

2

Part 2. 수익형 웹 서비스 구현하기(1) - 사칙연산 계산기 만들기

2-1. 사칙연산 계산기 미리보기

계산기

1	2	더하기	빼기	곱하기	나누기	3
---	---	-----	----	-----	-----	---

2-2. 사칙연산 계산기의 프로그래밍적 원리 이해하기

사용자가 입력하는 것은 무엇인가

웹서비스를 기획하기 앞서 우리는 먼저 사용자 입장에서 생각해야 합니다. 계산기를 사용하는 사용자는 어떤 행동을 해야하는지가 중요해요. 1차적으로 우리가 만드는 사칙연산 계산기는 단순하게 3가지를 입력받도록 할겁니다.

1. 첫 번째 숫자
2. 두 번째 숫자
3. 연산자 (더하기, 빼기, 곱하기, 나누기)

입력받은 값을 어떻게 처리할까

일단 4개의 연산자 중 더하기부터 생각합시다. 프로그래밍은 아래의 순서대로 실행되어야 합니다.

1. 첫 번째 숫자를 (어딘가에) 저장한다.
2. 두 번째 숫자를 (어딘가에) 저장한다.
3. 첫 번째 숫자와 두 번째 숫자를 더해서 더한 결과를 저장한다.
4. 더한 결과를 표시한다.

가장 효율적인 프로그래밍 순서는, [더하기] 버튼을 누르는 순간 1, 2, 3, 4번 모든 순서가 실행되면 한 번에 해결할 수 있습니다.



프로그래밍에서는 어떤 값을 저장하는 공간을 변수(Variable)라고 합니다.

2-3. HTML로 계산기 틀 만들기

보일러플레이트 코드 사용하기

보일러플레이트는 프로젝트 시작 시 반복해서 사용하는 코드를 의미합니다. 우리는 앞으로 이 보일러플레이트 코드를 프로젝트 시작 시에 사용하도록 할 거예요.

💡 원래 보일러플레이트는 19세기에 증기 보일러를 만들때 틀로 사용하는 강철판을 의미했다. 보일러 플레이트에는 보일러 이름 등의 제조 정보가 음각으로 새겨져 있어서 보일러플레이트를 통과한 강철판에는 보일러 이름 등이 새겨지게 되었다. 이게 인쇄 분야로 이어져서 반복해서 사용하는 텍스트 (주로 광고나 로고)를 의미하게 되었다.

index.html

```
<!DOCTYPE html> <html> <head> <meta charset="UTF-8"> <meta name="viewport"
content="width=device-width, initial-scale=1" /> <title>Hello, world!</title> <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet"> <link rel="stylesheet" href="style.css"> </head> <body> <main
class="container"> <h1>Hello, world!</h1> </main> <script
src="https://code.jquery.com/jquery-3.6.1.min.js"></script> <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js">
</script> <script src="script.js"></script> </body> </html>
```

index.html

script.js

```
$(document).ready(function() { });
```

script.js

style.css

```
/* 비어있음 */
```

style.css

숫자를 입력받기 위한 텍스트박스 만들기

계산기에서 첫 번째 숫자와 두 번째 숫자를 입력받을 텍스트박스를 2개 만들어봅시다. 텍스트박스를 만들기 위해서는 아래처럼 body의 main.container 안에 입력해줍니다.

```
<input type="text" id="first-number" /> <input type="text" id="second-number" />
```

더하기 버튼 만들기

```
<button type="button" id="plus">더하기</button>
```

연산 결과 보여줄 공간 만들기

```
<div id="result"></div>
```

2-4. jQuery를 활용한 계산기 동작 이해하기

jQuery는 JavaScript를 더 쉽게 사용하기 위해 만들어진 라이브러리입니다. 다양한 브라우저에서 잘 동작하고 사용하기 쉬운 API를 통해 HTML 문서 및 조작, 이벤트 처리, 애니메이션 및 Ajax와 같은 것들을 훨씬 쉽게 만들 수 있습니다. 다재다능함과 확장성의 조합으로, 현재 전세계적으로 수백만 명의 개발자들이 사용하고 있습니다.

