



Le Catering



About Catering

Tradition since 1889

The Catering was founded in blabla by Mr. Smith in lorem ipsum dolor sit amet. consectetur adipiscing elit consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. We only use seasonal ingredients.

Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

클론코딩 - 01

Catering homepage

Our Menu

Bread Basket

Assortment of fresh baked fruit breads and muffins 5.50

Honey Almond Granola with Fruits

Natural cereal of honey toasted oats, raisins, almonds and dates 7.00

Belgian Waffle

Vanilla flavored batter with malted flour 7.50

Scrambled eggs

Scrambled eggs, roasted red pepper and garlic, with green onions 7.50

Blueberry Pancakes

With syrup, butter and lots of berries 8.50



서보션

Contact

We offer full-service catering for any event, large or small. We understand your needs and we will cater the food to satisfy the biggest criteria of them all, both look and taste. Do not hesitate to contact us.

Catering Service, 42nd Living St, 43043 New York, NY

You can also contact us by phone 00553123-2323 or email catering@catering.com, or you can send us a message here:

Name

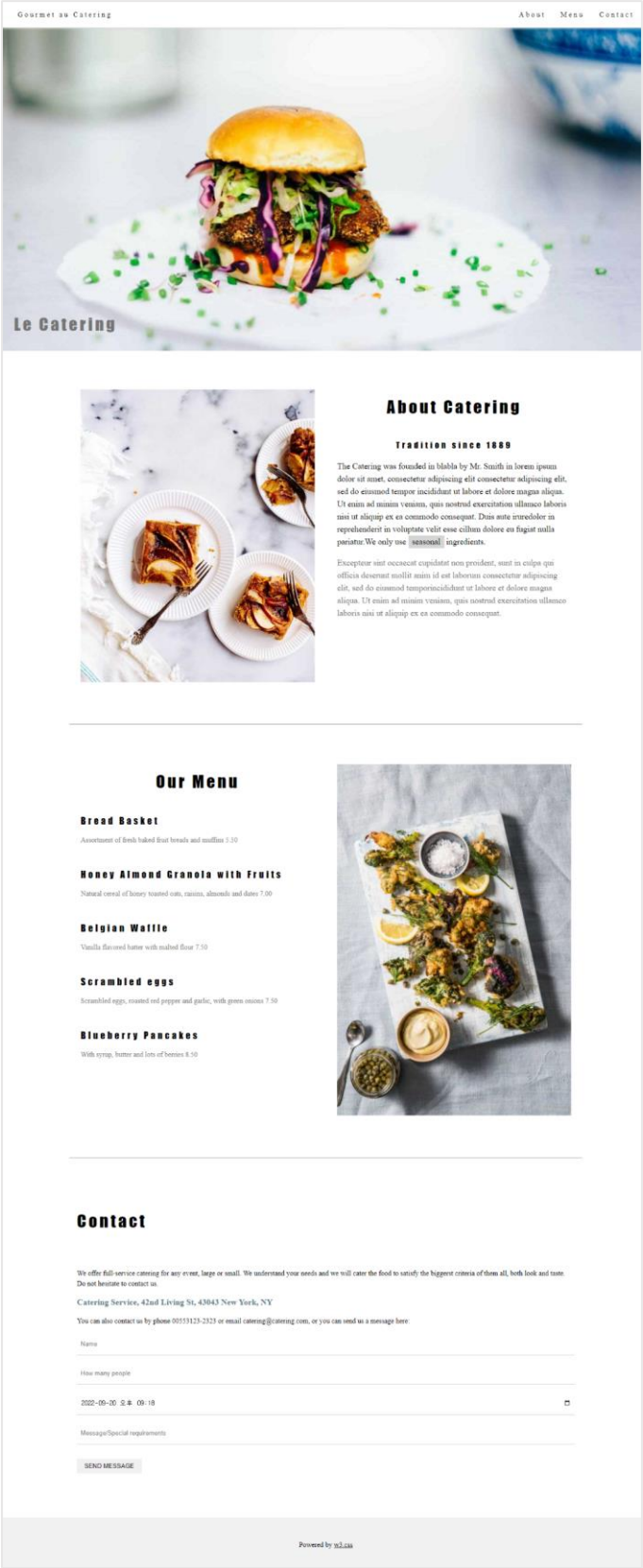
How many people

2022-09-20 12:00 09:18

Message/Special requirements

SEND MESSAGE

1. 최종 결과물



네비게이션 메뉴
배너

메인 콘텐츠 1

메인 콘텐츠 2

Contact Form

Footer

2. DOM Tree

HTML Console 창에

DOM Tree를 출력하기 위한 코드

```
1 <script type="module">
2   import exploreDom from './js/Dom.js';
3
4   const body = document.querySelector('body');
5   const i = exploreDom(body);
6   console.log(i);
7 </script>
```

```
1 let lineNumber = 0;
2
3 function exploreDom(ancestor) {
4   lineNumber++;
5   if (ancestor.children) {
6     console.group(`<${ancestor.tagName}>`);
7     Array.from(ancestor.children).forEach(v => {
8       exploreDom(v);
9     });
10    console.groupEnd();
11  } else {
12    console.log(ancestor.tagName);
13    return lineNumber;
14  }
15
16  return lineNumber;
17 }
18
19 export default exploreDom;
```

▼	<BODY>
▼	<NAV>
▼	<DIV>
▼	<A>
▼	<DIV>
▼	<A>
▼	<A>
▼	<A>
▼	<HEADER>
▼	
▼	<DIV>
▼	<H1>
▼	<ARTICLE>
▼	<SECTION>
▼	<DIV>
▼	
▼	<DIV>
▼	<H1>
▼	
▼	<H5>
▼	<P>
▼	
▼	<P>

