

클론 코딩 연습과제

현대자동차 메인페이지

목차

1. 레이아웃 분석
2. DOM 구조 설정
3. 구현 결과

1. 레이아웃 분석

■ aside

- 주요 메뉴바

■ header

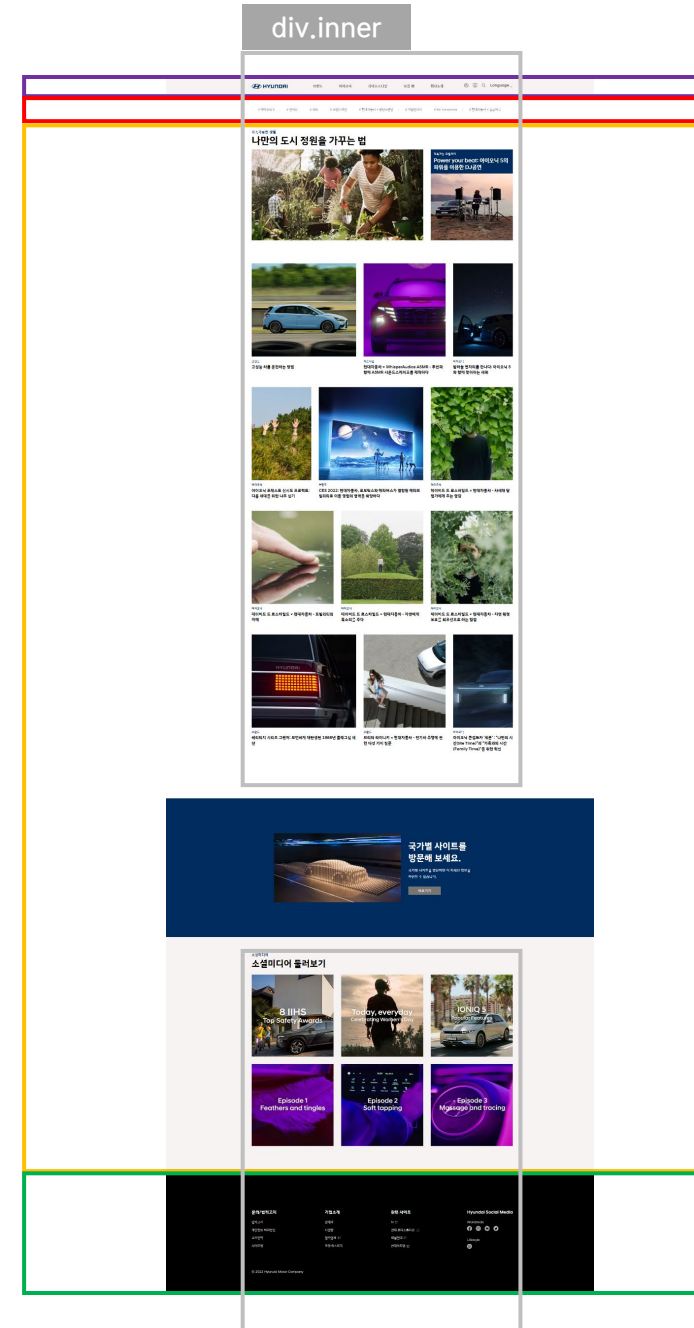
- 페이지 상단 영역

■ main

- 주요 콘텐츠

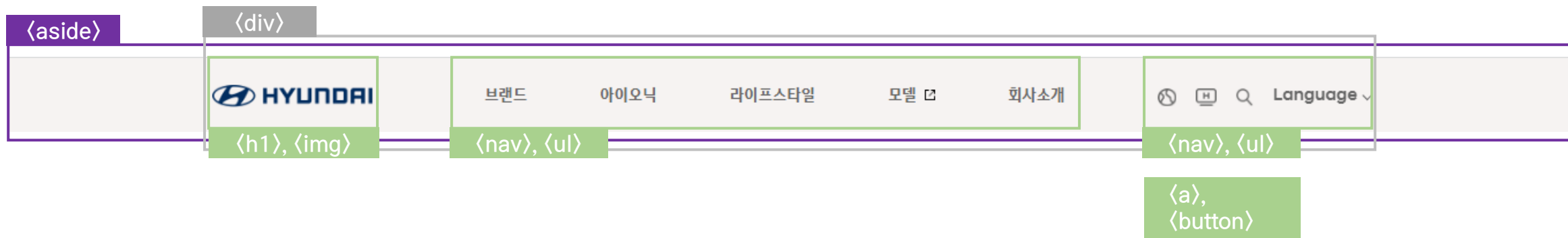
■ footer

- 기타 링크, 카피라이트



1. 레이아웃 분석

■ aside



시멘틱 구조

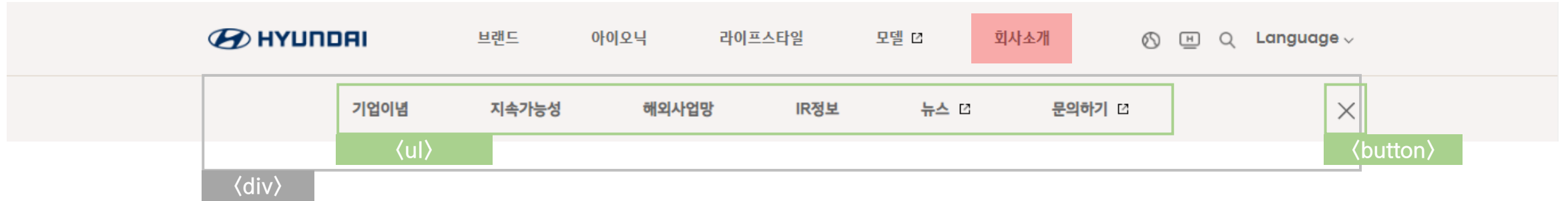
- 메인 메뉴는 `<nav>`, 검색 & 언어설정 등의 메뉴는 `<button>` 등 다양한 태그로 구현

웹접근성

- logo가 들어가는 부분에 보이지 않는 h1태그 활용 가능

1. 레이아웃 분석

■ aside

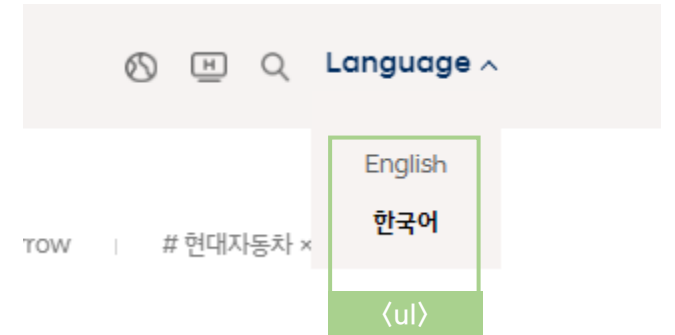
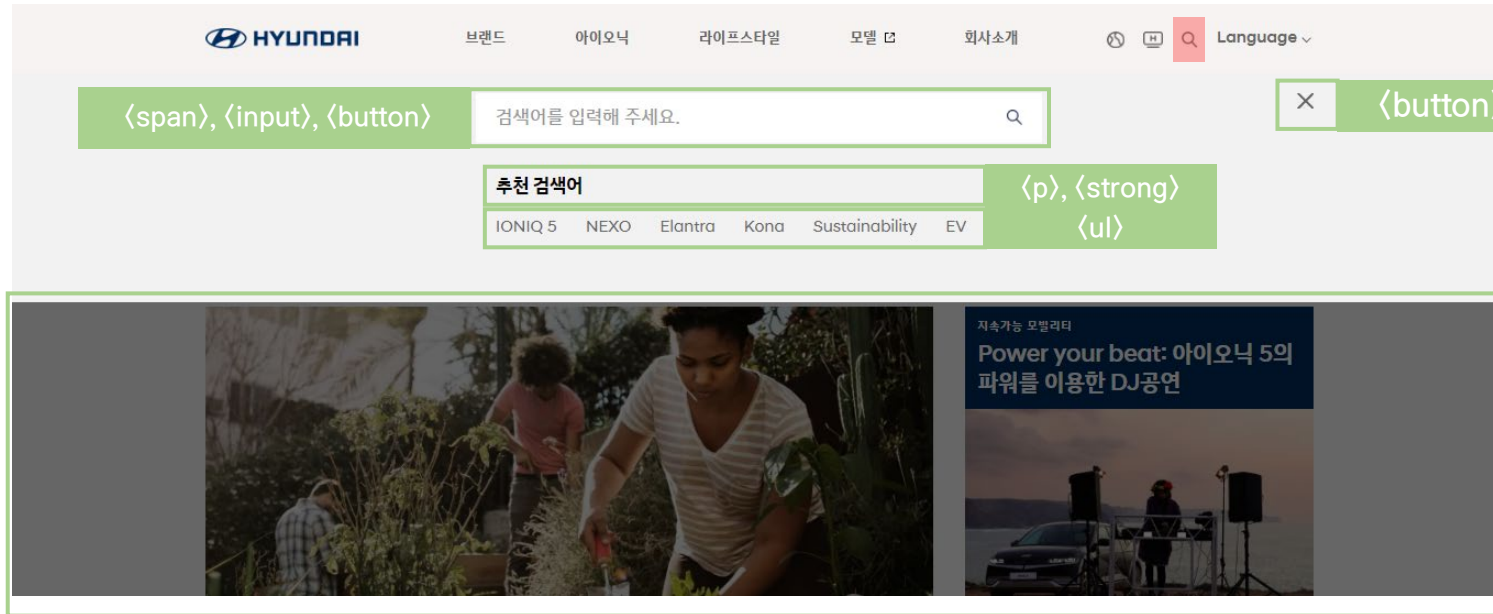