우리는 jQuery를 사용하여 사용자가 [더하기] 버튼을 누르는 순간을 캐치하고 버튼이 클릭되는 순간 첫 번째 숫자(first-number)와 두 번째 숫자(second-number)를 받아와 저장하고 첫 번째 숫자와 두 번째 숫자를 더해서 결과(result)를 표시하려고 합니다. 이것을 아래와 같은 순서로 나타낼 수 있겠죠.

1. 첫 번째 숫자(first-number)를 받아와 저장한다.
2. 두 번째 숫자(second-number)를 받아와 저장한다.
3. 첫 번째 숫자와 두 번째 숫자를 더한다.
4. 더한 결과를 표시한다.

클릭 인식하기

우리는 HTML을 작성할 때 버튼과 텍스트 인풋에 id를 각각 부여했습니다. 이 id는 각각의 버튼과 텍스트 인풋을 구분하기 위한 이름입니다. jQuery로 이 id를 이용하여 특정 요소를 선택할 수 있습니다. 예를 들어 버튼(plus)을 선택하고 싶다면 아래와 같이 코드를 작성합니다.

```
$("#plus")
```

위 코드는 뜻은 id(#)가 plus인 요소를 선택한다는 뜻입니다. 우리는 더하기 버튼을 클릭했을 때 뭔가를 실행하도록 만들고 싶습니다. 그러기 위해서는 다음과 같이 코드를 작성합니다.

```
function plusHandler() { /* 더하기 버튼을 클릭하면 이 부분이 실행됩니다. */ }  
$(document).ready(function() { $("#plus").click(plusHandler) });
```

텍스트 값 받아오기

clickHandler() 함수 안에 값을 받아오기 위한 코드를 작성합니다. 마찬가지로 id를 사용하여 first-number를 선택하여 값을 가져오겠습니다.

```
function plusHandler() { var firstNumber = $("#first-number").val();  
$("#result").text(firstNumber); }
```

val()을 사용하여 first-number의 값을 받아옵니다. 그리고 받아온 값을 firstNumber라는 변수에 저장합니다. 저장할 때는 = 기호를 사용합니다. 그리고 result라는 id를 가지고 있는 공간에 firstNumber 변수에 저장된 값을 텍스트로 출력(text)합니다.

first-number 텍스트 박스에 숫자를 입력하고 더하기 버튼을 누르면 텍스트 박스안에 있는 숫자가 그대로 표시됩니다.

계산기

1

자, 이런 원리만 알고 있다면 이제 우리는 무서울 것이 없습니다. 이번엔 두 번째 숫자도 받아와보죠.

```
function plusHandler() { var firstNumber = $("#first-number").val(); var secondNumber =
$("#second-number").val(); $("#result").text(firstNumber); }
```

second-number라는 id를 가지고 있는 요소를 선택하여 그것의 값(val)을 받아옵니다. 그리고 secondNumber라는 변수에 저장합니다.

더하기 구현하기

첫 번째 숫자와 두 번째 숫자를 가져왔으니 이제 두 숫자를 더할 차례입니다. 더하기는 정말 쉽습니다.

```
function plusHandler() { var firstNumber = $("#first-number").val(); var secondNumber =
$("#second-number").val(); var result = firstNumber + secondNumber;
$("#result").text(result); }
```

JavaScript에서 더하기는 + 기호로 구현합니다. firstNumber와 secondNumber에 저장된 값을 더해서 result 변수에 저장합니다. result 변수에 저장된 값을 id가 result인 요소에 출력합니다. 한 번 잘되는지 볼까요?

계산기

1	2	더하기
12		

우리가 예상했던 결과는 3인데 12라는 엉뚱한 값이 나왔습니다. 그 이유는 `val()`을 사용해서 값을 가져오면 숫자가 아닌 문자(String)로 인식하기 때문입니다. 단순히 1과 2를 앞뒤로 붙여서 12라는 결과가 나오는거죠. 3과 4를 입력하면 34가 나올 것입니다.

