

# 유노코딩과 함께하는 HTML & CSS 기초 강의

양식을 만들고  
입력 정보를 보내자





필요한 배경지식, 서버

입력 양식을 만들어 주는 <form> 태그는, 사용자가 입력한 데이터를 서버로 보내는 것을 목적으로 사용하는 태그이다. 따라서 서버에 대한 이해가 필요하다.

입력 양식이 사용된 페이지의 예

- 로그인 페이지
- 회원가입 페이지
- 검색어 입력 페이지

# NAVER

로그인

사용자 : 로그인 좀 시켜주세요

서버란, 정보를 제공하는 호스트(host)이다!

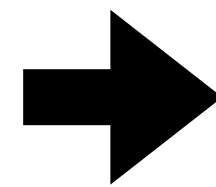
**NAVER**

my-mail

.....

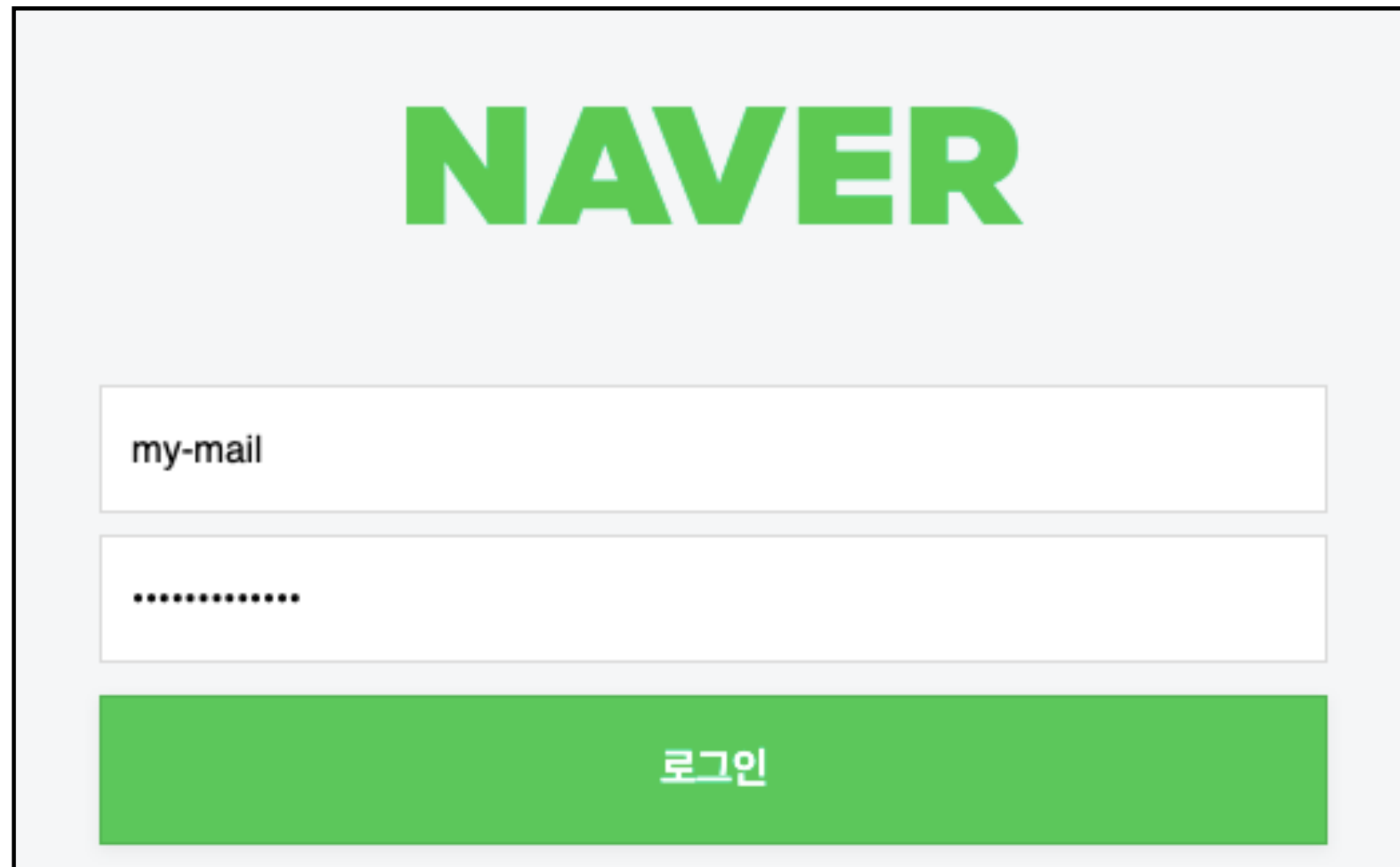
로그인

사용자 : 로그인 좀 시켜주세요

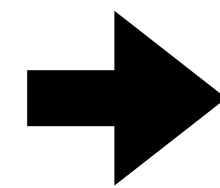


네이버 서버 : 알겠습니다

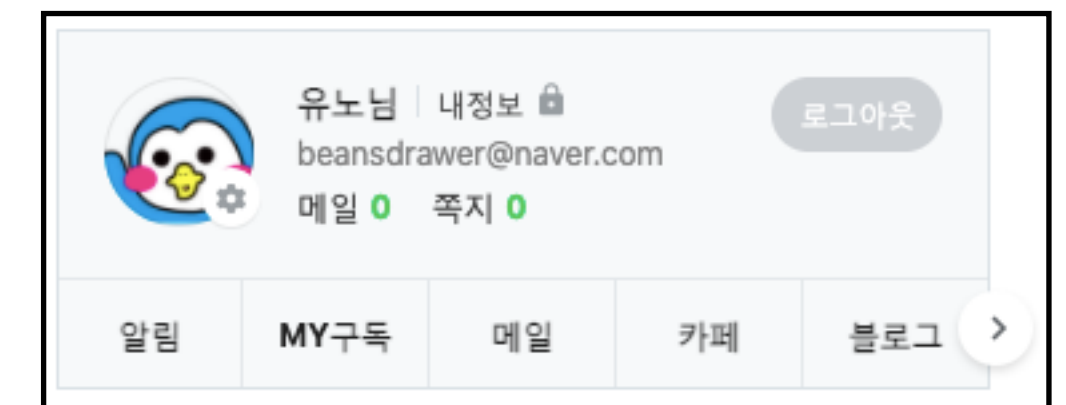
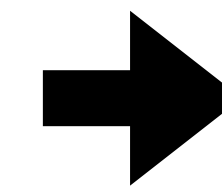
서버란, 정보를 제공하는 호스트(host)이다!

A screenshot of the Naver login page. At the top, the word "NAVER" is displayed in large green letters. Below it, there are two input fields: the first contains the text "my-mail", and the second contains a series of dots representing a password. At the bottom of the form is a green rectangular button with the white Korean text "로그인" (Login).

사용자 : 로그인 좀 시켜주세요



네이버 서버 : 알겠습니다



사용자 : 로그인 됐네요 ^^

서버란, 정보를 제공하는 호스트(host)이다!



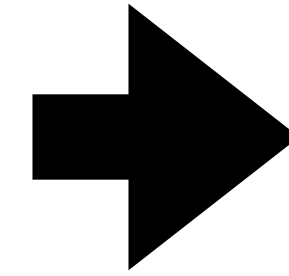
form

〈form〉은 입력 요소들을 감싸며, 입력 값을 서버 측으로 제출(submit)할 수 있다.  
〈form〉 태그의 안에 강의들에서 학습한 여러 입력 태그들을 포함시키면, 해당 입력 요소들은 양식의 개별 항목으로서 역할을 수행한다.

단순히 폼 요소 안에 입력 요소를 포함한 것만으로는 서버에 입력 값을 전송할 수 없다.  