추가기능

- '회사 소개' 메뉴에 hover 시, 서브 메뉴 시각화
- '브랜드', '아이오닉' 등 다른 메뉴 hover 시 / 검색창 활성화 시 / language 클릭 시 서브 메뉴 영역 삭제
- 그 외의 영역의 경우 커서가 위치해도 서브메뉴가 사라지지 않으며, X 표시를 누르면 영역 삭제

1. 레이아웃 분석

■ aside



추가기능

- 메인 메뉴 우측 상단의 돋보기 아이콘 클릭 시 검색창 시각화
- 검색 영역은 `<input type="search">`, `<button>`으로 구현한 후 전체 ``으로 감싸 구현 가능
- 메인 메뉴 우측 상단의 language 버튼 클릭 시 언어 선택 목록이 시각화되며, 활성화된 언어에 따라 굵게 표시됨

1. 레이아웃 분석

■ header

아이오닉 5 | # 인터뷰 | # 넥쏘 | # 브랜드비전 | # 현대자동차 × 방탄소년단 | # 아임인차지 | # for Tomorrow | # 현대자동차 × 블룸버그

, <a>

주요 기능

- 메뉴 선택 시 해당 해시태그 관련 내용으로 본문 영역 콘텐츠 변화
- 메뉴를 마우스로 drag / grab 가능 (이번 과제에서는 미구현)

1. 레이아웃 분석

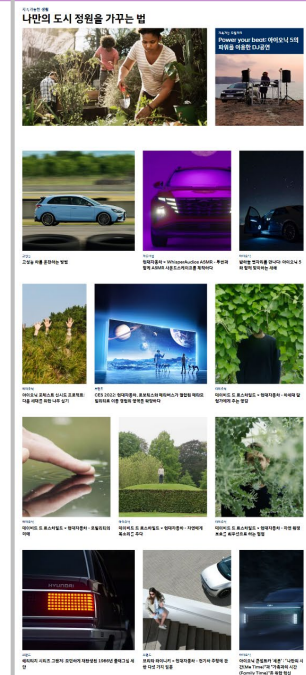
■ main

section 분리

- 크게 3개의 section으로 분리 → 내부에 article 설정
- 전체적으로 일정한 레이아웃을 갖추기 위해 div.inner 활용

〈section〉

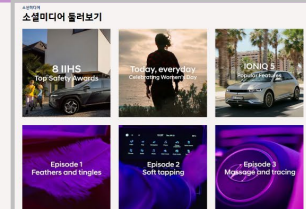
div.inner



〈section〉



〈section〉

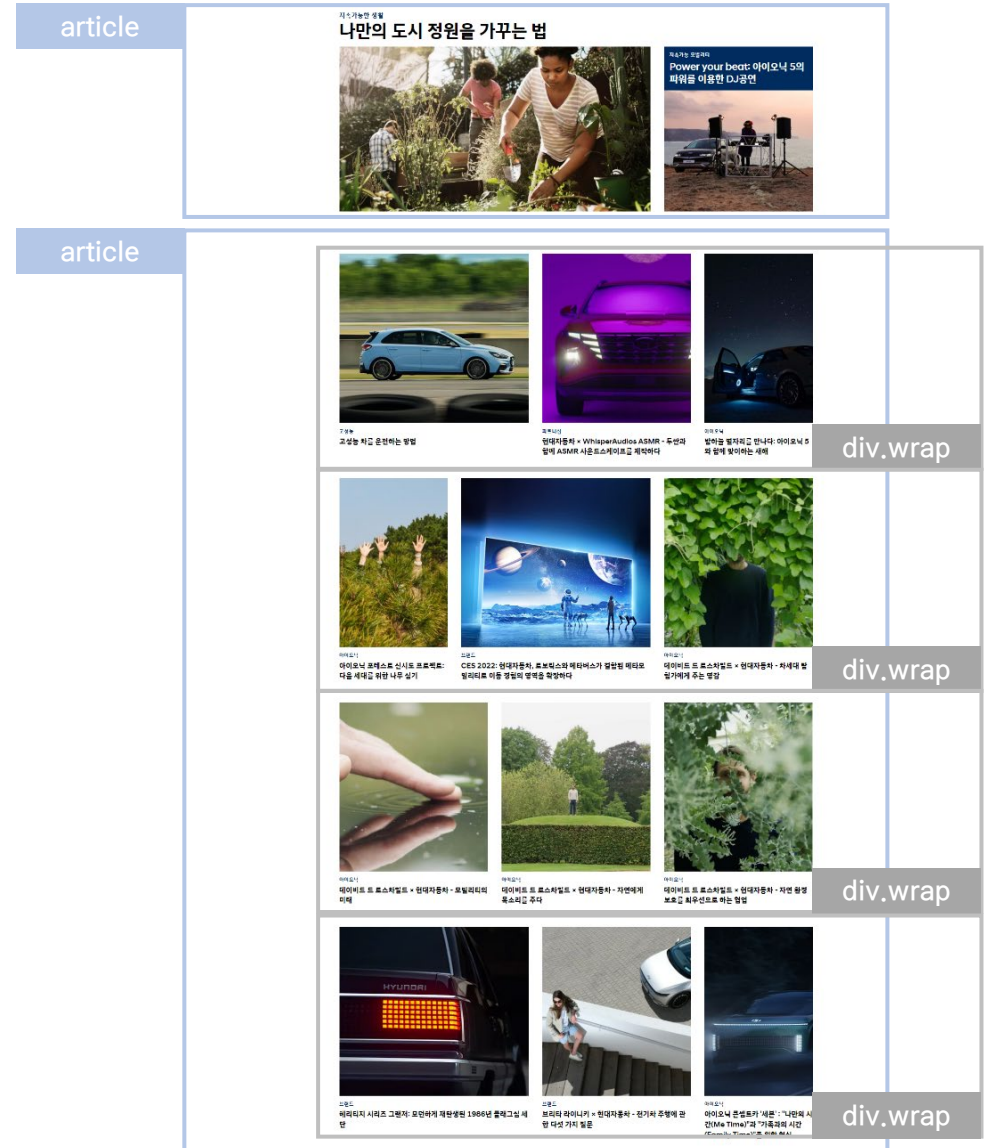


1. 레이아웃 분석

■ main

section1

- 주요 뉴스를 나타내는 article1, 콘텐츠를 나타내는 article2 세분
- article2는 다시 콘텐츠를 3개씩 담아 표시하도록 영역을 구분
 - article1의 제목 → h2
 - article2의 콘텐츠 카테고리명 → p, 타이틀 → h3