▼ <HR>

▼ <SECTION>

▼ <DIV>

▼ <H1>

▼

▼ <H4>

▼ <P>

▼

▼ <H4>

▼ <P>

▼

▼ <H4>

▼ <P>

▼

▼ <H4>

▼ <P>

▼

▼ <H4>

▼ <P>

▼ <DIV>

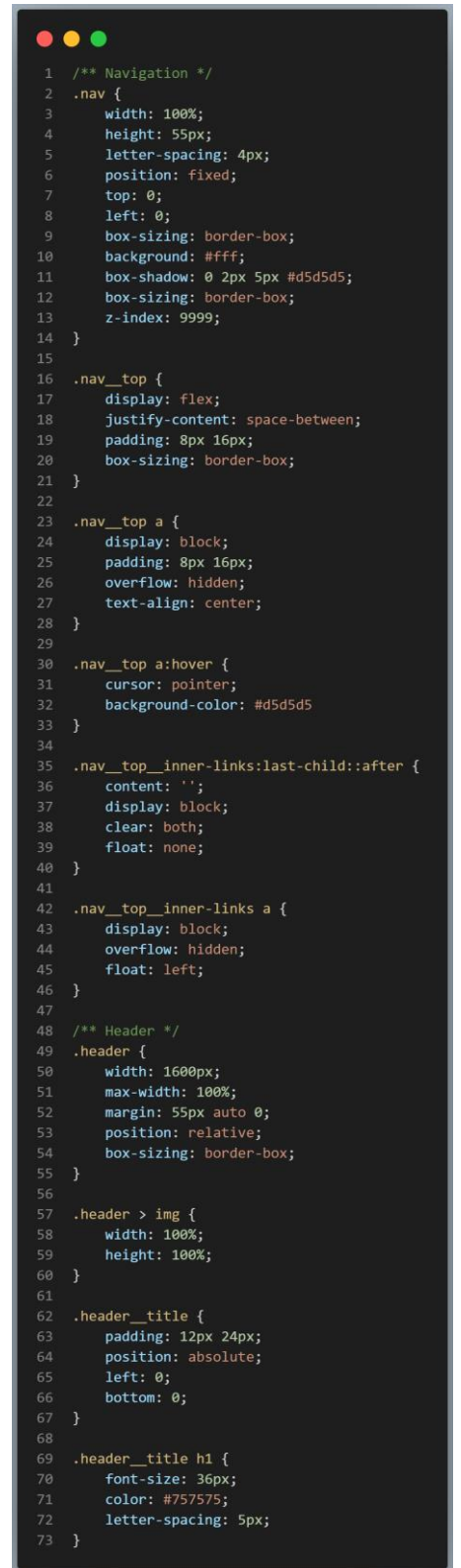
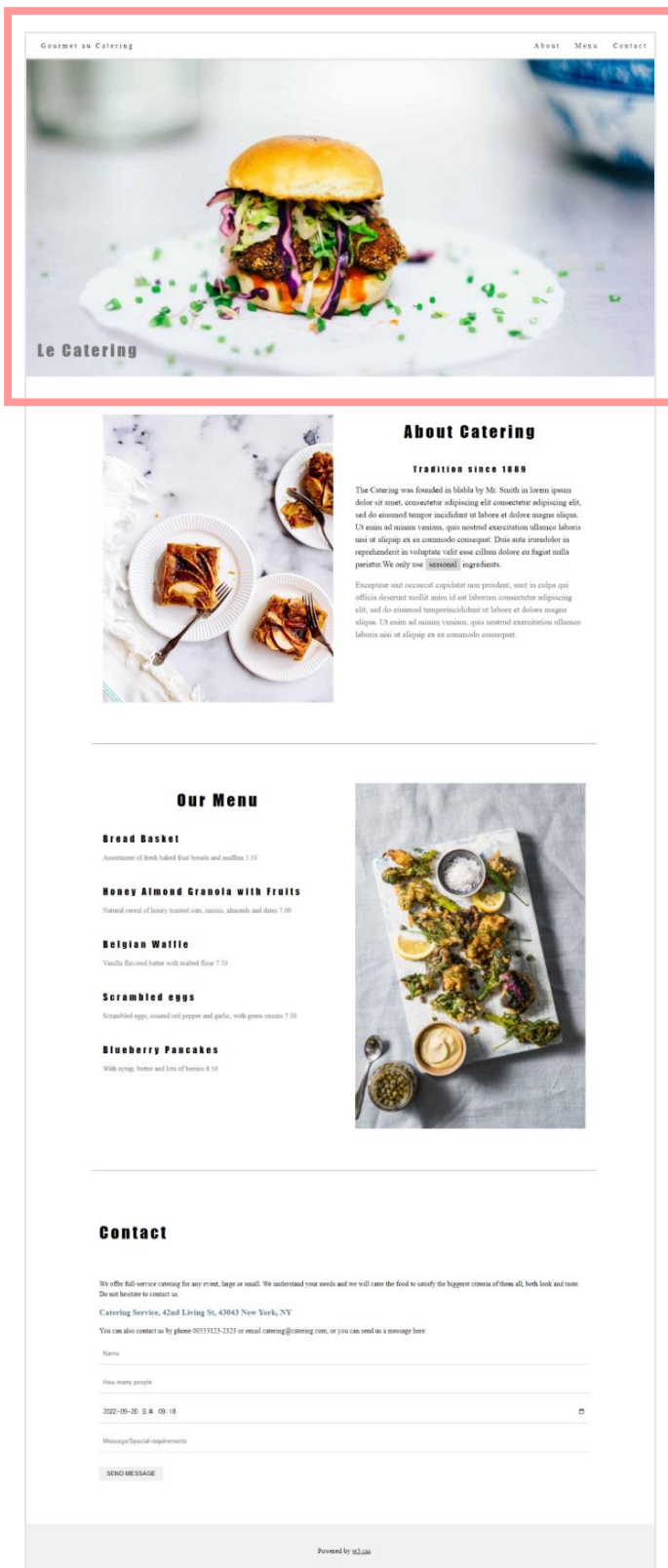
▼