문자를 숫자로 바꾸기 위해 우리는 한 번 더 코드를 작성해야 합니다.

```
function plusHandler() { var firstNumber = $("#first-number").val(); var secondNumber =  
$("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =  
parseFloat(secondNumber); var result = firstNumber + secondNumber;  
$("#result").text(result); }
```

`parseFloat()`이라는 함수는 문자를 숫자(실수)로 바꿔주는 역할을 합니다. `firstNumber`에 저장된 값을 숫자로 바꿔서 `firstNumber`에 다시 저장하는 코드죠. 이렇게 바꿔주면 더하기가 정상적으로 동작할 겁니다.

계산기

1	2	더하기
3		

전체 코드

```
function plusHandler() { var firstNumber = $("#first-number").val(); var secondNumber =  
$("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =  
parseFloat(secondNumber); var result = firstNumber + secondNumber;  
$("#result").text(result); } $(document).ready(function() { $("#plus").click(plusHandler)  
});
```

2-5. jQuery를 활용한 사칙연산 구현하기

빼기 구현하기

더하기를 구현했으니 이번엔 빼기를 구현해봅시다. 빼기도 간단합니다. 아래와 같이 구현합니다.
plusHandler() 함수를 복사해서 minusHandler로 만든다음 + 를 - 기호로 바꿔주면 끝!

```
<button type="button" id="minus">빼기</button>
```

index.html

```
function minusHandler() { var firstNumber = $("#first-number").val(); var secondNumber =  
$("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =  
parseFloat(secondNumber); var result = firstNumber - secondNumber;  
$("#result").text(result); } $(document).ready(function() { $("#plus").click(plusHandler)  
$("#minus").click(minusHandler) });
```

script.js

곱하기 구현하기

JavaScript에서 곱하기 기호는 * 로 표시합니다.

```
<button type="button" id="multiply">곱하기</button>
```

index.html

```
function multiplyHandler() { var firstNumber = $("#first-number").val(); var secondNumber =
= $("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =
parseFloat(secondNumber); var result = firstNumber * secondNumber;
$("#result").text(result); } $(document).ready(function() { $("#plus").click(plusHandler)
$("#minus").click(minusHandler) $("#multiply").click(multiplyHandler) });
```

script.js

나누기 구현하기

JavaScript에서 나누기 기호는 / 로 표시합니다.

```
<button type="button" id="divide">나누기</button>
```

index.html

```
function divideHandler() { var firstNumber = $("#first-number").val(); var secondNumber =
$("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =
parseFloat(secondNumber); var result = firstNumber / secondNumber;
$("#result").text(result); } $(document).ready(function() { $("#plus").click(plusHandler)
$("#minus").click(minusHandler) $("#multiply").click(multiplyHandler)
$("#divide").click(divideHandler) });
```

script.js

전체 코드

계산기

1	2	더하기	빼기
곱하기	나누기		
0.5			

```
<!DOCTYPE html> <html> <head> <meta charset="UTF-8"> <meta name="viewport"
content="width=device-width, initial-scale=1" /> <title>계산기</title> <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css"
rel="stylesheet"> <link rel="stylesheet" href="style.css"> </head> <body> <main
class="container"> <h1>계산기</h1> <input type="text" id="first-number" /> <input
type="text" id="second-number" /> <button type="button" id="plus">더하기</button> <button
type="button" id="minus">빼기</button> <button type="button" id="multiply">곱하기
</button> <button type="button" id="divide">나누기</button> <div id="result"></div>
</main> <script src="https://code.jquery.com/jquery-3.6.1.min.js"></script> <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.min.js">
</script> <script src="script.js"></script> </body> </html>
```

index.html

```
function plusHandler() { var firstNumber = $("#first-number").val(); var secondNumber =  
$("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =  
parseFloat(secondNumber); var result = firstNumber + secondNumber;  
$("#result").text(result); } function minusHandler() { var firstNumber = $("#first-  
number").val(); var secondNumber = $("#second-number").val(); firstNumber =  
parseFloat(firstNumber); secondNumber = parseFloat(secondNumber); var result =  
firstNumber - secondNumber; $("#result").text(result); } function multiplyHandler() { var  
firstNumber = $("#first-number").val(); var secondNumber = $("#second-number").val();  
firstNumber = parseFloat(firstNumber); secondNumber = parseFloat(secondNumber); var  
result = firstNumber * secondNumber; $("#result").text(result); } function  
divideHandler() { var firstNumber = $("#first-number").val(); var secondNumber =  
$("#second-number").val(); firstNumber = parseFloat(firstNumber); secondNumber =  
parseFloat(secondNumber); var result = firstNumber / secondNumber;  
$("#result").text(result); } $(document).ready(function() { $("#plus").click(plusHandler)  
$("#minus").click(minusHandler) $("#multiply").click(multiplyHandler)  
$("#divide").click(divideHandler) });
```