1. 레이아웃 분석

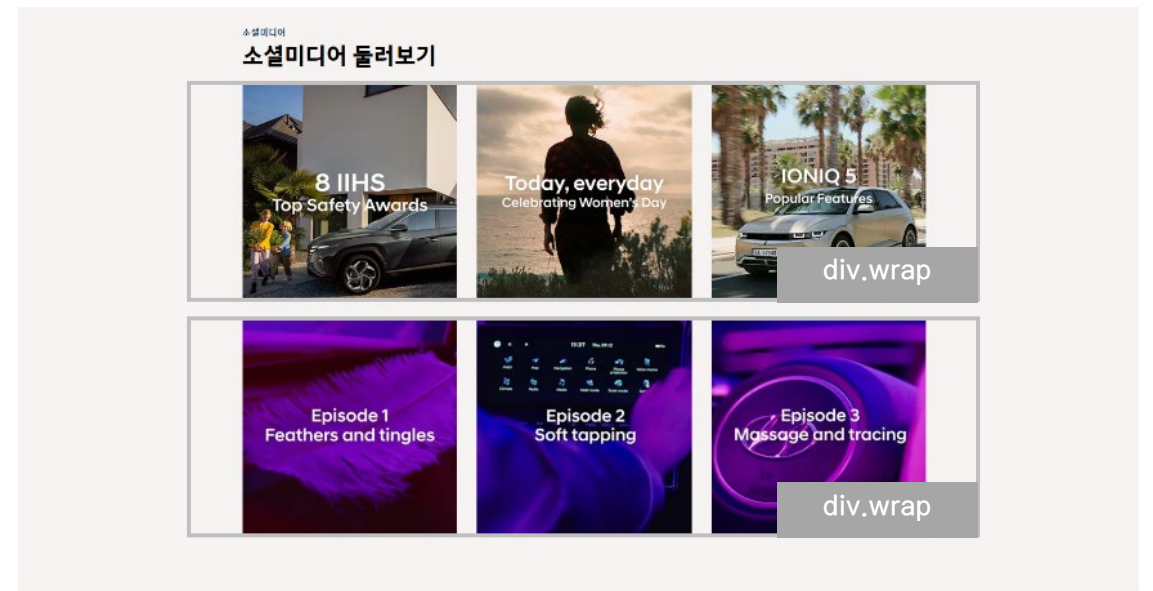
■ main

section2

- 이미지와 병렬표기

section3

- section2와 같이 콘텐츠를 3개씩 표기



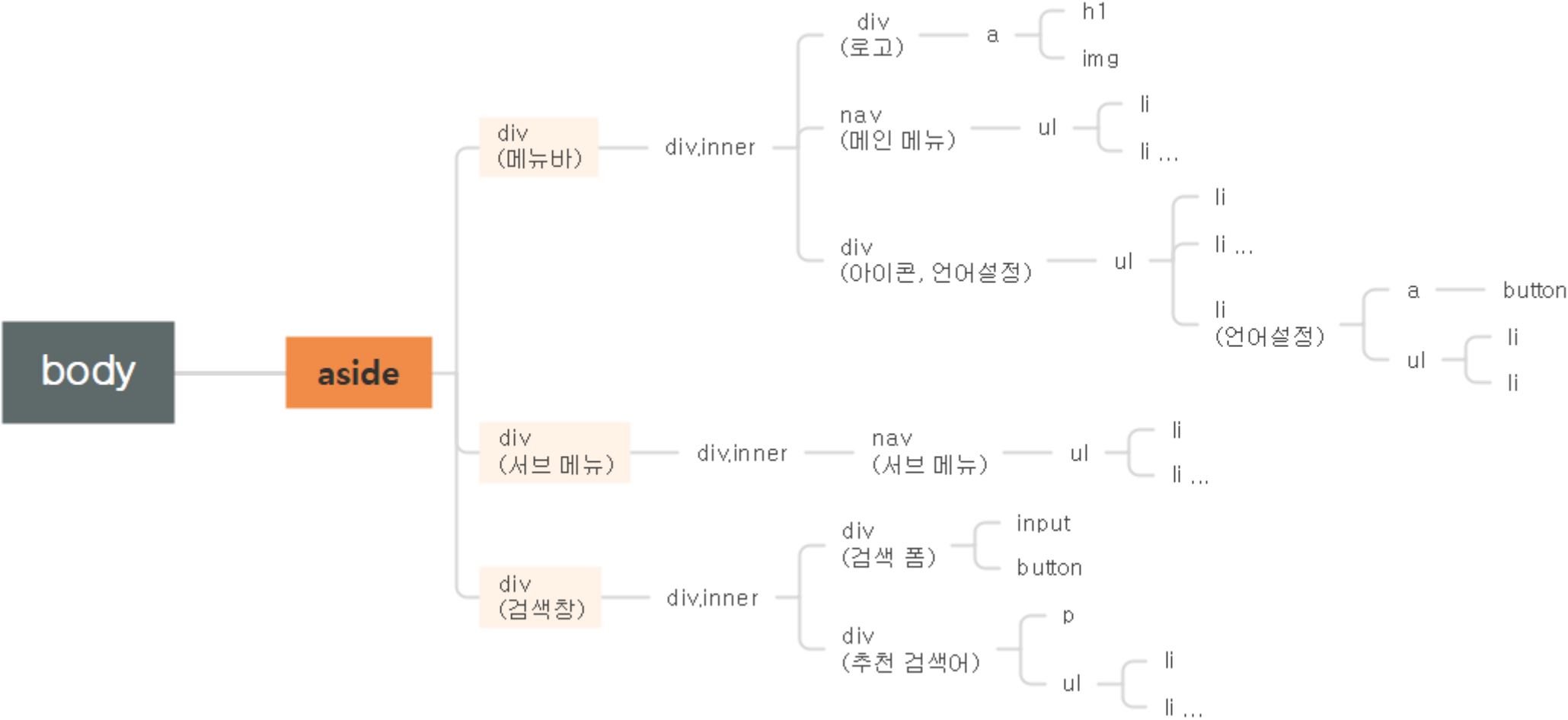
1. 레이아웃 분석

■ footer



- 각 영역을 ul로 구분하여 병렬표기
- 마우스 hover 시 글자의 명도가 변하는 효과가 있음

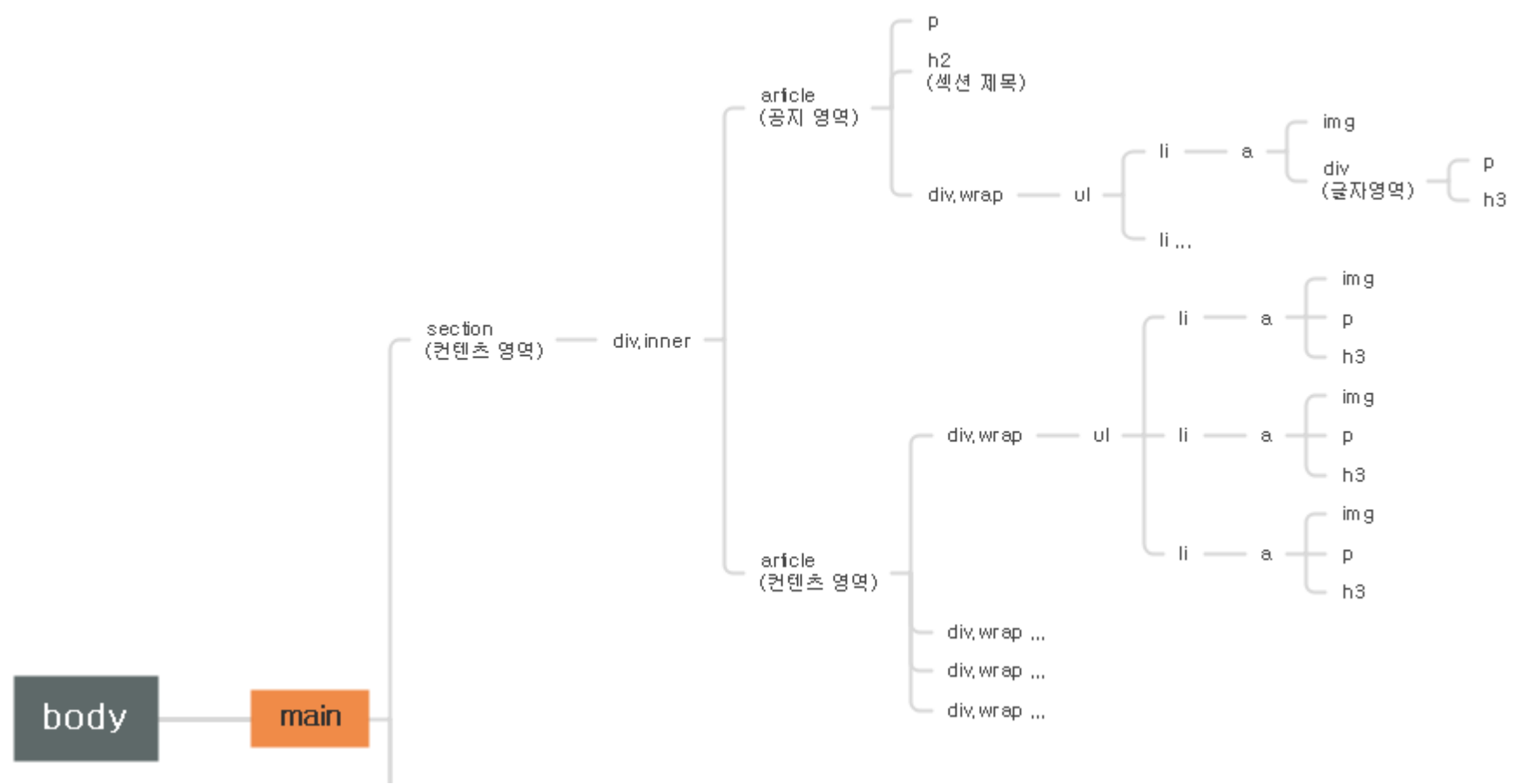