▼ <HR>

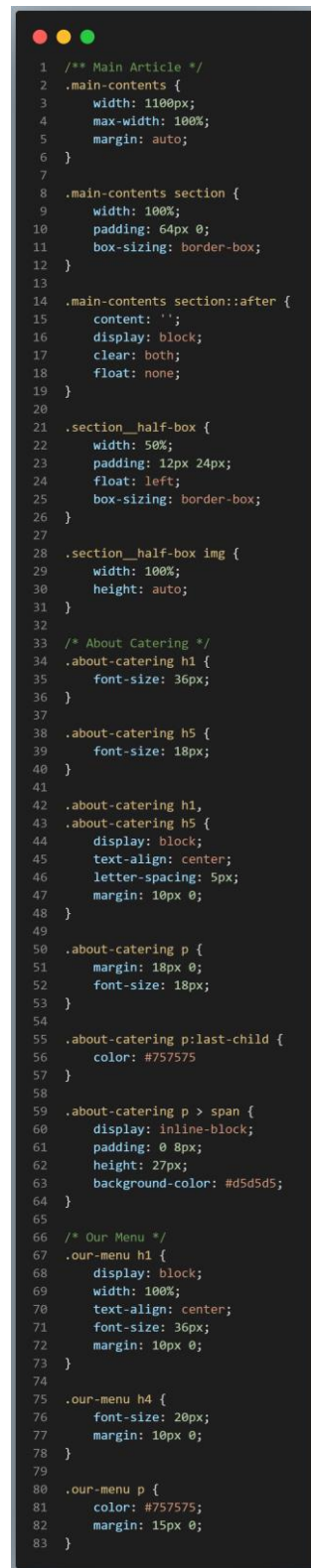
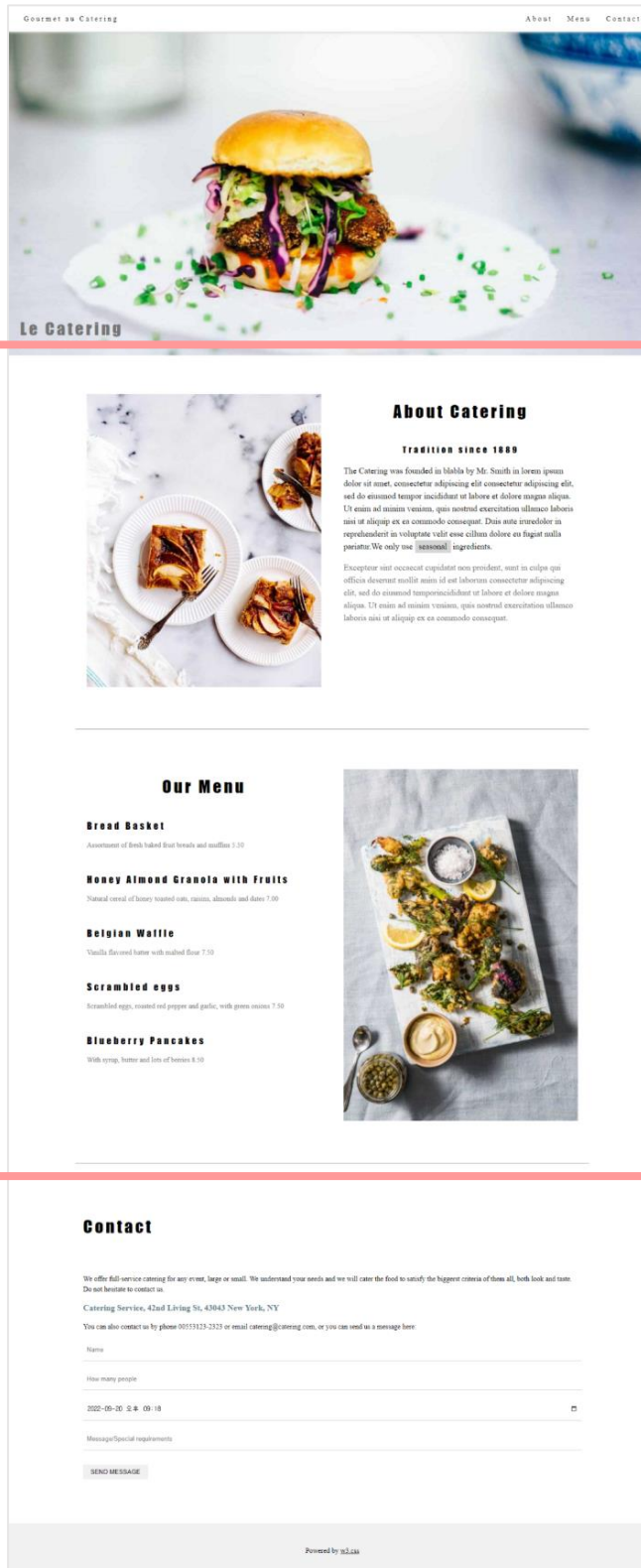
	▼ <SECTION>
	▼ <H1>
	▼
	▼ <P>
	▼ <P>
	▼
	▼ <P>
	▼ <FORM>
	▼ <P>
	▼ <INPUT>
	▼ <P>
	▼ <INPUT>
	▼ <P>
	▼ <INPUT>
	▼ <P>
	▼ <INPUT>
	▼ <P>
	▼ <BUTTON>
	▼ <FOOTER>
	▼ <P>
	▼ <A>
	▼ <SCRIPT>
	▼ <SCRIPT>
	▼ <SCRIPT>
	▼ <SCRIPT>
	▼ <SCRIPT>

*길이 문제로 나누어 개제하였습니다.

