깜빡하기도 한다

일상복처럼 입을 수 있고 가벼워서 휴대가 편하고

흡수, 흡건, 구김방지로 입고 세탁하는데 불편함이 적고

트레디한 디자인으로 '오운완' 사진도 남기고 싶어한다

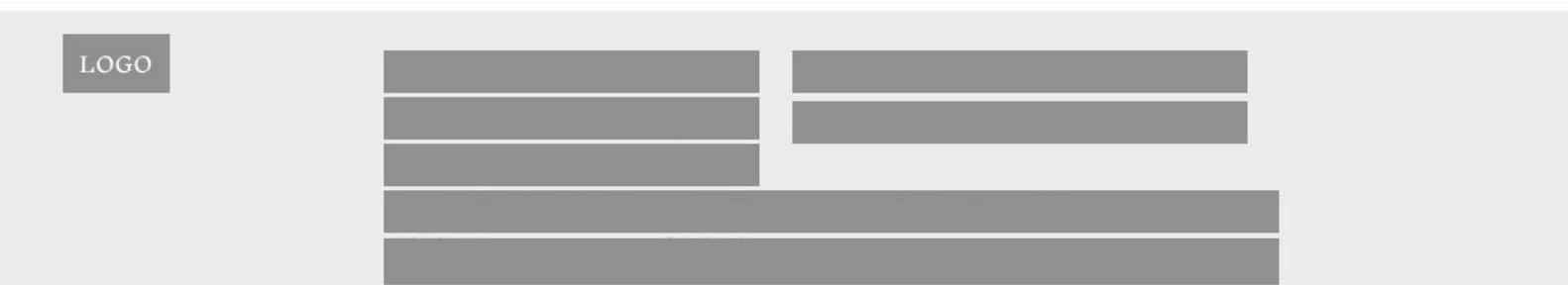
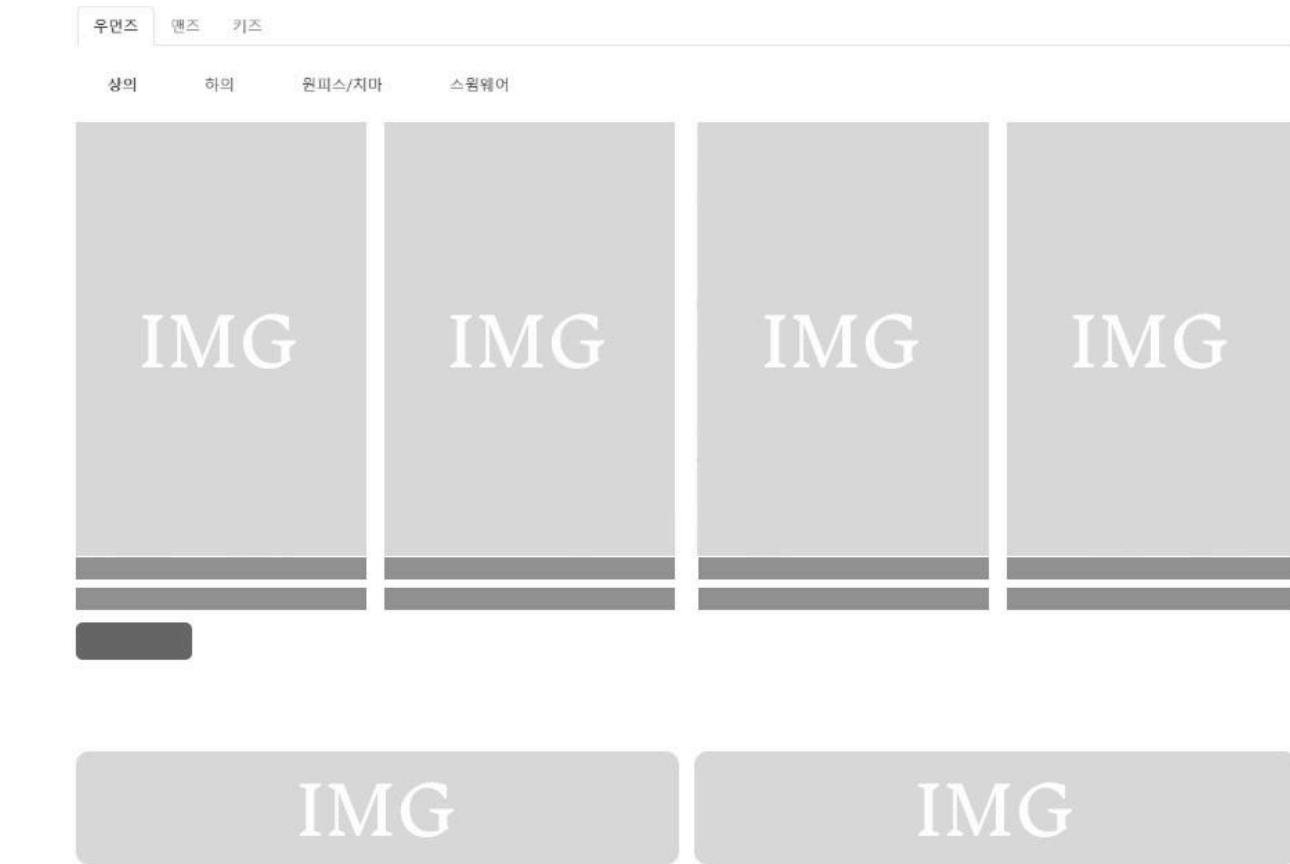