2. DOM 구조 설정



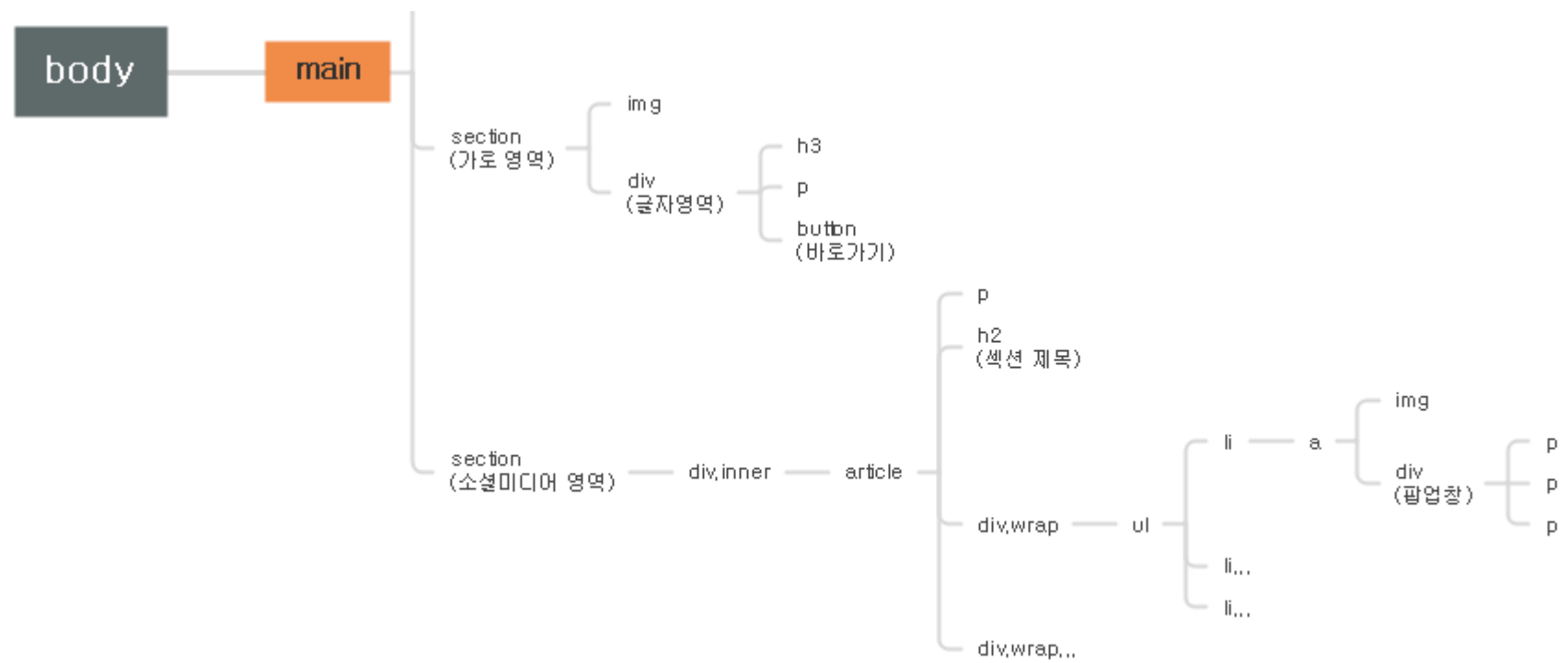
2. DOM 구조 설정



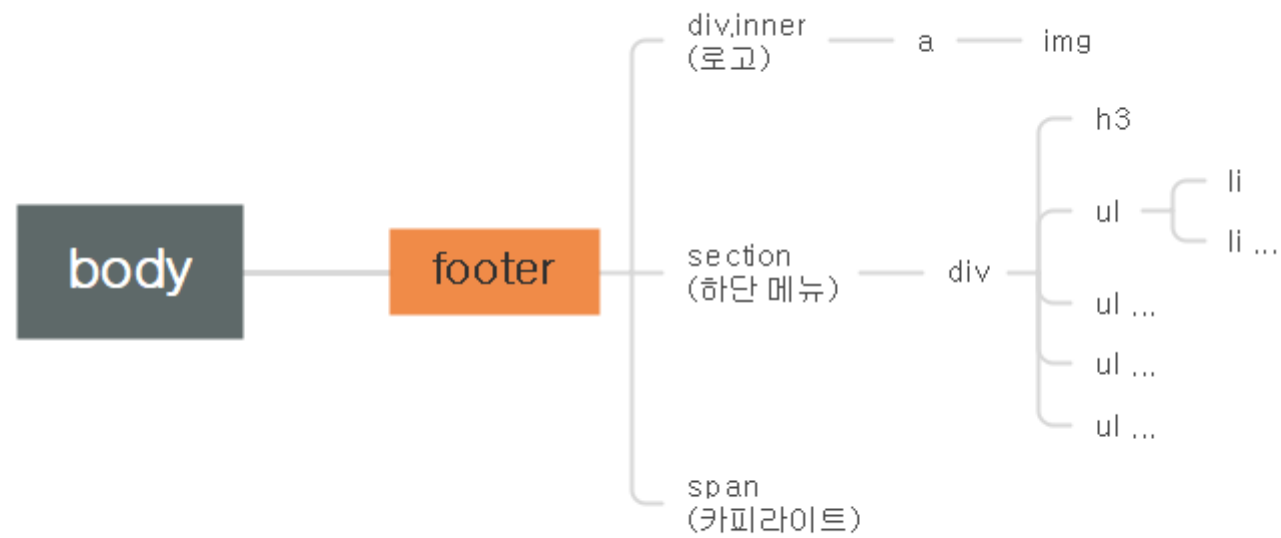
2. DOM 구조 설정

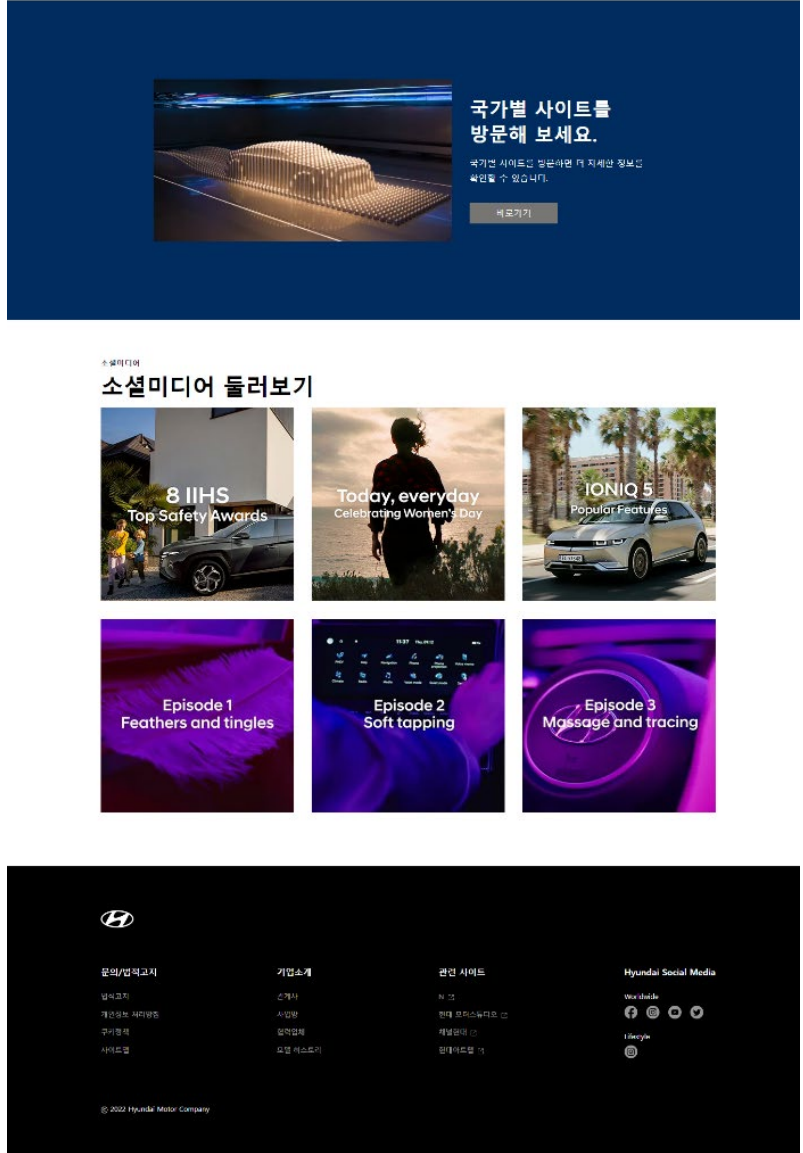
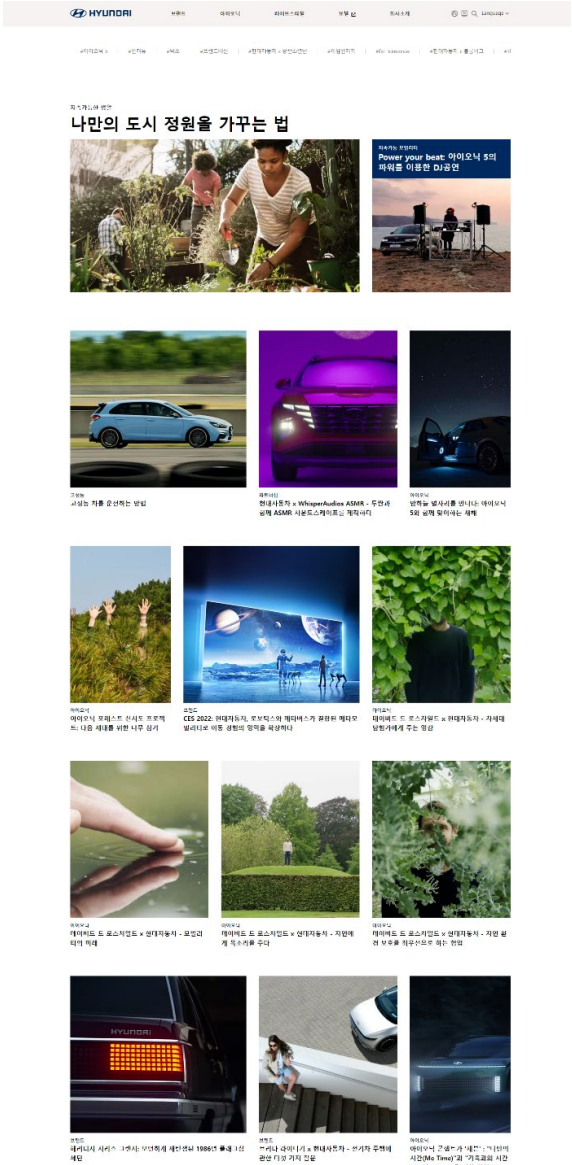
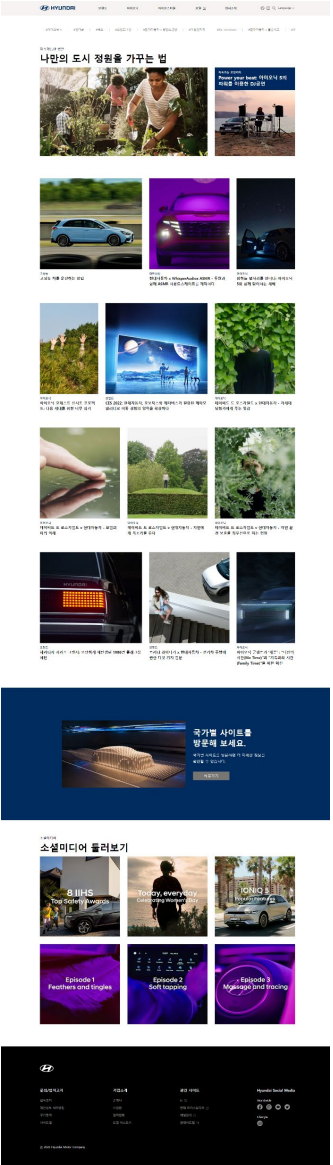


