

이 매크로를 호출해서 폼에 사용할 입력 요소를 간단하게 만들 수 있다.

```
{{ input('item') }}
```

이것은 다음과 같은 HTML을 반환한다

```
<div class="form">
  <input type="text" name="item" value="" size="20">
</div>
```

다음으로 템플릿 상속이 무엇인지, FastAPI에서 어떻게 사용되는지 살펴보자.

템플릿 상속

Jinja의 가장 강력한 기능은 템플릿 상속이다. 이 기능은 중복 배제 Don't Repeat Yourself (DRY) 원칙에 근거한 것이며 큰 규모의 웹 애플리케이션을 개발할 때 많은 도움이 된다. 템플릿 상속 template inheritance은 기본 템플릿을 정의한 다음 이 템플릿을 자식 템플릿이 상속하거나 교체해서 사용할 수 있게 한다.

Jinja의 템플릿 상속을 자세히 알고 싶다면 웹 페이지²⁶를 참고하자.

지금까지 Jinja의 구문을 살펴봤다. 이제 FastAPI에서 Jinja 템플릿을 어떻게 사용하는지 알아보자.

4.2 FastAPI에서 Jinja를 사용하는 방법

Jinja를 사용하려면 Jinja2 패키지를 설치하고 기존 작업 디렉터리에 `template`이라는 신규 폴더를 만들어야 한다. 이 폴더에 모든 Jinja 관련 파일(Jinja 구문이 섞여 있는 HTML 파일)이 저장된다. 이 책에서는 사용자 인터페이스 user interface (UI) 디자인을 다루지 않으므로 스타일을 직접 작성하지 않고 CSS 부트스트랩 라이브러리를 활용한다.

²⁶ <https://jinja.palletsprojects.com/en/3.0.x/templates/#template-inheritance>