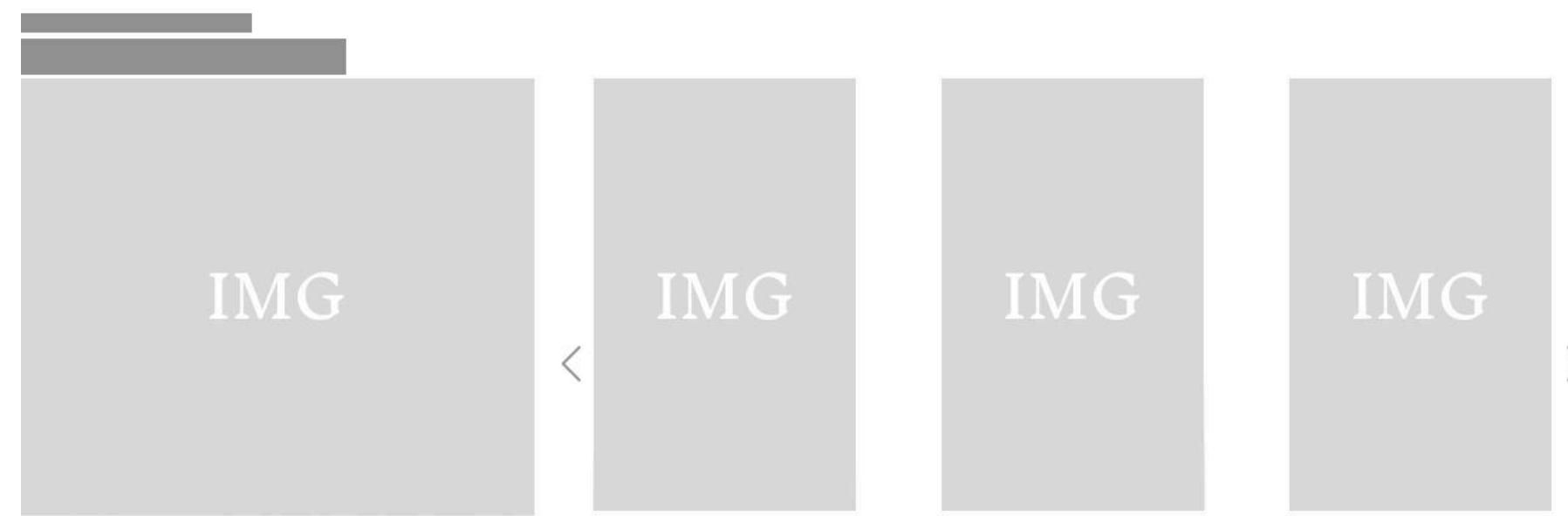
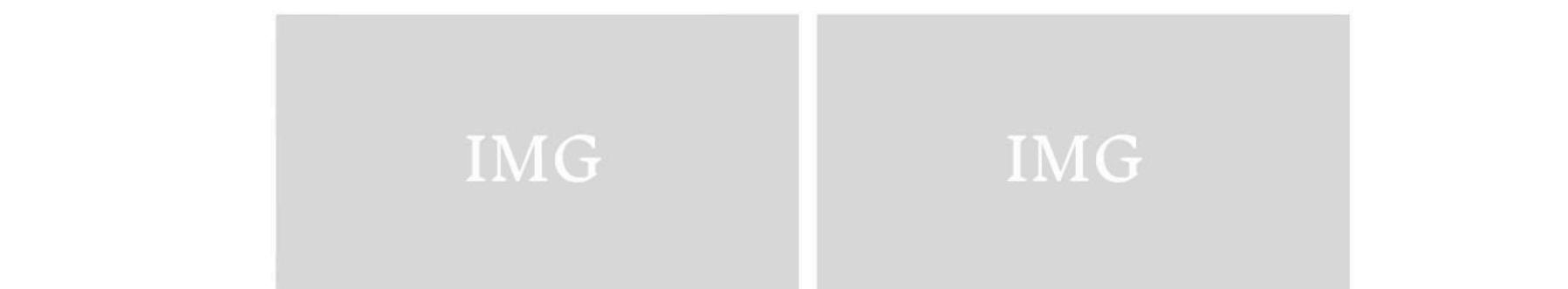
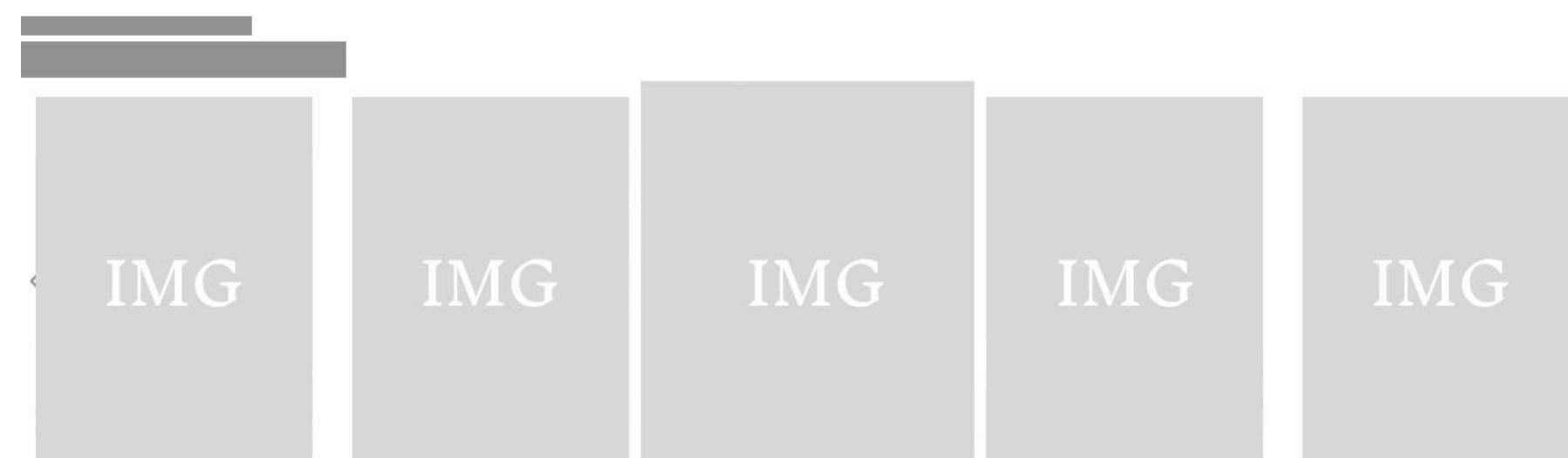
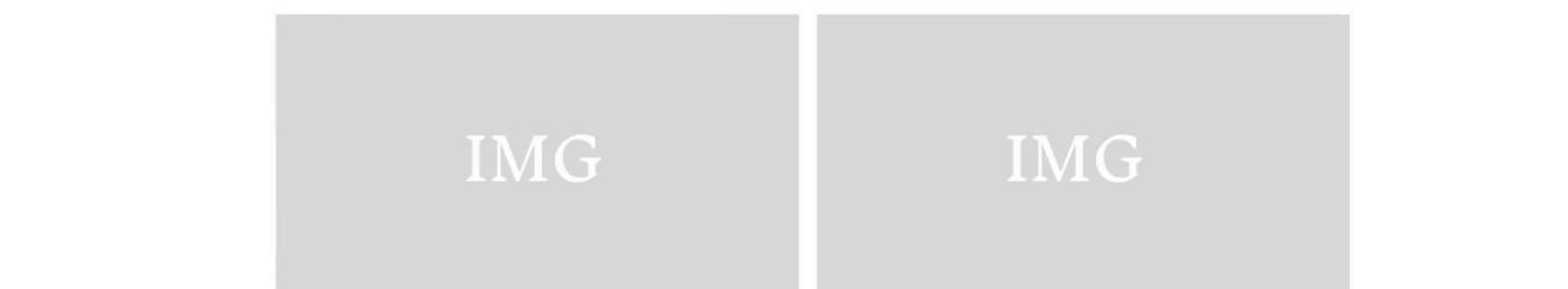
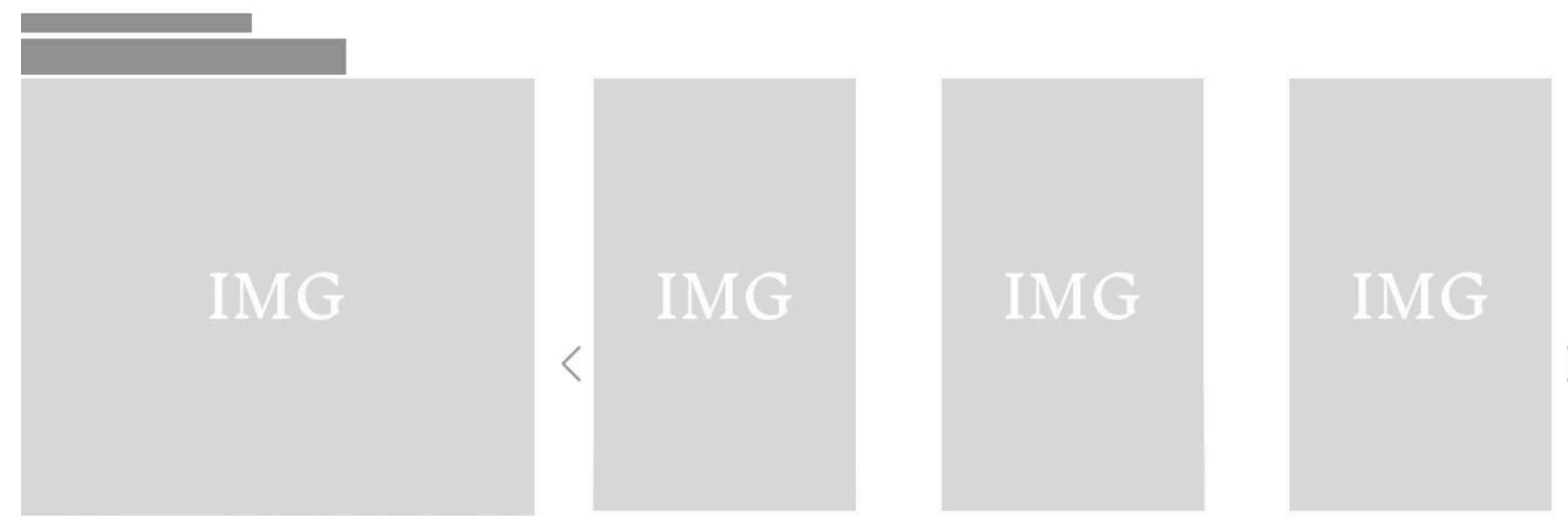