2. DOM 구조 설정



2. DOM 구조 설정





3. 구현 결과

■ 전체 영역

리더기로 읽히지만 보이지 않는 글자 (웹접근성)

```
/* *****
Main Color
***** */
$color1: #002c5f;
$color2: #f6f3f2;

/* *****
Default display
***** */
* {
  box-sizing: border-box;
}

html,
body {
  min-width: 100%;
  max-width: 100%;
  position: relative;
  word-break: keep-all;
}

a {
  all: unset;
  cursor: pointer;
}

button {
  all: unset;
  display: block;
  cursor: pointer;
}

li > a {
  display: block;
}

.blind-text {
  position: absolute;
  width: 1px;
  height: 1px;
  margin: -1px;
  overflow: hidden;
  clip: rect(0, 0, 0, 0);
  clip-path: inset(50%);
}

.flex-row {
  display: flex;
}

.float-left {
  float: left;
}

.float-right {
  float: right;
}

.clearfix::after {
  content: "";
  display: block;
  float: none;
  clear: both;
}

// 커스텀
.hide {
  display: none;
}


a[target="_blank"] {
  position: relative;
  &::after {
    content: '';
    display: inline-block;
    width: 10px;
    height: 10px;
    margin-left: .5rem;
    background-image: url(../img/ico_gnb_newwin2.png);
    vertical-align: middle;
  }
}

div.inner {
  width: 1120px;
  margin: 0 auto;
}

div.wrap ul {
  display: flex;
  justify-content: space-between;
}

div.dimmed {
  position: absolute;
  width: 100%;
  height: 100%;
  background-color: #00000090;
  z-index: 10;
}
```

새창열기로 설정된 모든 a 태그
뒤에 가상요소 삽입

모델 

검색창 활성화 시 화면을 가리는
검은 효과

3. 구현 결과

■ aside - include로 삽입

```
<!-- 상단 메뉴 -->
<div class="nav-bar">
  <div class="inner flex-row">
    <div class="logo">
      <a href="index_ko.html">
        <h1 class="blind-text">현대자동차</h1>
        
      </a>
    </div>
    <nav class="main-menu">
      <ul class="flex-row" id="main-menu">
        <li><a href="#">브랜드</a></li>
        <li><a href="#">아이오닉</a></li>
        <li><a href="#">라이프스타일</a></li>
        <li><a href="#" target="_blank">모델</a></li>
        <li><a href="#" id="comp-nav">회사소개</a></li>
      </ul>
    </nav>
    <div class="flex-row">
      <ul class="item-list flex-row">
        <li class="item"><a href="#" class="icon1"></a></li>
        <li class="item"><a href="#" class="icon2"></a></li>
        <li class="item"><button id="search-btn" class="icon3"></button></li>
        <li class="item">
          <a href="#" id="lang-btn">
            <button>Language</button>
          </a>
          <ul class="lang-list hide">
            <li class="eng"><a href="index_en.html">English</a></li>
            <li class="kor"><a href="index_ko.html">한국어</a></li>
          </ul>
        </li>
      </ul>
    </div>
  </div>
</div>
```

```
<!-- 회사소개 메뉴 -->
<div class="nav-sub">
  <div class="inner flex-row">
    <nav class="sub-menu" id="sub-menu">
      <ul class="flex-row">
        <li><a href="#">기업이념</a></li>
        <li><a href="#">지속가능성</a></li>
        <li><a href="#">해외사업망</a></li>
        <li><a href="#">IR정보</a></li>
        <li><a href="#" target="_blank">뉴스</a></li>
        <li><a href="#" target="_blank">문의하기</a></li>
      </ul>
    </nav>
    <button class="exit-btn" id="exit-menu"></button>
  </div>
</div>
<!-- 검색 메뉴 -->
<div class="nav-search">
  <div class="inner">
    <div class="search-ipt clearfix">
      <input type="search" class="float-left" placeholder="검색어를 입력해 주세요.">
      <button id="search-btn" class="float-left"></button>
    </div>
    <div class="search-keys">
      <p><strong>추천 검색어</strong></p>
      <ul class="search-list flex-row">
        <li><a href="#">IONIQ 5</a></li>
        <li><a href="#">NEXO</a></li>
        <li><a href="#">Elantra</a></li>
        <li><a href="#">Kona</a></li>
        <li><a href="#">Sustainability</a></li>
        <li><a href="#">EV</a></li>
      </ul>
    </div>
    <button class="exit-btn"></button>
  </div>
</div>
```

3. 구현 결과

■ aside

```
/*
*****
Aside
*****
*/
aside {
  background-color: #333;
  font-size: 0.8rem;
  color: #fff;
  .exit-btn {
    width: 1.1rem;
    height: 1.1rem;
    margin-left: 10rem;
    text-align: center;
    line-height: 1.1;
  }
  .exit-btn::after {
    content: "\f00d";
    font-family: "FontAwesome";
    font-size: 1.2rem;
    color: #555;
  }
  .exit-btn:hover {
    transform: rotate(180deg);
    transform-origin: center;
    transition: 0.4s;
  }
}

aside .nav-bar {
  .inner {
    justify-content: space-between;
    align-items: center;
  }
  .logo img {
    width: 155px;
  }
  .main-menu ul {
    width: 40rem;
    justify-content: space-between;
  }
  .main-menu ul a {
    line-height: 5;
    padding: 0 1rem;
    font-weight: bold;
  }
  .item-list {
    & > * {
      margin: 0 0.3rem;
    }
    & > li > a {
      padding: 0.5rem 0;
      display: block;
      width: 1rem;
      height: 1rem;
    }
    & > li > button {
      padding: 0.5rem 0;
      display: block;
      width: 1rem;
      height: 1rem;
    }
  }
  .icon1 {
    background: url(../img/btn_topmenu_language.png) no-repeat center;
    background-size: 1rem 1rem;
  }
  .icon2 {
    background: url(../img/btn_topmenu_Channelh.png) no-repeat center;
    background-size: 1rem 1rem;
  }
  .icon3 {
    background: url(../img/btn_topmenu_search.png) no-repeat center;
    background-size: 1rem 1rem;
  }
}

li:last-of-type a {
  width: auto;
  height: auto;
  position: relative;
}
#lang-btn button::after {
  display: inline-block;
  margin-left: 0.3rem;
  content: "\f107";
  font-family: "FontAwesome";
  font-size: 0.6rem;
  transition: .3s;
}
#lang-btn.on button::after {
  margin-left: 0.3rem;
  content: "\f107";
  font-family: "FontAwesome";
  font-size: 0.6rem;
  transform: rotate(180deg);
}
.lang-list {
  position: absolute;
  width: 5rem;
  padding: 1rem 0;
  text-align: center;
  background-color: #333;
  z-index: 100;
}
li {
  margin-top: 1rem;
  a {
    line-height: 2;
  }
}
li.active {
  font-weight: bold;
}
```