〈form〉 태그가 역할을 수행할 수 있도록 다음 속성들을 사용하자.

- action : 입력값을 전송할 서버의 url
- method : 클라이언트가 입력한 데이터를 어떤 식으로 전송할지(GET or POST)

```
<form method="post" action="fake_server.php">  
  <label for="id">아이디</label>  
  <input type="text" id="id">  
  <br>  
  <label for="pw">비밀번호</label>  
  <input type="password" id="pw">  
  <br>  
  <input type="button" value="로그인">  
</form>
```



아이디

비밀번호

form 태그 사용법 및 결과



## GET vs POST

GET : 서버에 요청을 보내어 응답을 받아낸다.  
서버로부터 정보를 '가져오겠다'는 성격의 요청이다.

POST : 서버에 요청을 보내어 작업을 수행한다.  
서버에 있는 데이터를 추가/수정/삭제 한 후에 응답을 받아낸다.  
서버의 정보를 '조작하겠다'는 성격의 요청이다.

=> 더 자세한 이야기는 HTML을 공부한 이후라도 늦지 않습니다!





이름을 지어주세요

코드 작성자는 폼 요소 안에 필요한 만큼의 입력 요소를 포함해 다양한 형태의 양식을 만들 수 있다. 이때 폼 요소 안에 포함되는 각 입력 요소에는 name 속성을 추가하여 각 입력 항목의 역할을 구별해줄 수 있다.

```
<form method="post" action="fake_server.php">
  <label for="name">이름</label>
  <input type="text" id="name" name="name">
  <br>
  <label for="age">나이</label>
  <input type="number" id="age" name="age">
  <br>
  <label for="character">성격</label>
  <textarea id="character" name="character"></textarea>
  <br>
  <input type="button" value="전송">
</form>
```

서버가 정보를 받으면,  
name 속성으로 항목을 구별한다.  
따라서 name의 역할은  
무척 중요하다!



수고하셨습니다 :)