03

기획

■ 와이어프레임

└ MAIN

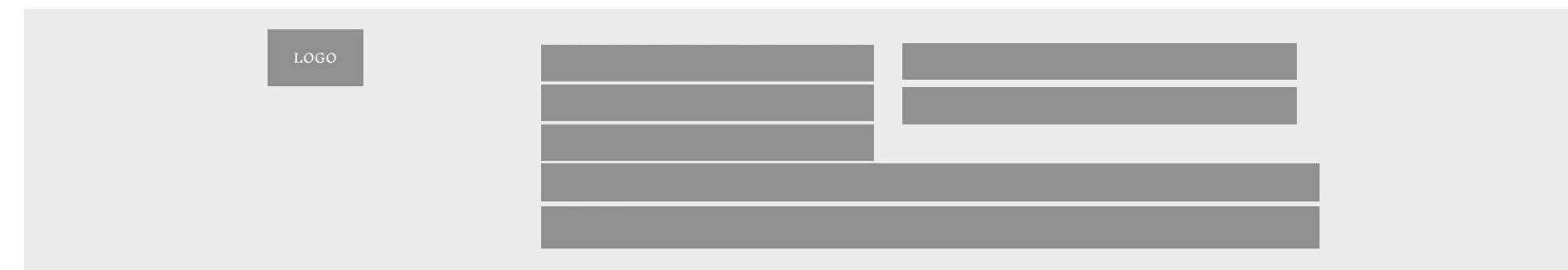
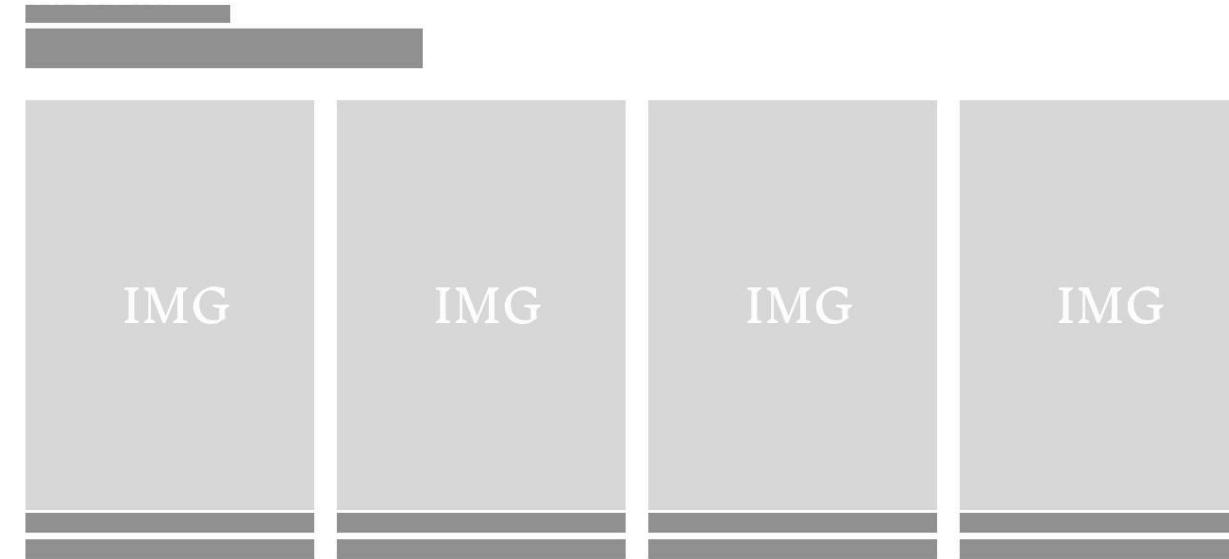


03

기획

■ 와이어프레임

- └ Sub
- └ 상품 그리드 목록
 - 디자인 동일하게 구성
- └ 신상, 베스트, 우먼즈
맨즈, 키즈



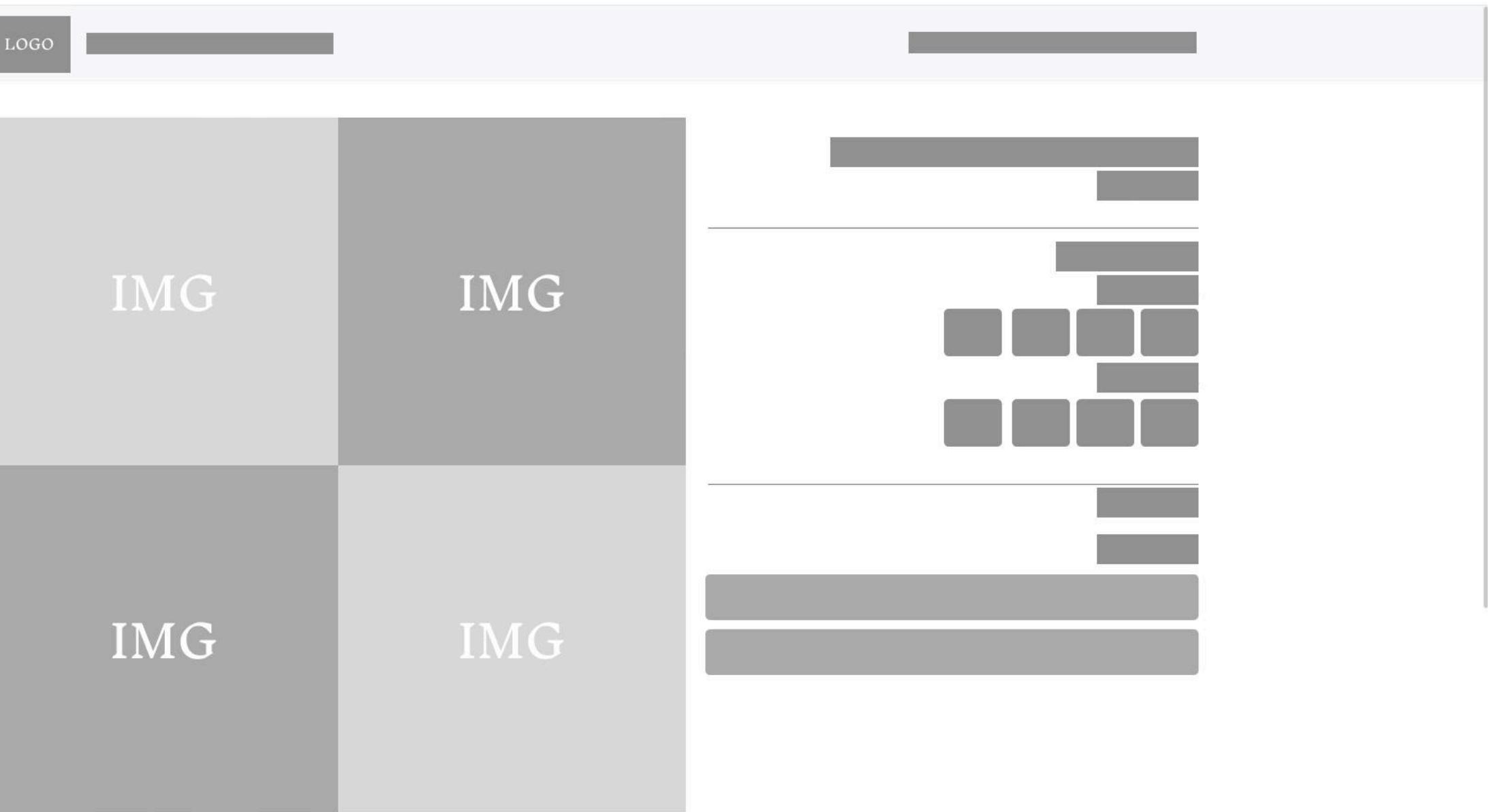
03

기획

■ 와이어프레임

- └ Sub
- └ 상품 상세보기
- └ 장바구니

Product Detail



Cart



03

기획

■ 프로토 타입

└ MAIN

└ Bootstrap

└ Swiper

andar 신상 베스트 우먼즈 맨즈 키즈

장바구니 마이페이지 로그인 회원가입

NEW
안다르 25 SS 신상

* 25 SS신상은 한 달마다 NEW 아이콘을 확인해주세요.

WOMENS MENS KIZ

25SS NEW ITEM
주목해야 할 안다르 신상

더위까지 시원하게 풀스윙,
여름 기능성 골프웨어
골하고 쾌적하게 라운딩을 즐겨보세요.

어반액티브 맨즈 하프 팬츠 65,000
어반액티브 베유다 쇼츠 69,000
아이스프레시 맨즈 스윙 폴로 숏슬리브 59,000

전국 백화점 및
프리미엄 아울렛 매장 보기

자세히 보기

andar's 시그니처 원단

SHOP BY ACTIVITY
운동별 추천 스타일

아웃도어
Golf
Running
Swimwear
Pilates/Yoga

LOOK AROUND ANDAR
다양한 안다르 둘러보기

우먼즈 맨즈 키즈

상의 하의 원피스/치마 스웨트

에어리션셀 숏슬리브 크롭티 39,000
소프트 모달 숏슬리브 커버업 39,000
컴피니티 멜란지 크롭탑 39,000
아이스 오픈백 숏슬리브 크롭티 25,000

상품 더 보기

이미 안다르 회원이라면?
멤버십 전용 혜택 확인 5,000원

안다르가 처음이라면?
1만 원 할인 쿠폰 받기 10,000원

BEST SELLERS
가장 사랑받는 베스트 아이템

우먼즈 맨즈 키즈 홍트 용품 & ACC

에어클링 지니 시그니처 레깅스 59,000
컴퍼린네 와이드 팬츠 79,000
에어리끗 와이드 팬츠 59,000
에어리끗 오버핏 티셔츠 63,000

상품 더 보기

STYLE BY INSTAGRAM
인스타그램 안다르

#안다르 #레깅스 #조거팬츠 #아이스크림레이스
#안다르 #에어리끗와이드 #45부리걸스 #산책
#안다르 #에어리끗3부 #연인합 #일상
#안다르 #예술가자제 #골프 #아이스크림레이스

마슬레이저 1위, 4년 연속 대한민국 패션 브랜드 대상 스포츠 애슬레저
부문
고객센터 | 1566-378
평일 AM 10:00 ~ PM 05:30 / 주말, 공휴일 휴무
점심 PM 12:00 ~ PM 01:00

주식회사 안다르 | 대표이사 : 김철홍, 공설아 | 사업자등록번호: 103-88-00990 | 통신판매업신고: 2024-서울승파-3789 |
주소: 05510 서울 송파구 올림픽로 35(다산 42) 한국무터회관 9, 11, 12층 | TEL: 1566-7318 | 호스팅제공자: 카카오(주) | 이용 문의: help@andar.co.kr |
마케팅제공: 퓨얼리파트너십@andar.co.kr | 해외수출 문의: overseas@andar.co.kr

Copyright by 안다르, all rights reserved.