3. 구현 결과

■ aside

```
aside .nav-sub {
  position: absolute;
  width: 100%;
  max-height: 0;
  overflow: hidden; // 체크할 것 : 이거 없으
  background-color: #color2;
  transition: max-height 0.1s ease-out;

  .inner {
    align-items: center;
    justify-content: space-between;
  }
  .sub-menu {
    margin-left: 9rem;
  }
  .sub-menu ul {
    width: 50rem;
    justify-content: space-around;
  }
  .sub-menu ul a {
    line-height: 5;
    padding: 0 1rem;
    font-weight: bold;
  }
}

aside .nav-search {
  position: absolute;
  width: 100%;
  max-height: 0;
  overflow: hidden;
  background-color: #color2;
  transition: max-height 0.1s ease-out;
  z-index: 100;

  .inner {
    position: relative;
    padding: 0 20rem;
    height: 10rem;
  }
  .inner ul {
    justify-content: center;
  }
}

.search-ipt {
  width: 100%;
  margin: 1.5rem 0;
  input {
    display: block;
    width: 90%;
    height: 2.5rem;
    padding: 0 1rem;
    border: none;
    outline: none;
  }
  #search-btn {
    width: 10%;
    height: 2.5rem;
    background: #fff url(../img/btn_search.png) no-repeat center;
  }
}

.search-keys {
  padding: 0 1rem;
  strong {
    font-weight: bold;
    font-size: 1rem;
  }
}

.search-list {
  justify-content: space-between;
  margin: 1rem 0;
}

.exit-btn {
  position: absolute;
  top: 0;
  right: 0;
}
```

3. 구현 결과

■ aside

```
/*
 * type      : function
 * description : 시각화 메뉴 원상복귀 (영역 삭제)
 */
const subMenuRemover = () => {
  document.querySelector('.nav-sub').style.maxHeight = 0;
  document.querySelector('.nav-sub').style.borderTop = 'none'
  document.querySelector('.nav-sub').style.borderBottom = 'none'
}

const langRemover = () => {
  document.querySelector('#lang-btn')
    .parentElement.querySelector('.lang-list')
    .classList.add('hide');
}

const searchRemover = () => {
  document.querySelector('.nav-search').style.maxHeight = 0;
  document.querySelector('.dimmed').classList.add('hide');
}
```

- 메인 메뉴 중 마지막 li에 hover 시에 서브메뉴가 나타나며 다른 메뉴에 hover시 삭제
- X 버튼 눌렀을 때 메뉴 삭제
- 만약 검색창이나 언어설정 창이 열려 있을 때 [회사소개] 메뉴 hover 시 검색창 및 언어설정 창 영역 삭제

```
/*
 * type      : Event Handler
 * description : 서브 메뉴 시각화
 * param      : 이벤트 객체, 메뉴 인덱스, 메뉴 총 길이
 * p-description : 메뉴 인덱스를 받아 마지막 메뉴 외에 다른 메뉴에 hover 시 영역 삭제
 */
const subMenuHandler = (e, index, length) => {
  const box = e.currentTarget;
  if (index === (length - 1)) {
    document.querySelector('.nav-sub').style.maxHeight = box.scrollHeight + 'px';
    document.querySelector('.nav-sub').style.borderTop = '1px solid #ccc'
    document.querySelector('.nav-sub').style.borderBottom = '1px solid #ccc'
  } else subMenuRemover()
}

const mainMenuList = document.querySelector('#main-menu').children;
Array.from(mainMenuList).forEach((v, i) => {
  const length = Array.from(mainMenuList).length;
  v.addEventListener('mousemove', e => {
    subMenuHandler(e, i, length);
    // 기타 영역 삭제
    langRemover();
    searchRemover();
  });
});

document.querySelector('#exit-menu').addEventListener('click', e => {
  subMenuRemover();
})
```

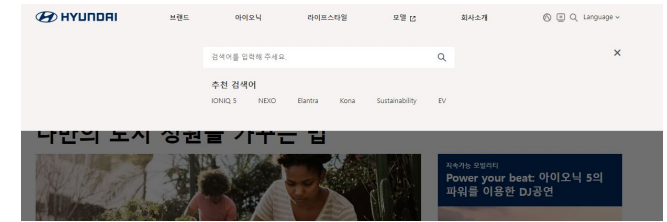

3. 구현 결과

■ aside

```
/*
 * type      : Event Handler
 * description : 검색창 시각화
 */
const searchHandler = () => {
  const target = document.querySelector('.nav-search')
  if(!target.style.maxHeight || target.style.maxHeight === '0px') {
    target.style.maxHeight = document.querySelector('.nav-search').scrollHeight + 'px';
  } else {
    searchRemover();
  }
}

document.querySelector('#search-btn').addEventListener('click', () => {
  document.querySelector('.dimmed').classList.toggle('hide');
  searchHandler();
  langRemover();
  subMenuRemover();
})
```

- 돋보기 아이콘 클릭 시 검색창 영역 시각화
- 돋보기 아이콘 클릭 시 서브메뉴, 언어설정 창이 활성화되어 있다면 해당 영역 삭제



3. 구현 결과

■ aside



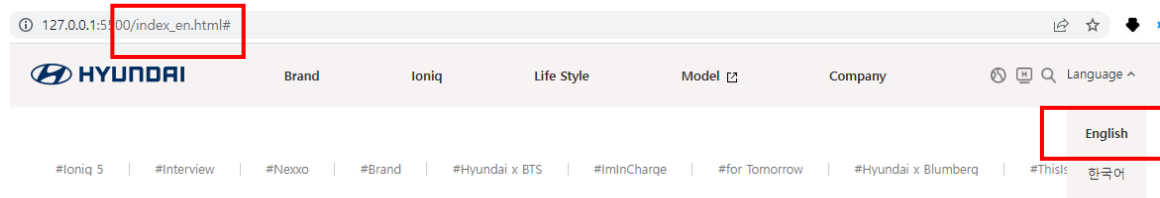
지속가능한 생활

나만의 도시 정원을 가꾸는 법



지속가능 모빌리티

Power your beat: 아이오닉 5의
파워를 이용한 DJ공연



Site Page Sample

Site Page Sample



Site Page Sample

Power your bear: Site Page

```
/*
 * type      : Event Handler
 * description : 언어 선택 영역 시각화
 */
document.querySelector('#lang-btn').addEventListener('click', e => {
  e.currentTarget.classList.toggle('on');
  e.currentTarget.parentElement.querySelector('.lang-list').classList.toggle('hide');
  // 기타 영역 삭제
  subMenuRemover();
  searchRemover();
});

/*
 * type      : Event Handler
 * description : URL에서 언어 키워드를 읽어와 현재 언어설정에 맞게 글자 강조 표시
 */
const urlString = location.pathname;
if(urlString.includes('ko')) {
  document.querySelector('.kor').classList.add('active');
  document.querySelector('.eng').classList.remove('active');
} else if (urlString.includes('en')) {
  document.querySelector('.kor').classList.remove('active');
  document.querySelector('.eng').classList.add('active');
} else {
  console.error('데이터 오류')
}
```



- language 버튼 클릭 시 선택창 활성화, 검색창 및 서브메뉴 영역 삭제
- 웹 주소에서 ko, en 등 키워드를 확인해 현재 설정된 언어 메뉴에 bold 처리



■ header

```
<header>
  <div class="inner">
    <ul id="hash" class="flex-row">
      <li><a href="assets/html/ioniq5.html">#아이오닉 5</a></li>
      <li><a href="">#인터뷰</a></li>
      <li><a href="">#넥쏘</a></li>
      <li><a href="">#브랜드비전</a></li>
      <li><a href="">#현대자동차 x 방탄소년단</a></li>
      <li><a href="">#아임인차지</a></li>
      <li><a href="">#for Tomorrow</a></li>
      <li><a href="">#현대자동차 x 블룸버그</a></li>
      <li><a href="">#ThisIsUs</a></li>
      <li><a href="">#친환경</a></li>
    </ul>
  </div>
</header>
```

```

/*****
Header
*****/

header {
  padding: 3.5rem 0 0;
  font-size: 0.79rem;
  color:   #777;

  #hash {
    white-space: nowrap;
    overflow: hidden;
    li {
      border-right: 1px solid   #ccc;
    }
    a {
      line-height: 1;
      margin: 0 1.6rem;
    }
  }
}
}

