

장고 차근차근 시작하기

Second Edition

당신의 파이썬/장고 페이스메이커가 되겠습니다. ;)

EP22. 템플릿 상속을 통한 중복 제거

템플릿 상속의 필요성

- 각 뷰에 연결된 템플릿은 독립적으로 동작합니다.
 - 그런데, 독립적으로 동작하는 템플릿은 같은 레이아웃/스타일을 가지게 됩니다.
 - [네이버 블로그](#) 내 각 링크들은 서로 다른 주소 (광고에서는 서로 다른 뷰/템플릿) 이지만, 같은 스타일을 가집니다.
 - 즉, 템플릿 코드 중복이 발생하게 됩니다.
- 이러한 코드 중복을 해결하기 위한 솔루션 ⇒ 템플릿 상속

템플릿 상속의 특징

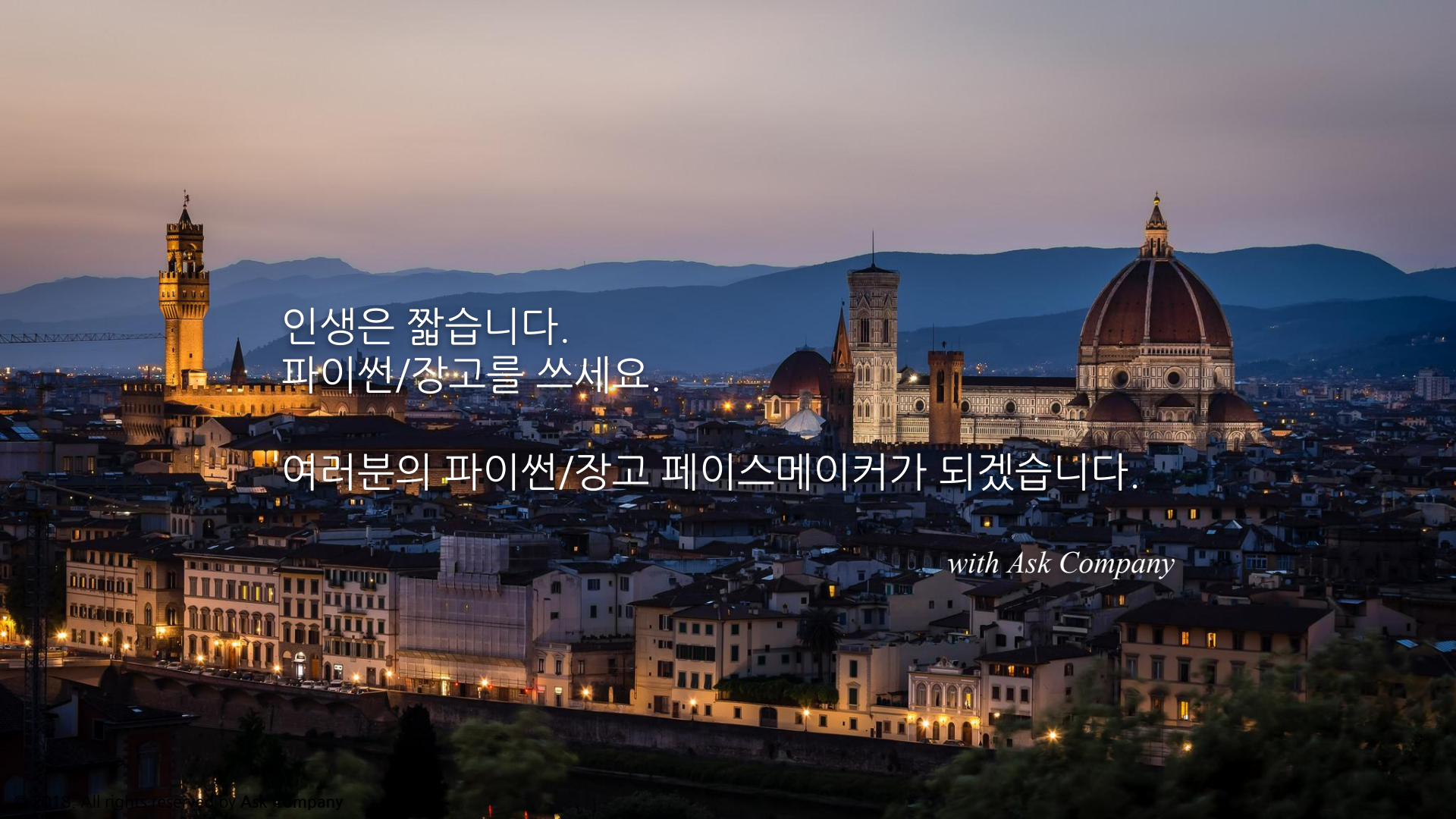
- 기본 특징
 - 상속은 여러 단계로 이뤄질 수 있다.
 - block에는 이름을 할당해야하며, 이름을 통해 구분하며, 한 템플릿 내에서 그 이름은 유일해야 한다.
- 부모 템플릿
 - 전체 레이아웃을 정의
 - 자식 템플릿이 비집고 들어올 수 있는 영역(block)을 다수 정의할 수 있다.
 - block이 없다면, 자식 템플릿은 상속만 받을 뿐, 어떠한 변경도 수행할 수 없다.
- 자식 템플릿
 - 상속받을 부모를 1개 지정할 수 있다.
 - 상속받은 부모에서 정의한 block에 대해, block 내용을 재정의하여 그 내용을 추가/변경/제거
 - block 바깥에 정의한 내용은 모두 무시된다.
 - 부모가 정의하지 않은 block을 재정의하여도 이는 무시된다.

사용하는 템플릿 태그

- {% extends “부모템플릿경로” %}
- {% block 블럭이름 %}블럭내용{% endblock %}
 - 부모에서 사용하면, 블럭 정의
 - 자식이 사용하면, 부모의 블록 재정의
- {{ block.super }}
- 자식 템플릿에서 사용할 때, 지정 위치에 부모 block 내용 출력

기본적으로 2단계의 상속을 추천

- 프로젝트 전반적인 레이아웃 템플릿 : askcompany/templates/layout.html
 - 각 앱 별 레이아웃 템플릿 : app/templates/app/layout.html -> layout.html 상속
 - 템플릿#1 : app/templates/app/post_list.html -> app/layout.html 상속
 - 템플릿#1 : app/templates/app/post_detail.html -> app/layout.html 상속
 - 템플릿#1 : app/templates/app/post_form.html -> app/layout.html 상속



인생은 짧습니다.
파이썬/장고를 쓰세요.

여러분의 파이썬/장고 페이스메이커가 되겠습니다.

with Ask Company