03

기획

■ 프로토 타입

└ Sub

└ 상품 그리드 목록

디자인 동일 구성

└ 신상, 베스트,

우먼즈, 맨즈, 키즈

andar 신상 베스트 우먼즈 맨즈 키즈 장바구니 마이페이지 로그인 회원가입

NEW
안다르 25 SS 신상

* 25 SS 신상은 썸네일 위 NEW 아이콘을 확인해주세요.

담을수록 커지는 증정혜택
10만원 이상 구매 시
리유저블 소퍼백 L
15만원 이상 구매 시 (택 1)
나이克斯 / 맨즈 섹스
20만원 이상 구매 시
메쉬 포켓 비치백

25SS NEW ITEM
주목해야 할 앤다르 신상

어반액티브 맨즈 하프 팬츠
65,000원
어반액티브 버뮤다 쇼츠
69,000원
아이스프레시 맨즈 스윙 폴로 숏
슬리브
59,000원
에어리핏 PK 폴로 슬리브리스
51,000원

아이스 에어데님 와이드 청바지
82,000원
아이스 에어데님 맨즈 스트레이트
핏 청바지
82,000원
에어리핏 맨즈 폴로 티셔츠
94,000원
우븐 브리즈 슬리브리스 롱 원피스
69,000원

에어리스트 맨즈 아이스 슬랙스
79,000원
여름 출근룩 베스트 팩
139,000원
에어프라임 맨즈 슬랙스
129,000원
에어리핏 와이드 크롭 팬츠
55,000원

andar 애슬레저 1위, 4년 연속 대한민국 패션종합대상 스포츠 애슬레저
부문
고객센터 | 1566-7378 | 평일 AM 10:00 ~ PM 05:30 / 주말, 공휴일 휴무 |
점심 PM 12:00 ~ PM 01:00
주식회사 앤다르 | 대표이사 : 김철웅, 공성아 | 사업자등록번호: 103-88-00990 | 통신판매업신고: 2024-서울송파-3789 |
마케팅제휴 문의: partnership@andarc.co.kr | 해외수출 문의: overseas@andarc.co.kr
Copyright by 앤다르, all rights reserved.

BEST SELLERS
가장 사랑받는 베스트 아이템

유연즈 맨즈 키즈
품토 용품 & ACC

미니 숄더 백
69,000원
노슬립 퍼포먼스 토스백
19,000원
에어풀라이 투웨이 프리진 백 3.0
35,000원
UV 프로텍션 골프 마스크(블랙)
23,000원

1&1 저스트풀라이(우먼즈)
119,000원
하이 서포트 팔꿈치 보호대
24,000원
서포트 허리 보호대
24,900원
하이 서포트 무릎 보호대
34,000원

엑티브 라이드 백
65,000원
올라운드 베킷셋
65,000원
팀스탑 포켓 숄링백
55,000원
백연슬링 2way 백
39,000원

ANDAR WOMENS
테크니컬 애슬레저 앤다르 우먼즈

상의 하의 원피스/치마 스윔웨어

에어리션셀 숏슬리브 크롭티
39,000원
소프트 모달 숏슬리브스 캐비엄
39,000원
컴파니티 멀란지 크롭탑
39,000원
아이스 오른백 숏슬리브 크롭티
28,000원

에어리핏 슬리브리스
39,000원
에어리핏 PK 스탠다드 웜 솔 슬리브
51,000원
에어리핏 스텐다드핏 웜 솔 슬리브
41,000원
에어리핏 웜 솔리브리스
39,000원

에어리핏 슬림핏 슬리브리스 크롭
39,000원
에어리핏 오버핏 품슬리브
41,000원
에어리핏 오버핏 숏슬리브
41,000원
에어리핏 PK 풀로 슬리브리스
83,000원

ANDAR MENS
테크니컬 애슬레저 앤다르 맨즈

상의 하의 비즈니스 애슬레저 스윔웨어

에어리핏 맨즈 풀로 숏슬리브
49,000원
에어리핏 맨즈 포인트 풀로 숏스리브
59,000원
아이스프레시 맨즈 스웨 풀로 숏슬리브
59,000원
에어리핏 맨즈 풀로 티셔츠
59,000원

시그니처 로고 볼로우터치 맨즈
숏슬리브
49,000원
아이스프레시 맨즈 숏슬리브
47,000원
클리어스 맨즈 숏슬리브
39,000원
에어리핏 맨즈 오션 카라 숏슬리브
69,000원

큐브 부스트 맨즈 라닝 싱글랫
59,000원
브리즈핏 맨즈 슬리브리스
79,000원
에어리핏 맨즈 오션 카라 숏슬리브
49,000원
실렉션 스테이너블 맨즈 스트라이프
43,000원

ANDAR KIDS
프리미엄 키즈 애슬레저

공용 상의 공용 하의 여아

키즈 언더다이어 풀로우터치 & 반팔티
99,000원
키즈 언더다이어 그레이픽 반팔티
49,000원
키즈 멜로우터치 캐시미어
79,000원
키즈 에어리핏 PK 반팔티
69,000원

키즈 멜로우터치 오버핏 반팔티
49,000원
키즈 에어리핏 PK 반팔티
65,000원
키즈 포켓 반팔티
49,000원
키즈 플레이타임 페인트
59,000원

키즈 그레이데이션 반팔티
59,000원
키즈 레깅린 7부 티셔츠
45,000원
키즈 라이트 우븐 반팔 침엽
129,000원
키즈 엔드우터치 반팔 멘투멘
49,000원

03

기획

■ 프로토 타입

- └ Sub

- └ 상품 상세보기

- └ 장바구니

상품 금액	0원
배송비	3,000원
전체 합계	3,000원



기획

■ DATA

- └ 데이터 작성
- └ REDUX

활용하여 적용

REDUX DATA

```
JS best.js  
JS cart.js  
JS kiz.js  
JS mens.js  
JS newitems.js  
JS womens.js
```

■ REDUX DATA
직접 만든 데이터를 REDUX
를 통해 적용 및 활용

```
import { createSlice } from "@reduxjs/toolkit"

const initialState = [
    // 우먼즈
    {
        id: 0,
        catagory: 'womens',
        imgUrl: '/img/womens/wbest1.jpg',
        title: '에어클링 지니',
        price: 59000,
        proImg: ['/img/womens/wbest1-1.jpg', '/img/womens/wbest1-2.jpg'],
    },
    {
        id: 1,
        catagory: 'womens',
        imgUrl: '/img/womens/wbest2.jpg',
        title: '컴파린네 와이드 팬츠',
        price: 79000,
        proImg: ['/img/womens/wbest2-1.jpg', '/img/womens/wbest2-2.jpg'],
    },
    {
        id: 2,
        catagory: 'womens',
        imgUrl: '/img/womens/wbest3.jpg',
        title: '에어리핏 와이드 팬츠',
        price: 59000,
        proImg: ['/img/womens/wbest3-1.jpg', '/img/womens/wbest3-2.jpg'],
    },
    {
        id: 3,
        catagory: 'womens',
        imgUrl: '/img/womens/wbest4.jpg',
        title: '에어리핏 오버핏 티셔츠',
        price: 63000,
        proImg: ['/img/womens/wbest4-1.jpg', '/img/womens/wbest4-2.jpg'],
    },
]
```

```
const navigate = useNavigate();
const [visibleCount, setVisibleCount] = useState(4)

const Womens = () => {
    const womens = useSelector(state => state.womens)
```