```

- 구현 확인을 위해 [아이오닉 5] 메뉴에만 링크 적용하였음

3. 구현 결과

■ main

```
<main>
  <!-- sec 1 -->
  <section class="section1">
    <div class="inner">
      <article class="main-ntc">
        <p class="sec-cate">지속가능한 생활</p>
        <h2 class="sec-title">나만의 도시 정원을 가꾸는 법</h2>
      </article>

      <article class="main-art">
        <!-- 여기에 ajax 추가 -->
      </article>
    </div>
  </section>

  <!-- sec 2 -->
  <section class="section2 flex-row">
    
    <div>
      <h3>국가별 사이트를 방문해 보세요.</h3>
      <p>국가별 사이트를 방문하면 더 자세한 정보를 확인할 수 있습니다.</p>
      <button>바로가기</button>
    </div>
  </section>

  <!-- sec3 -->
  <section class="section3">
    <div class="inner">
      <article class="main-scl">
        <p class="sec-cate">소셜미디어</p>
        <h2 class="sec-title">소셜미디어 둘러보기</h2>
        <!-- 여기에 ajax 추가 -->
      </article>
    </div>
  </section>
</main>
```

3. 구현 결과

■ main

```

/*****
Main
*****/
// sec 1
section {
  margin: 4.2rem 0;
}
article {
  margin-bottom: 6rem;
  h2, h3, p {
    font-weight: bold;
    line-height: 1.5;
  }
  p.sec-cate {
    font-size: 0.9rem;
    color: #000080;
  }
  h2.sec-title {
    font-size: 2.6rem;
    margin-bottom: .5rem;
  }
}

.section1 {
  .main-art {
    word-break: break-all;
    img {
      margin-bottom: 0.5rem;
    }
    p {
      font-size: 0.75rem;
      color: #000080;
    }
  }
  div.wrap {
    margin-bottom: 4.5rem;
    li {
      position: relative;
    }
    &.w-type1 {
      li:first-of-type {width: 40%;}
      li:nth-of-type(2) {width: 31.4%;}
      li:last-of-type {width: 22.85%;}
    }
    &.w-type2 {
      li:first-of-type {width: 22.85%;}
      li:nth-of-type(2) {width: 40%;}
      li:last-of-type {width: 31.4%;}
    }
    &.w-type3 {
      li:first-of-type {width: 31.4%;}
      li:nth-of-type(2) {width: 31.4%;}
      li:last-of-type {width: 31.4%;}
    }
  }
  .wrap-item div {
    padding: 0.8rem 1rem;
    color: #000000;
    position: absolute;
    top: 0;
    left: 0;
    p {
      font-size: 0.75rem;
    }
    h3 {
      font-size: 1.3rem;
      line-height: 1.3;
    }
  }
}

.section2 {
  justify-content: center;
  align-items: center;
  width: 100%;
  padding: 9rem 13rem;
  background-color: #000080;
  color: #000000;
  div {
    width: 24rem;
    padding: 0 2rem;
  }
  h3 {
    font-size: 2.2rem;
    font-weight: bold;
    line-height: 1.3;
    margin-bottom: 1rem;
  }
}

p {
  line-height: 1.7;
  margin-bottom: 2rem;
}
button {
  width: 10rem;
  height: 2.4rem;
  background-color: #000000;
  text-align: center;
}

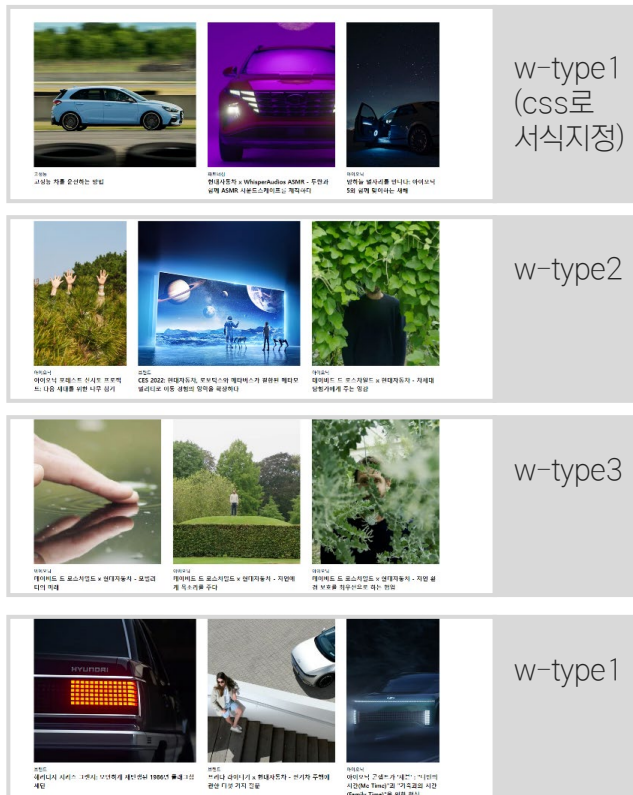
.section3 {
  div.wrap {
    margin-bottom: 2rem;
  }
  .popup-item {
    position: relative;
  }
  .popup-box {
    width: 18.5rem;
    height: 18.5rem;
    padding: 2rem;
    background-color: #000080;
    color: #000000;
    font-weight: normal;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    opacity: 0;
    transition: opacity .4s;
  }
  &.show {
    opacity: 1;
  }
  .insta-icon {
    display: block;
    width: 1.5rem;
    height: 1.5rem;
    margin-top: 7rem;
    background: url(../img/instagram.png) no-repeat center;
    background-size: contain;
  }
}

```

3. 구현 결과

■ main

- 메인 영역의 중앙 콘텐츠는 전부 ajax로 출력
- 3개씩 묶음으로 만들어질 수 있도록 작성
- getArtWrap 함수에서 wrapCount는 각 wrap을 구분하는 클래스를 부여하기 위해 작성됨



```
/*
 * @filename: contents.js
 * @author: 발미소
 * @description: section1 각 article 영역에 백엔드에서 읽어온 콘텐츠를 출력한다.
 */

/*
 * type      : function
 * description : HTML 요소 신규 생성 및 속성 부여
 * param     : HTML 요소명 / 속성명 / 속성값
 */
const createElement = (item, property = null, value = null) => {
  let a = document.createElement(item);
  if(property && value) {
    a.setAttribute(property, value);
  }
  return a;
}

/*
 * type      : function
 * description : article.main-ntc 영역에 콘텐츠 작성
 * param     : article.json > news
 */
const getNtcWrap = (data) => {
  let count = 0;
  const ul = createElement('ul');

  Array.from(data).forEach( v => {
    const li = createElement('li', 'class', 'wrap-item');
    const a = createElement('a', 'href', v.url);
    const img = createElement('img', 'src', `${v.img}.png`);
    const div2 = createElement('div');
    const p = createElement('p', 'class', 'art-item-cate');
    const h3 = createElement('h3', 'class', 'art-item-tit');

    p.innerHTML = v.category;
    h3.innerHTML = v.title;

    div2.appendChild(p);
    div2.appendChild(h3);
    a.appendChild(img);
    a.appendChild(div2);
    li.appendChild(a);
    ul.appendChild(li);

    count++;
  });
}
```

```
if (count === 2) {
  const div = createElement('div', 'class', 'wrap');
  div.appendChild(ul);
  document.querySelector('.main-ntc').appendChild(div);
}

/*
 * type      : function
 * description : article.main-art 영역에 콘텐츠 작성
 * param     : article.json > news
 */
const getArtWrap = (data) => {
  // count(li 개수 카운트), wrapCount(div 개수 카운트)
  let count = 0, wrapCount = 0;

  // ul 생성(li 3개 단위로 신규 생성)
  let ul = null;

  Array.from(data).forEach( v => {
    if (count === 0) {
      ul = createElement('ul');
    }

    const li = createElement('li');
    const a = createElement('a', 'href', v.url);
    const img = createElement('img', 'src', `${v.img}.png`);
    const p = createElement('p', 'class', 'art-item-cate');
    const h3 = createElement('h3', 'class', 'art-item-tit');

    p.innerHTML = v.category;
    h3.innerHTML = v.title;

    a.appendChild(img);
    a.appendChild(p);
    a.appendChild(h3);
    li.appendChild(a);
    ul.appendChild(li);

    count++;

    if (count > 2) {
      count = 0;
      wrapCount = ++wrapCount % 3 === 0 ? 3 : wrapCount % 3;
    }
  });
}
```

