

4.1 Jinja

Jinja는 파이썬으로 작성된 템플리팅 엔진으로, API 응답을 쉽게 렌더링할 수 있도록 한다. 모든 템플리팅 언어에는 변수 *variable*가 사용되며 템플릿이 렌더링될 때 이 변수가 실제 값으로 변환된다. 이 외에도 태그를 사용해 템플릿 로직을 제어한다.

Jinja는 중괄호 `{}`를 사용해서 템플릿 파일의 일반적인 HTML, 텍스트 등을 표현식 및 구문과 구분한다. `{{ }}` 구문을 변수 블록 *variable block*이라고 하며 이 안에 변수를 지정한다. `{% %}`는 `if/else`, 반복 `loop`, 매크로 같은 구조를 제어할 때 사용된다.

Jinja 언어에서 자주 사용되는 구문은 다음과 같다.

- `{% ... %}`: 구조(처리)를 제어하기 위한 명령을 지정할 때 사용된다.
- `{{todo.item}}`: 식의 값을 전달할 때 사용된다.
- `{# 이 책은 훌륭한 API 책이다! #}`: 주석을 기입할 때 사용되며 웹 페이지상에는 표시되지 않는다.

Jinja는 문자열로 변환 가능한 모든 파이썬 유형 또는 객체를 템플릿 변수로 사용할 수 있다. 모델, 리스트, 딕셔너리 유형을 템플릿에 전달해서 값이나 속성을 사용할 수 있는데, 이때 `{{ }}` 구문이 사용된다.

이어서 필터에 대해 알아본다. 필터는 모든 템플리팅 엔진에서 가장 중요한 요소이며 특정 함수를 실행할 수 있게 해준다. 예를 들어 리스트의 값들을 병합 *join*하는 함수나 객체의 길이를 추출하는 함수 등을 사용할 수 있다.

이 외에도 Jinja의 기본 기능인 `if`문, 반복문, 매크로, 템플릿 상속을 살펴본다.

필터

파이썬과 Jinja는 유사한 구문을 사용하지만 문자열 병합이나 첫 문자를 대문자로 변환하기 등 파이썬의 문자열 수정 구문은 Jinja에서 사용할 수 없다. 따라서 이런 수정 작업은 Jinja의 필터 기능을 사용해야 한다.

필터는 다음과 같이 파이프 기호(`|`)를 사용해서 변수와 구분하며 괄호를 사용해 선택적 인수를 지정한다.