■ 카테고리 값을 넣어
카테고리 필터링에 용이하게
데이터 작성

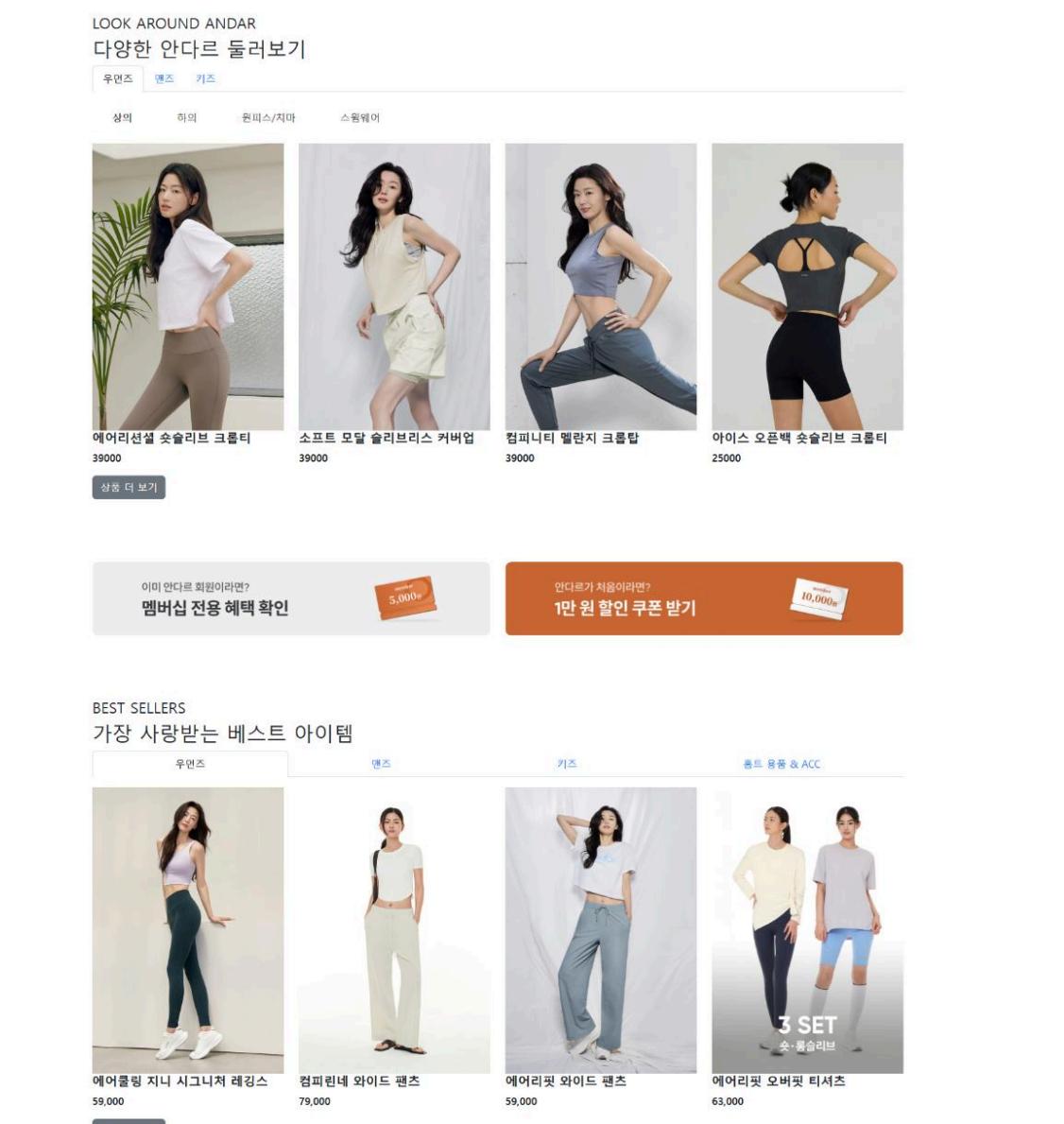
■ 상품 그리드가 출력되는 컨텐츠와
서브 페이지에서 데이터를
연결하여 활용

MAIN DETAIL PRODUCT BOOTSTRAP

■ REACT BOOTSTRAP
리액트 부트스트랩 TAB 활용하여
코드 구성 및 CSS 시간을 단축

■ COMPONENT

컴포넌트를 분리하여 코드를 가독성 있게 작성
카테고리 필터링 재활용 및 코드를 재활용하여
각 카테고리별 작성 시간을 단축



구현

■ MAIN 코드 구현

MAIN DETAIL PRODUCT CODE

```

return (
  <div style={padding}>
    <Container>
      <Row>
        <div className="title">
          <h4>LOOK AROUND ANDAR</h4>
          <h2>다양한 앤다르 둘러보기</h2>
        </div>
      </Row>
      <Row>
        <Col>
          <Tabs defaultActiveKey="womens" id="fill-tab-example" className="mb-3" fill>
            <Tab eventKey="womens" title="우먼즈">
              <BestContent key="womens" best={best} category="womens" />
            </Tab>
            <Tab eventKey="mens" title="맨즈">
              <BestContent key="mens" best={best} category="mens" />
            </Tab>
            <Tab eventKey="kiz" title="키즈">
              <BestContent key="kiz" best={best} category="kiz" />
            </Tab>
            <Tab eventKey="hometraining" title="홈트 유품 & ACC">
              <BestContent key="hometraining" best={best} category="hometraining" />
            </Tab>
          </Tabs>
        </Col>
      </Row>
      <Row>
        <Col xs={6} lg={3} key={item.id} onClick={() => { navigate('/detail/mens/' + item.id) }} style={{ cursor: 'pointer' }}>
          <div className="tab_content">
            <img src={process.env.PUBLIC_URL+item.imgUrl} alt={item.title} />
            <h5>{item.title}</h5>
            <p>{item.price.toLocaleString()}</p>
          </div>
        </Col>
      </Row>
      <Row>
        <Col>
          {visibleCount < filtered.length && (
            <Button variant="secondary" onClick={() => setVisibleCount(v => v + 4)}>
              상품 더 보기
            </Button>
          )}
        </Col>
      </Row>
    </Container>
  </div>
)

```

■ BEST CONTENT

MAIN DETAIL PRODUCT CODE

```

const Mens = () => {
  const navigate = useNavigate();
  const mens = useSelector(state => state.mens);

  const [category, setCategory] = useState('t') // 기본값: 상의
  const [add, setAdd] = useState(4)

  const handleCategory = (cat) => {
    setCategory(cat)
    setAdd(4)
  }

  const categoryTabs = [
    { key: 't', label: '상의' },
    { key: 'p', label: '하의' },
    { key: 'b', label: '비즈니스 애슬레저' },
    { key: 's', label: '스윔웨어' }
  ]

  const filtered = mens.filter(item => item.category === category)
  const visibleItems = filtered.slice(0, add)

  <div className="tab_list">
    <Row>
      <Col>
        {/* 카테고리 탭 */}>
        <ul className="tab_detail">
          {categoryTabs.map(tab => (
            <li key={tab.key} onClick={() => handleCategory(tab.key)} style={{ cursor: 'pointer', fontWeight: category === tab.key ? 'bold' : 'normal' }}>
              {tab.label}
            </li>
          ))}
        </ul>
      </Col>
    </Row>
    <Row>
      <Col xs={6} lg={3} key={idx} onClick={() => { navigate('/detail/mens/' + item.id) }} style={{ cursor: 'pointer' }}>
        <div className="tab_content">
          {visibleItems.map((item, idx) => (
            <Col xs={6} lg={3} key={idx} onClick={() => { navigate('/detail/mens/' + item.id) }} style={{ cursor: 'pointer' }}>
              <img src={process.env.PUBLIC_URL+item.imgUrl} alt={item.title} />
              <h5>{item.title}</h5>
              <p>{item.price}</p>
            </Col>
          ))}
        </div>
      </Col>
    </Row>
  </div>
)

```

■ BEST TAB

MAIN DETAIL PRODUCT CODE

```

const BestContent = ({ best, category }) => {
  const navigate = useNavigate();
  const [visibleCount, setVisibleCount] = useState(4)

  useEffect(() => {
    setVisibleCount(4)
  }, [category])

  // 'wbest'가 아니라면 카테고리별 필터
  const filtered = category === 'wbest'
    ? best
    : best.filter(item => item.category === category)

  return (
    <Row>
      <Col>
        <Tabs defaultActiveKey="womens" id="fill-tab-example" className="mb-3" fill>
          <Tab eventKey="womens" title="우먼즈">
            <BestContent key="womens" best={best} category="womens" />
          </Tab>
          <Tab eventKey="mens" title="맨즈">
            <BestContent key="mens" best={best} category="mens" />
          </Tab>
          <Tab eventKey="kiz" title="키즈">
            <BestContent key="kiz" best={best} category="kiz" />
          </Tab>
          <Tab eventKey="hometraining" title="홈트 유품 & ACC">
            <BestContent key="hometraining" best={best} category="hometraining" />
          </Tab>
        </Tabs>
      </Col>
    </Row>
  )
}