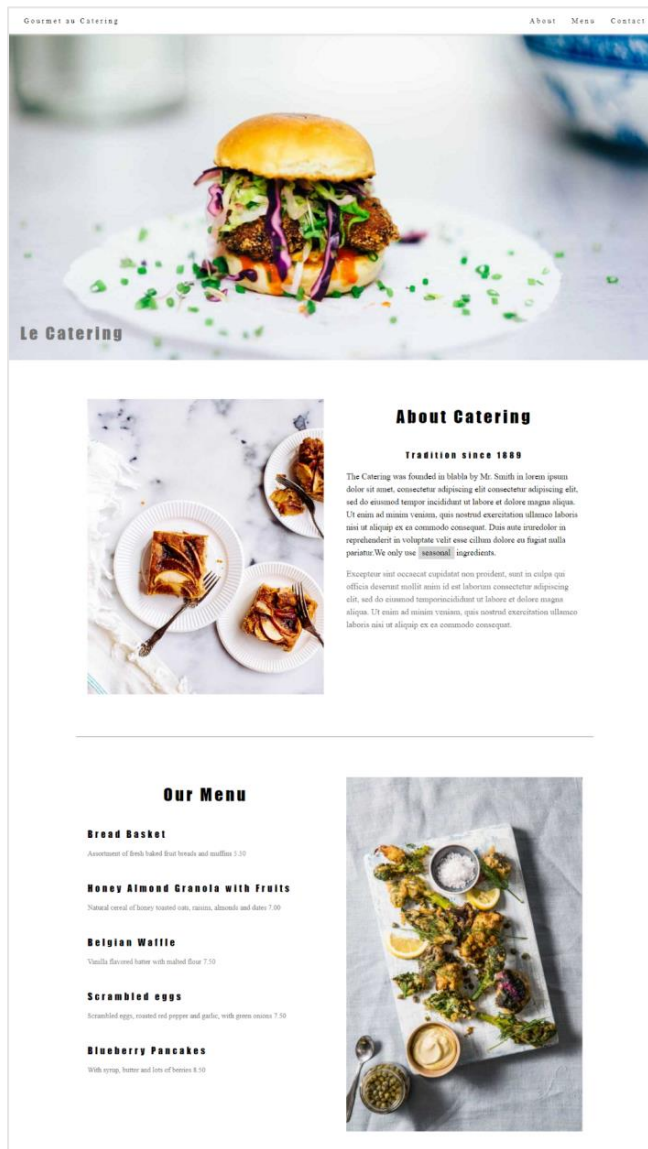
3. CSS nav, header



3. CSS article, section



3. CSS section, footer



```
1  /* Contact */
2  section.contact {
3      padding: 64px 16px;
4  }
5
6  .contact h1 {
7      font-size: 36px;
8  }
9
10 .contact b {
11     font-size: 18px;
12     color: #607d8b;
13 }
14
15 .contact__form,
16 .contact__form p,
17 .contact__form input {
18     display: block;
19     width: 100%;
20     box-sizing: border-box;
21 }
22
23 .contact__form p {
24     margin: 15px 0;
25 }
26
27 .contact__form input {
28     padding: 16px 8px ;
29     border: none;
30     border-bottom: 1px solid #cccccc;
31 }
32
33 .contact__submit {
34     margin: 16px 0;
35     padding: 8px 16px;
36     background-color: #f1f1f1;
37     border: none;
38 }
39
40 .contact__submit:hover {
41     cursor: pointer;
42 }
43
44 /** Footer */
45 .footer {
46     width: 100%;
47     background-color: #f1f1f1;
48     padding: 32px 0;
49 }
50
51 .footer p {
52     display: block;
53     width: 100%;
54     margin: 15px 0;
55     text-align: center;
56 }
57
58 .footer p a {
59     text-decoration: underline;
60     color: black;
61 }
```

3. CSS 공통, Media Query

```
1  body {
2      width: 100%;
3      margin: 0;
4      padding: 0;
5      font-size: 15px;
6      line-height: 1.5;
7      font-family: "Times New Roman", Georgia, Serif;
8      box-sizing: border-box;
9  }
10
11  h1, h2, h3, h4, h5, h6 {
12      font-family: "Playfair Display";
13      font-weight: 500;
14      letter-spacing: 5px;
15  }
16
17  a {
18      text-decoration: none;
19      color: #000;
20  }
21
22  li {
23      list-style: none;
24  }
```

```
1  @media screen and (max-width: 570px){
2      .nav__top__inner-links {
3          display: none;
4      }
5
6      .section__half-box {
7          width: 100%;
8          float: none;
9      }
10 }
```