script.js

2-6. Bootstrap을 활용한 계산기 디자인 꾸미기

우리는 동작하는 계산기를 만들었습니다. 한 가지 아쉬운 점은 디자인이겠죠? 유려한 디자인을 가진 서비스일 수록 사람들이 선호하기 마련입니다. 우리는 디자인을 하기위해 Bootstrap(부트스트랩)이라는 HTML, CSS 프레임워크를 사용하도록 하겠습니다.

Bootstrap은 강력하고 확장 가능하며 기능이 풍부한 프론트엔드 툴킷입니다. 다양한 그리드 시스템과 구성요소를 활용하며 웹 서비스에 활기를 불어넣을 수 있습니다.

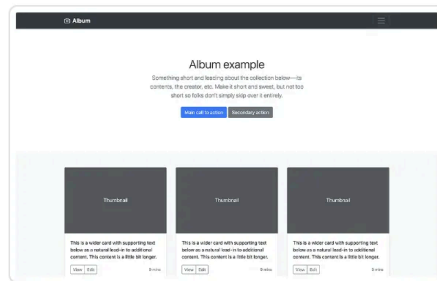
💡 위키피디아의 설명에 따르면 부트스트랩이란 Boot(s) + Strap의 합성어로서 부츠(Boots)의 뒷 부분에 달린 끈 혹은 고리(Strap)를 말합니다. 이 부트스트랩은 요즘 신발에도 달려나온다. 불가능한 일을 자력으로 해낸다는 의미인데 컴퓨터 분야에서는 보통 한 번 시작하면 다른 외부의 도움없이 스스로 진행하는 행위를 통틀어 사용하고 있습니다.

출처: <https://www.wisewiredbooks.com/term-dict/common/bootstrap.html>



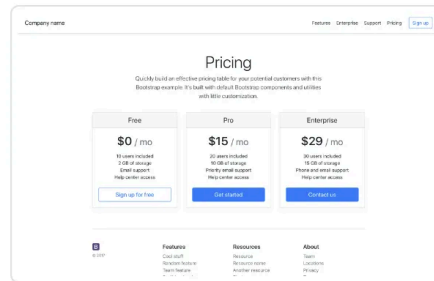
Bootstrap 예시

<https://getbootstrap.com/docs/5.2/examples/>



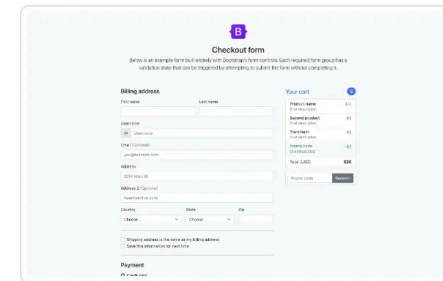
Album

Simple one-page template for photo galleries, portfolios, and more.



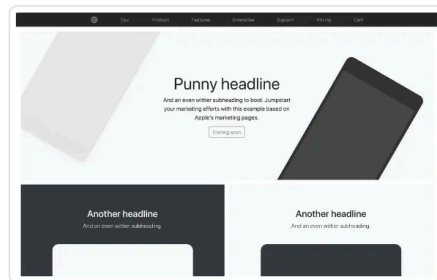
Pricing

Example pricing page built with Cards and featuring a custom header and footer.



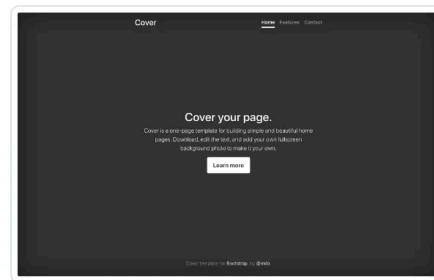
Checkout

Custom checkout form showing our form components and their validation features.