```

■ 베스트 셀러 탭

MAIN DETAIL PRODUCT CODE

```

const BestContent = ({ best, category }) => {
  const navigate = useNavigate();
  const [visibleCount, setVisibleCount] = useState(4)

  useEffect(() => {
    setVisibleCount(4)
  }, [category])

  // 'wbest'가 아니라면 카테고리별 필터
  const filtered = category === 'wbest'
    ? best
    : best.filter(item => item.category === category)

  return (
    <Row>
      <Col>
        <div style={{ cursor: 'pointer' }}>
          {filtered.slice(0, visibleCount).map(item => [
            <Col xs={6} lg={3} key={item.id} onClick={() => { navigate('/detail/best/' + item.id) }} style={{ cursor: 'pointer' }}>
              <div className="tab_content">
                <img src={process.env.PUBLIC_URL+item.imgUrl} alt={item.title} />
                <h5>{item.title}</h5>
                <p>{item.price}</p>
              </div>
            </Col>
          ])}
        </div>
      </Col>
    </Row>
    <Row>
      <Col>
        {visibleCount < filtered.length && (
          <Button variant="secondary" onClick={() => setVisibleCount(v => v + 4)}>
            상품 더 보기
          </Button>
        )}
      </Col>
    </Row>
  )
}

```

■ 베스트 셀러 컨텐츠

MAIN DETAIL PRODUCT CODE

■ 카테고리 필터링과 상품 더 보기 버튼을 재활용하여 코드 구성 시간 단축 및 작업 효율성을 높임



구현

■ MAIN 코드 구현

MAIN DETAIL PRODUCT

LOOK AROUND ANDAR
다양한 앤다로 둘러보기

- 우먼즈 헌즈 키즈
- 상의 하의 원피스/치마 스윔웨어






상품 더 보기

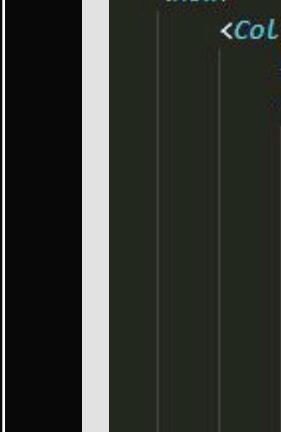
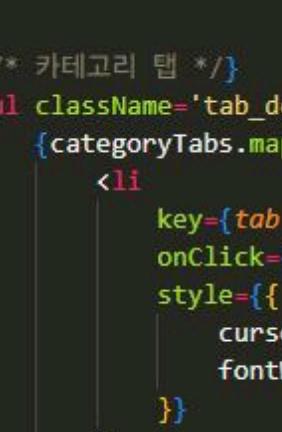
LOOK AROUND ANDAR
다양한 앤다로 둘러보기

- 우먼즈 헌즈 키즈
- 상의 하의 원피스/치마 스윔웨어







상품 더 보기

■ 상품 더 보기
버튼 클릭시 상품이 4개씩 추가되며 slice()
를 활용하여 보여주는 상품 개수를 추가할 수
있는 구조
더 이상 보여줄 상품이 없으면 버튼이 사라진
다

MAIN DETAIL PRODUCT CODE

```
const Womens = () => {
  const womens = useSelector(state => state.womens)
  // 리덕스 데이터
```

```
// 네비게이터
let navigate = useNavigate();
```

```
// 필터링
const [category, setCategory] = useState('t')
const [add, setAdd] = useState(4)
```

```
const handleCategory = (cat) => {
  setCategory(cat)
  setAdd(4)
}
```

```
const filtered = womens.filter(item => item.category === category)
const visibleItems = filtered.slice(0, add) // 카테고리 필터링
```

```
// 탭 이름과 키값 매칭
const categoryTabs = [
  {key: 't', label: '상의'},
  {key: 'p', label: '하의'},
  {key: 'o', label: '원피스/치마'},
  {key: 's', label: '스윔웨어'}
]
```

■ 상품 더 보기 버튼
클릭시 상품 목록
4개씩 증가

```
<Row>
  <Col>
    /* 카테고리 탭 */
    <ul className="tab_detail">
      {categoryTabs.map(tab => (
        <li key={tab.key} onClick={() => handleCategory(tab.key)} style={{
          cursor: 'pointer',
          fontWeight: category === tab.key ? 'bold' : 'normal',
        }}>
          {tab.label}
        </li>
      ))}
    </ul>
  </Col>
</Row>
```

■ 카테고리 필터링 적용

```
<div className="tab_list">
  <AnimatePresence>
    {visibleItems.map((item, idx) => (
      <Col xs={6} lg={3} key={idx} onClick={() => { navigate('/detail/womens/' + item.id) }} style={{
        cursor: 'pointer',
      }}>
        <motion.div
          initial={{ opacity: 0, y: 20 }}
          animate={{ opacity: 1, y: 0 }}
          exit={{ opacity: 0, y: -20 }}
          transition={{ duration: 0.4 }}>
          <img src={process.env.PUBLIC_URL+item.imgUrl} alt={item.title} />
          <h5>{item.title}</h5>
          <p>{item.price}</p>
        </motion.div>
      </Col>
    ))}
  </AnimatePresence>
</div>
```

■ .map() 사용
코드 간소화

■ 모션 프레임

```
// 디테일 상품보기
```

```
.tab_detail {
  width: 460px;
  height: 50px;
  display: flex;
  justify-content: space-between;
  align-items: center;
  line-height: 30px;
  ul {
    padding: 0 !important;
    margin: 0 !important;
  }
  li {
    color: #444;
  }
}
```

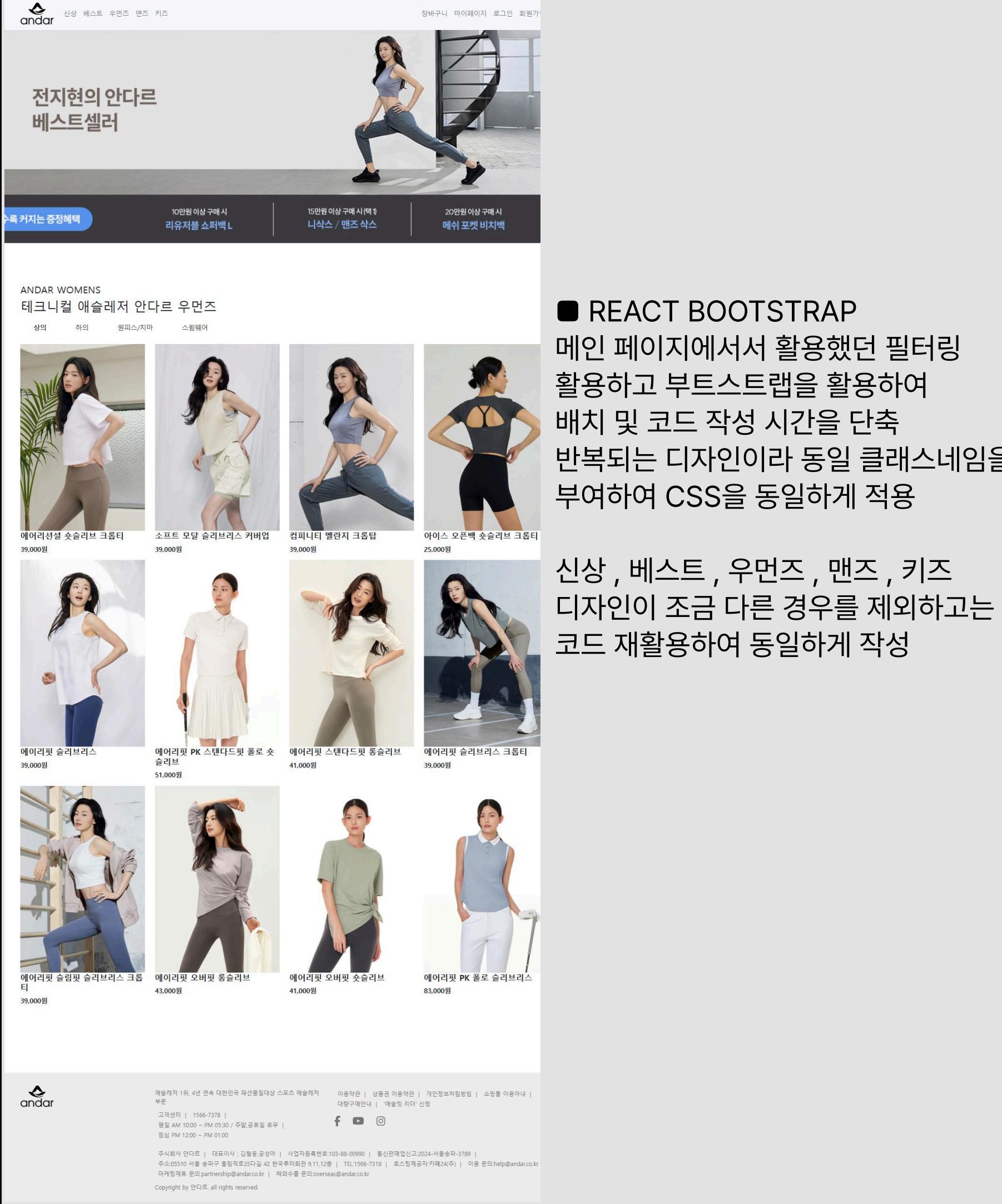
```
.tab_content {
  img {
    width: 100%;
  }
  h5, p {
    font-weight: bold;
  }
}
```