4. JS

```
1  <!-- 네비게이션 내부 링크 처리 -->
2  <script type="module">
3      import Link from './js/InnerLinks.js';
4
5      document.querySelectorAll('a[data-target]')
6      .forEach(v => {
7          v.addEventListener('click', e => {
8              Link(e.currentTarget);
9              e.currentTarget.blur();
10         })
11     })
12 </script>
13
14 <!-- datetime-local <input> 태그의 시작 value 초기화 -->
15 <script type="module">
16     import timeSetter from './js/TimeSetter.js';
17     timeSetter();
18 </script>
19
20 <!-- form submit button 페이지 막기 // action="#"으로 처리해둠 -->
21 <script type="module">
22     // document.querySelector('#contact__submit').addEventListener('click', e => {
23     //     e.preventDefault();
24     // });
25 </script>
26
27 <script type="module">
28     import exploreDom from './js/Dom.js';
29
30     const body = document.querySelector('body');
31     const i = exploreDom(body);
32     console.log(i);
33 </script>
```

```
1  function Link(cur) {
2      const target = cur.dataset.target; // .header
3      document.querySelector(target).scrollIntoView();
4  }
5
6  export default Link;
```

**scrollIntoView 메소드와
dataset을 이용한 내부 링크**

```
1  function timeSetter() {
2      const date = new Date();
3      date.setHours(date.getHours() + 9);
4      document.querySelector("input[type='datetime-local']").value = date.toISOString().substring(0, 16);
5  }
6
7  export default timeSetter;
```

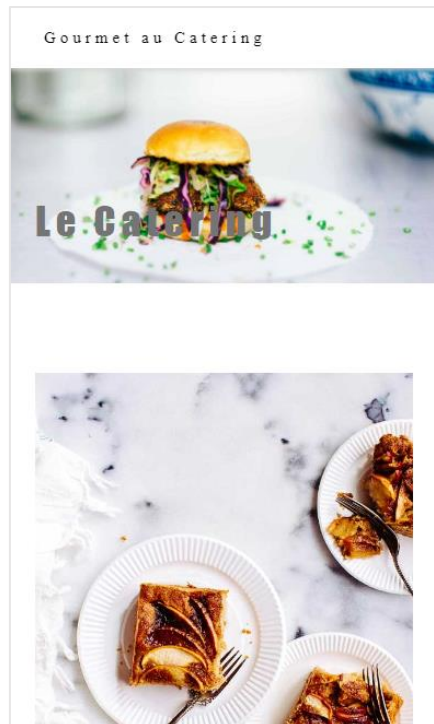
Date 객체를 이용한 시간 초기화

5. 작업 중 에러 처리

모바일 브라우저 사이즈에서
네비게이션 링크의 overflow,
메인 배너의 잘림 현상

-> Media Query와
max-width 속성 활용

본문의 내용이
브라우저 사이즈와 무관하게 항상 유지
-> float, media query 활용
창 크기 별 width 변경



메인 배너의 크기로 인해
모바일 화면에서 브라우저 밖으로
배너가 흘러 넘쳐
화면 쓸림 현상 발생
-> min-width 속성 삭제

입력 값 검사와
최초 시간 초기화
-> 수업 예제 활용 js 코드 처리

Contact

We offer full-service catering for any event, large or small. We understand your needs and we will cater the food to satisfy the biggest criteria of them all, both look and taste. Do not hesitate to contact us.

Catering Service, 42nd Living St, 43043 New York, NY

You can also contact us by phone 00553123-2323 or email catering@catering.com, or you can send us a message here:

Name

How many people

2022-09-21 오전 12:19



Message/Special requirements

SEND MESSAGE

6. 작업 소요 시간 및 소감

총 작업 소요 시간:

7시간

CSS는 역시 꾸준한 실습이 중요함을
깨닫게 되는 도전이었습니다.
첫 클론 코딩이나 마찬가지로인데,
어떠한 부분에서 원하는 결과가 나오지 않는지를
직접 실습하며 데이터를 쌓아보았습니다.
자신이 작성하는 코드의 결과가 어떻게 실행될지
명확히 이해하고 예측하며 작업하는 것이
매우 중요함을 알아가는 작업이었습니다.

감사합니다

↓ 결과물 웹 주소

http://sqhtjs0104.dothome.co.kr/catering_clone/