3. 구현 결과

■ main

```
if (count > 2) {
  count = 0;
  wrapCount = ++wrapCount % 3 === 0 ? 3 : wrapCount % 3;

  const div = createElement('div', 'class', 'wrap');
  div.classList.add(`w-type${wrapCount}`);
  div.appendChild(ul);
  document.querySelector('.main-art').appendChild(div);
}
});
}

/*
 * type      : function
 * description : article.main-scl 영역에 컨텐츠 작성
 * param      : article.json > social
 */
const getSclWrap = (data) => {
  // count(li 개수 카운트), wrapCount(div 개수 카운트)
  let count = 0, wrapCount = 0;

  // ul 생성(li 3개 단위로 신규 생성)
  let ul = null;

  Array.from(data).forEach(v => {
    if (count === 0) {
      ul = createElement('ul');
    }

    const li = createElement('li', 'class', 'popup-item');
    const a = createElement('a', 'href', v.url);
    const img = createElement('img', 'src', `${v.img}.png`);
    const div = createElement('div', 'class', 'popup-box');
    const p1 = createElement('p', 'class', 'pop-content');
    const p2 = createElement('p', 'class', 'hash');
    const span = createElement('p', 'class', 'insta-icon');

    p1.innerHTML = v.content;
    p2.innerHTML = v.hash;

    div.appendChild(p1);
    div.appendChild(p2);
    div.appendChild(span);
    a.appendChild(img);
    a.appendChild(div);
    li.appendChild(a);
    ul.appendChild(li);
  });
}
```

```
if (count > 2) {
  count = 0;
  wrapCount = ++wrapCount % 3 === 0 ? 3 : wrapCount % 3;

  const div2 = createElement('div', 'class', 'wrap');
  div2.appendChild(ul);
  document.querySelector('.main-scl').appendChild(div2);
}
});

/*
 * type      : function (async)
 * description : 백엔드에서 데이터를 전송받아 필요한 함수 호출
 * param      : -
 */
(async e => {
  let json = null;

  try {
    json = await axios.get('http://localhost:3000/news');
  } catch (e) {
    alert('데이터 전송 중 에러가 발생하였습니다.');
```

```
    return;
  }

  {let {data} = json; getArtWrap(data);}

  try {
    json = await axios.get('http://localhost:3000/main');
  } catch (e) {
    alert('데이터 전송 중 에러가 발생하였습니다.');
```

```
    return;
  }

  {let {data} = json; getNtcWrap(data);}

  try {
    json = await axios.get('http://localhost:3000/social');
  } catch (e) {
    alert('데이터 전송 중 에러가 발생하였습니다.');
```

```
    return;
  }

  {let {data} = json; getSclWrap(data);}
})();
```

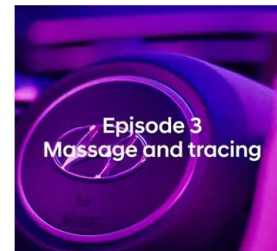
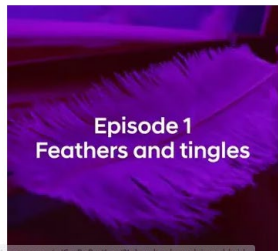
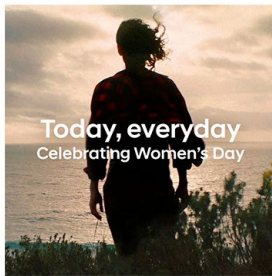
3. 구현 결과

■ main

- 소셜미디어 목록에 hover 시 탐색 팝업창이 나타나는 애니메이션 구현
- 해당 이벤트는 특정 클래스를 가진 요소들을 모두 불러와 event listener를 붙여 구현했는데, ajax 처리 속도가 느린 것인지 클래스 요소를 읽어올 시점에 콘텐츠 HTML이 미처 작성되지 않아 node list가 빈칸으로 반환되는 사고가 있었음
- 이에 억지로 코드 실행 시간을 딜레이하였으나 올바른 해결책은 아닌 것으로 보임

소셜미디어

소셜미디어 둘러보기



```
/**
 * @filename: animation.js
 * @author: 방미소
 * @description: 페이지에서 애니메이션 효과를 발생시킨다.
 */

/*
 * type      : Event Handler
 * description : index 페이지 section3 영역에 팝업 효과
 * event      : mouseover / mouseout
 */

console.log('hello')

const popUpHandler = e => {
  e.currentTarget.querySelector('.popup-box').style.opacity = '1';
}

const popRemoveHandler = e => {
  e.currentTarget.querySelector('.popup-box').style.opacity = '0';
}

let count = 0;
const timer = setInterval(() => {
  const popupList = document.querySelectorAll('.popup-item');
  Array.from(popupList).forEach(v => {
    v.addEventListener('mouseover', e => popUpHandler(e))
    v.addEventListener('mouseout', e => popRemoveHandler(e))
  })
  count++;

  if (count === 1) {clearInterval(timer)}
  console.log(popupList);
}, 500);
```

3. 구현 결과

■ footer – include로 삽입

```
<!-- 하단 메뉴 -->
<div class="inner">
  <div class="logo">
    <a href=""></a>
  </div>
  <section class="list-con flex-row">
    <div>
      <h3>문의/법적고지</h3>
      <ul>
        <li><a href="">법적고지</a></li>
        <li><a href="">개인정보 처리방침</a></li>
        <li><a href="">쿠키정책</a></li>
        <li><a href="">사이트맵</a></li>
      </ul>
    </div>
    <div>
      <h3>기업소개</h3>
      <ul>
        <li><a href="">관계사</a></li>
        <li><a href="">사업망</a></li>
        <li><a href="">협력업체</a></li>
        <li><a href="">모델 히스토리</a></li>
      </ul>
    </div>
  </section>
</div>
```

```
<div>
  <h3>관련 사이트</h3>
  <ul>
    <li><a href="" target="_blank">N</a></li>
    <li><a href="" target="_blank">현대 모터스튜디오</a></li>
    <li><a href="" target="_blank">채널현대</a></li>
    <li><a href="" target="_blank">현대아트랩</a></li>
  </ul>
</div>
<div class="social-con">
  <h3>Hyundai Social Media</h3>
  <p>Worldwide</p>
  <ul class="icon-wrap flex-row">
    <li><a href="" class="icon1"></a></li>
    <li><a href="" class="icon2"></a></li>
    <li><a href="" class="icon3"></a></li>
    <li><a href="" class="icon4"></a></li>
  </ul>
  <p>Lifestyle</p>
  <ul>
    <li><a href="" class="icon5"></a></li>
  </ul>
</div>
</section>
<span class="copyright">© 2022 Hyundai Motor Company</span>
</div>
```


3. 구현 결과

■ footer

```
/*
Footer
*/
footer {
  color: #fff;
  background-color: #000;

  .inner {
    padding: 5rem 0;
    font-size: 0.8rem;
    line-height: 1.3;
  }

  .list-con {
    justify-content: space-between;
    h3 {
      font-size: 1rem;
      font-weight: bold;
      margin-bottom: 1.6rem;
    }

    ul li {
      margin-bottom: 1rem;
    }

    ul a {
      opacity: .6;
      &:hover {
        opacity: 1;
      }
      &[target="_blank"]::after {
        background-position: 0 -10px;
      }
    }
  }
}

.social-con {
  p {
    margin-bottom: .5rem;
  }
  ul {
    margin-bottom: .5rem;
  }
  li {
    margin-right: 1rem;
  }
  a {
    width: 1.5rem;
    height: 1.5rem;
  }

  .icon1 {
    background: url(../img/facebook.png) no-repeat center;
    background-size: contain;
  }
  .icon2 {
    background: url(../img/instagram.png) no-repeat center;
    background-size: contain;
  }
  .icon3 {
    background: url(../img/youtube.png) no-repeat center;
    background-size: contain;
  }
  .icon4 {
    background: url(../img/twitter.png) no-repeat center;
    background-size: contain;
  }
  .icon5 {
    background: url(../img/instagram.png) no-repeat center;
    background-size: contain;
  }
}
```


소요시간

- 코딩 : 15시간 전후

후기

- CSS는 할 때마다 ‘이게 왜 안 되지’와 ‘이게 왜 되지’의 사이에서 갈팡질팡 하는 것 같아요. 가상요소나 애니메이션 사용이 아직 익숙하지 않아서 더 많이 연습하려고 합니다. 키프레임도 잘 써보고 싶습니다.
- HTML DOM 구조는 많이 익숙해진 것 같습니다. 적재적소에 필요한 태그를 잘 삽입할 수 있게 신경 쓰려고 해요.
- 이번 과제는 언뜻 보기엔 심플했지만 여기저기 시각적으로 신경 쓴 부분이 많은 사이트였던 것 같습니다. 특히 header 메뉴에 drag&grab 스크롤 효과가 있는 것이 인상 깊었는데 결국 라이브러리를 찾지 못해서 구현하지 못한 것이 조금 아쉽습니다.