■ 부트스트랩 활용을 통해
CSS를 최소화
SCSS 사용

SUB DETAIL PRODUCT

04 구현

■ SUB 코드 구현



■ REACT BOOTSTRAP
메인 페이지에서 활용했던 필터링
활용하고 부트스트랩을 활용하여
배치 및 코드 작성 시간을 단축
반복되는 디자인이라 동일 클래스네임을
부여하여 CSS를 동일하게 적용

신상, 베스트, 우먼즈, 맨즈, 키즈
디자인이 조금 다른 경우를 제외하고는
코드 재활용하여 동일하게 작성

SUB DETAIL PRODUCT CODE

```
return (
  <div>
    <div className="detatil_top">
      <img src={process.env.PUBLIC_URL + "/img/topslide5.jpg"} alt="top" />
      <img src={process.env.PUBLIC_URL + "/img/detatilbanner.jpg"} alt="banner" />
    </div>

    <Container>
      <div className="title">
        <h4>ANDAR WOMENS</h4>
        <h2>테크니컬 애슬레저 앤다르 우먼즈</h2>
      </div>

      /* 카테고리 탭 */
      <ul className="tab_detail">
        {categoryTabs.map(tab => (
          <li key={tab.key}
            onClick={() => handleCategory(tab.key)}
            style={{
              cursor: 'pointer',
              fontWeight: category === tab.key ? 'bold' : 'normal',
            }}
          >
            {tab.label}
          </li>
        ))}
      </ul>

      /* 상품 리스트 */
      <div className="tab_list">
        <Row>
          <AnimatePresence>
            {womens
              .filter(item => item.catagory === category)
              .map(item => (
                <Col
                  xs={6}
                  lg={3}
                  key={item.id}
                  onClick={() => navigate('/detail/womens/' + item.id)}
                  style={{ cursor: 'pointer' }}
                >
                  <motion.div
                    className="tab_content"
                    initial={{ opacity: 0, y: 20 }}
                    animate={{ opacity: 1, y: 0 }}
                    exit={{ opacity: 0, y: -20 }}
                    transition={{ duration: 0.4 }}
                  >
                    <img src={process.env.PUBLIC_URL + item.imgUrl} alt={item.title} />
                    <h5>{item.title}</h5>
                    <p>{item.price.toLocaleString()}원</p>
                  </motion.div>
                </Col>
              ))}
          </AnimatePresence>
        </Row>
      </div>
    </Container>
  </div>
)
```

```
const DetailW = () => {
  const womens = useSelector(state => state.womens);
  const navigate = useNavigate();
  const [tabParams, setTabParams] = useSearchParams();
  const category = tabParams.get('cat') || 't';

  const handleCategory = (cat) => setTabParams({ cat });
  const categoryTabs = [
    { key: 't', label: '상의' },
    { key: 'p', label: '하의' },
    { key: 'o', label: '원피스/치마' },
    { key: 's', label: '스윔웨어' }
  ];
}
```

//디테일 상품보기

```
.tab_detail {
  display: flex;
  justify-content: space-between;
  padding: 10px;
}

.tab_content {
  padding: 0 !important;
  margin: 0 !important;
}

.tab_content {
  img {
    width: 100%;
  }
  h5, p {
    font-weight: bold;
  }
}
```

■ 상품 목록 그리드 디자인을
통일하여 작업 효율성 높이는데
중점을 맞춰 작업



구현

■ SUB DETAIL 코드 구현

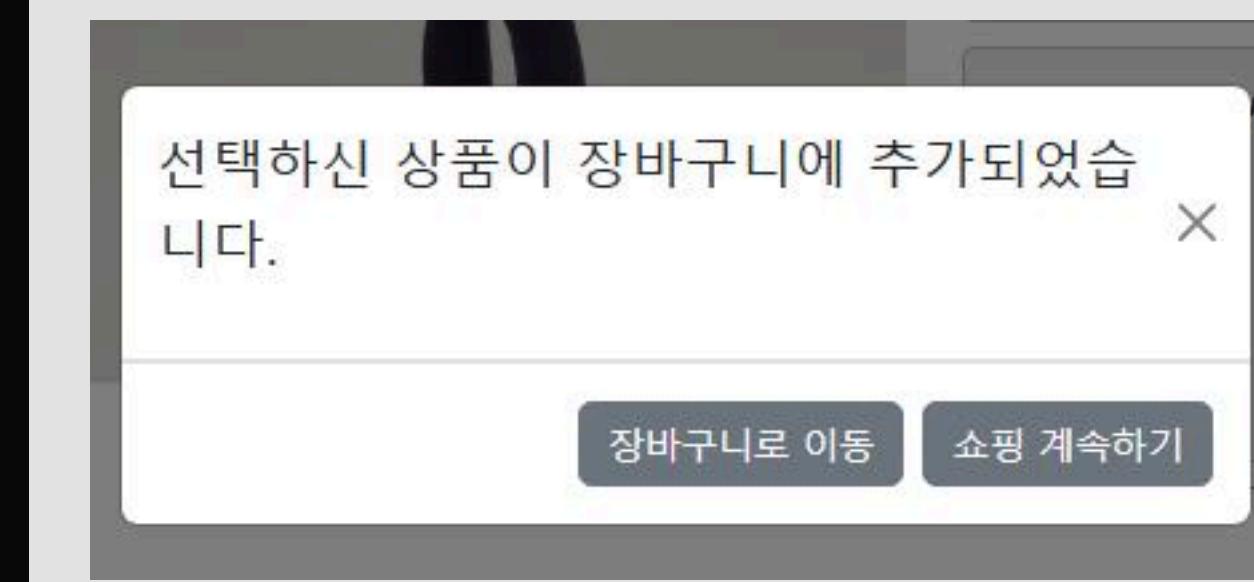
SUB DETAIL PRODUCT

상세보기 페이지
상품 상세보기 컴포넌트를 분리 작성하여 모든 상품들을 한 컴포넌트로 활용할 수 있게 `useParams` 활용

옵션 선택
사이즈와 색상을 모두 선택하면 선택 완료한 상품을 보여주는 div가 출력되며 한가지라도 선택하지 않고 장바구니를 클릭시 '옵션을 선택해주세요.'라는 alert 실행

bhlim08.github.io 내용:
옵션을 선택해주세요.

SUB DETAIL PRODUCT



```
function MyVerticallyCenteredModal(props) {
  const navigate = useNavigate();
  return (
    <Modal {...props} size="md" centered>
      <Modal.Header closeButton>
        <p>선택하신 상품이 장바구니에 추가되었습니다.</p>
      </Modal.Header>
      <Modal.Footer>
        <Button variant="secondary" onClick={() => navigate('/cart')}>장바구니로 이동</Button>
        <Button variant="secondary" onClick={props.onHide}>쇼핑 계속하기</Button>
      </Modal.Footer>
    </Modal>
  );
}
```

■ 장바구니 버튼 클릭시 모달창

```
const productList = [
  new: items.newitems,
  best: items.best,
  womens: items.womens,
  mens: items.mens,
  kiz: items.kiz
];

const item = productList[0];
const addOptionItem = (size, color) => {
  const existIdx = selectedItems.findIndex(i => i.size === size && i.color === color);

  if (existIdx > -1) {
    const updated = [...selectedItems];
    updated[existIdx].count += 1;
    setSelectedItems(updated);
  } else {
    setSelectedItems(prev => [...prev, { ...item, size, color, count: 1 }]);
  }

  setSelSize('');
  setSelColor('');
};

const changeCount = (index, diff) => {
  setSelectedItems(prev => {
    prev.map((item, i) =>
      i === index ? { ...item, count: Math.max(1, item.count + diff) } : item
    );
  });
};

const selItemCart = (item) => {
  setSelectedItems(selectedItems.filter(i => i !== item));
  setModalShow(true);
};

const total = () => {
  const total = selectedItems.reduce((acc, item) => acc + item.price * item.count, 0);
  return total;
};
```