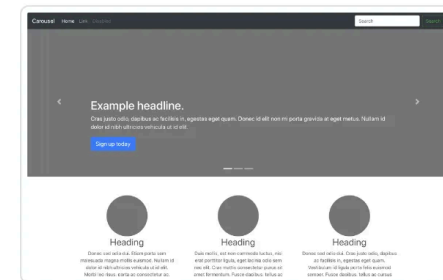
Product

Lean product-focused marketing page with extensive grid and image work.



Cover

A one-page template for building simple and beautiful home pages.



Carousel

Customize the navbar and carousel, then add some new components.

Bootstrap 문서

<https://getbootstrap.com/docs/5.2/getting-started/introduction/>

Bootstrap의 문서를 참고해서 한 번 디자인해보도록 하죠. 위의 문서를 보면 Bootstrap에서 제공하는 다양한 구성요소(Components)들을 복사+붙여넣기로 사용할 수 있습니다.

우리는 버튼 애드온으로 한 번 구현해보도록 하겠습니다.

버튼 애드온 예제: <https://getbootstrap.com/docs/5.2/forms/input-group/#button-addons>

Button addons

Button

Recipient's username

Button



Button

Button

Recipient's username

Button

Button

HTML  

```
<div class="input-group mb-3">
  <button class="btn btn-outline-secondary" type="button" id="button-addon1">Button</button>
  <input type="text" class="form-control" placeholder="" aria-label="Example text with
</div>

<div class="input-group mb-3">
  <input type="text" class="form-control" placeholder="Recipient's username" aria-label
  <button class="btn btn-outline-secondary" type="button" id="button-addon2">Button</button>
</div>

<div class="input-group mb-3">
  <button class="btn btn-outline-secondary" type="button">Button</button>
  <button class="btn btn-outline-secondary" type="button">Button</button>
  <input type="text" class="form-control" placeholder="" aria-label="Example text with
</div>

<div class="input-group">
  <input type="text" class="form-control" placeholder="Recipient's username" aria-label
```

```
<button class="btn btn-outline-secondary" type="button">Button</button>
<button class="btn btn-outline-secondary" type="button">Button</button>
</div>
```

Button Addon 구현하기

버튼 애드온 예제에서 가장 마지막 예제를 복사해서 index.html 파일에 붙여넣습니다.

```
<div class="input-group"> <input type="text" class="form-control"
placeholder="Recipient's username" aria-label="Recipient's username with two button
addons"> <button class="btn btn-outline-secondary" type="button">Button</button> <button
class="btn btn-outline-secondary" type="button">Button</button> </div>
```

계산기

Recipient's username		Button	Button
		더하기	빼기
곱하기	나누기		

버튼 애드온을 조금씩 수정해서 우리가 원하는 모양으로 만들어줍니다.

버튼 색 변경 예제: <https://getbootstrap.com/docs/5.2/components/buttons/#outline-buttons>

```
<div class="input-group"> <input type="text" class="form-control" placeholder="첫 번째 숫자"> <input type="text" class="form-control" placeholder="두 번째 숫자"> <button class="btn btn-outline-success" type="button">더하기</button> <button class="btn btn-outline-danger" type="button">빼기</button> <button class="btn btn-outline-primary" type="button">곱하기</button> <button class="btn btn-outline-warning" type="button">나누기</button> <span class="input-group-text"></span> </div>
```

계산기

첫 번째 숫자	두 번째 숫자	더하기	빼기	곱하기	나누기	
				더하기	빼기	
곱하기		나누기				

디자인이 조금 나아졌습니다. 하지만 계산기가 동작하지 않습니다. 왜일까요?

우리는 jQuery를 사용하여 id로 값을 받아오고 클릭을 인식했는데 우리가 지금 작성한 HTML 코드에는 id가 없어서 동작하지 않습니다. 각각의 텍스트 박스와 버튼 그리고 결과 공간에 id를 부여해줍시다. 그리고 처음에 작성한 코드는 지워줍니다. 그리고 계산기를 완성합니다!