■ 옵션 선택 사이즈, 컬러 선택

```
<div className="product_option_wrap">
  <h6>옵션을 선택해주세요</h6>
  <div className="size">
    <h6>사이즈 Size</h6>
    {[ 'S', 'M', 'L', 'XL' ].map(size => (
      <button key={size}
        variant={selSize === size ? 'dark' : 'light'}
        onClick={() => handleSelectOption('size', size)}
      >{size}</button>
    ))}
  </div>
  <div className="color">
    <h6>색상 Color</h6>
    {[ 'Black', 'Gray', 'White', 'Blue' ].map(color => (
      <button key={color}
        variant={selColor === color ? 'dark' : 'light'}
        onClick={() => handleSelectOption('color', color)}
        style={{
          backgroundColor: color.toLowerCase(),
          color: color === 'White' ? 'black' : 'white'
        }}
      >{color}</button>
    ))}
  </div>
</div>

{selectedItems.map((opt, i) => (
  <div key={i} className="payment_select_wrap">
    <div className="select_text">
      <p>opt.title</p>
      <p>opt.size</p>
      <p>opt.price.toLocaleString()원</p>
    </div>
    <div className="del_btn">
      <button variant="secondary" size="sm" onClick={() => setSelectedItems(selectedItems.filter(_ , idx) => idx !== i)}>X</button>
    </div>
    <div className="select_count">
      <div className="count_btn">
        <button className='minus_btn' onClick={() => changeCount(i, -1)}>-</button>
        <input type="text" value={opt.count} readOnly />
        <button className='add_btn' onClick={() => changeCount(i, 1)}>+</button>
      </div>
      <div className="sel_h5">
        <h5>opt.price *</h5>
        <h5>opt.count</h5>
      </div>
    </div>
  </div>
));}

<div className="payment_total_wrap">
  <div className="payment_total">
    <span>총 상품 금액</span>
    <h3>total.toLocaleString()원</h3>
  </div>
</div>

<div className="payment_btn">
  <button variant="secondary" size="lg" className="cart_btn" onClick={selItemCart}>장바구니</button>
  <MyVerticallyCenteredModal show={modalShow} onHide={() => setModalShow(false)} />
  <button variant="secondary" size="lg" onClick={() => setModalShow(true)}>구매하기</button>
</div>
```

■ 장바구니 버튼 클릭시 모달창

■ 사이즈, 컬러 선택시 선택한 옵션 보여주기 출력 동일한 옵션일 경우 수량 추가 다른 옵션일 경우 따로 추가



구현

CART 코드 구현

SUB DETAIL PRODUCT

장바구니

□ 전체 선택

	에어쿨링 지니 시그니처 레깅스 M / Gray / 2	59,000원 × 2개 118,000원	X
	에어쿨링 지니 시그니처 레깅스 XL / White / 1	59,000원 × 1개 59,000원	X

선택 상품 금액
118,000원

■ 장바구니
상세 보기 페이지에서 옵션 선택 후 장바구니 버튼을 클릭시
장바구니 테이블에 상품 추가

■ 체크박스
체크 박스를 체크한 상품마다 합계 금액에 반영되어 출력되며
전체 선택 체크박스를 통해 전체 체크과 전체 해제 가능

■ 배송비
기존 배송비는 3천원으로 표시되며 체크 박스를 체크하여 금액
이
7만원 이상이 될 경우 무료배송으로 변경

SUB DETAIL PRODUCT

```
let navigate = useNavigate();
const dispatch = useDispatch();

const cartItem = useSelector(state => state.cart);
const [checkedItems, setCheckedItems] = useState([]);

const selectedTotal = cartItem.reduce((sum, item, idx) => {
  return checkedItems.includes(idx) ? sum + item.price * item.count : sum;
}, 0);
```

```
<div className='cart_wrap'>
  <table>
    <thead>
      <tr>
        <td>
          <input type="checkbox" checked={checkedItems.length === cartItem.length && cartItem.length > 0} onChange={(e) => {
            if (e.target.checked) {
              setCheckedItems(cartItem.map((_, idx) => idx));
            } else { setCheckedItems([]); }
          }} />
        </td>
        <td colSpan={2}>전체 선택</td>
      </tr>
    </thead>
```

**■ 전체선택 , 상품 개별 선택
체크박스**

```
{cartItem.map((item, index) => (
  <tr key={index}>
    <td>
      <input type="checkbox" checked={checkedItems.includes(index)} onChange={(e) => {
        if (e.target.checked) {
          setCheckedItems(prev => [...prev, index]);
        } else {
          setCheckedItems(prev => prev.filter(i => i !== index));
        }
      }} />
    </td>
    <td>
      <div className="cart_img">
        <img src={process.env.PUBLIC_URL+item imgUrl} alt="cartimg" />
      </div>
    </td>
    <td>
      <div className="cart_text">
        <h5>{item.title}</h5>
        <p>{item.size} / {item.color} / {item.count}</p>
      </div>
    </td>
    <td>
      <h6>{item.price.toLocaleString()}원 X {item.count}개</h6>
      <h5>({item.price * item.count}).toLocaleString()원</h5>
    </td>
  </tr>
))}
```

■ 장바구니에 담은 상품

```
<div className="payment">
  <table>
    <thead>
      <tr>
        <td>상품 금액</td>
        <td>{selectedTotal.toLocaleString()}원</td>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>배송비</td>
        <td>
          {selectedTotal >= 70000 ? '무료배송' : '3,000원'}
        </td>
      </tr>
    </tbody>
    <tfoot>
      <tr className='total'>
        <td>전체 합계</td>
        <td>
          {selectedTotal >= 70000
            ? `${selectedTotal.toLocaleString()}원`
            : `${(selectedTotal + 3000).toLocaleString()}원`}
        </td>
      </tr>
    </tfoot>
  </table>
</div>
```

**■ 기본 배송비는 3천원으로 표시
7만원이상시 무료배송으로 변경**

**■ 체크한 상품만 합계 + 배송비
7만원 이상이면 배송비 없이
합계 계산 출력**

05

마무리

R
e
f
l
e
c
t
i
o
n

리액트를 처음 배울때는 생소하기도하면서 어렵게 느껴지다보니 컴포넌트로 분리해서 쓰는 것이 편리하다'라는 것이 와닿지 않았지만, 제가 원하는 배치 및 기능의 코드를 잘 구성하고 잘 정리해두면 다른 작업을 할 때에도 재활용하여 사용하기 편리하며, 가독성이 좋아 유지보수하기에도 편리하다고 느꼈습니다.
자바스크립트 언어 기반인만큼 자바스크립트 언어에 대한 이해도가 부족하여 어려움이 많았지만 자주 사용하는 기능들을 이해하고 코드를 잘 짜놓을 수 있다면 다른 작업들을 할 때에도 효율성을 올릴 수 있기에 동기부여가 되는 작업이였습니다.

리액트 부트스트랩과 스와이퍼는 처음 사용하는 것은 아니지만 제가 원하는대로 CSS를 적용할 때 생각한 것처럼 잘 되지 않아 원래 하던 방식으로 해볼까도 고민했었지만, 작업의 효율성 측면에서는 부트스트랩과 스와이퍼를 사용하였을 때 단점보다 장점이 더 많다고 생각되어 적극적으로 활용하였으며, 일부 기능들을 이용하여 제가 원하는 방향으로의 CSS를 적용할 수 있도록 공부를 더 해야겠다는 걸 느꼈습니다.

리덕스를 처음 사용하고 공부할 때보다 더 많은 데이터를 만들다보니 시간이 꽤 걸리긴 했지만, 프롭스보다 자유롭게 데이터 활용이 가능하여 편리하였으며, 마무리가 되어갈쯤 더 넣고 싶은 기능들이 생겼지만 데이터를 추가해야 하는 작업이였기에 데이터를 추가하는데 시간을 더 투자하기엔 시간이 부족해 기능 넣지 못하고 마무리하게 되어 아쉽기도 했습니다.

새로운 작업들을 할 때면 느껴지는 부족함들이 자신감을 잃게도 하지만 제가 부족한 점이 무엇인지 알 수 있어 공부할 때 어떤 점을 더 많이 공부하고 채워 넣어야 할지 알게 되는 계기가 되기도 합니다.

다음 포트폴리오를 기획을 할 때에는 기획 단계에서 조금 더 디테일하게 계획을 짜고 구성하여 효율성을 높이고 아쉬움이 남지 않게 작업 할 수 있도록 좀 더 꼼꼼하게 기획하여 아쉬움보다 뿌듯함이 남는 작업을 할 수 있도록 해야겠다는 생각을 했습니다.

감사합니다

