

Date: 0308-afternoon

# Django Intro

---

## Template 상속

---

- {% extends %}
  - 자식 템플릿이 부모 템플릿을 상속받는다는 걸 의미
  - 반드시 최상단에 작성되어야함
- {% block %}
  - overridden(재정의) 블록
  - 이 곳에는 하위템플릿이 따로 내용을 채울 수 있다

## django template system(설계 철학)

---

- 표현(template)과 로직(view)을 분리
  - template의 기능이 제한적이기 때문에 로직을 포함시키려는 순간 한계에 도달
  - 기본 목표를 넘어서려하면 안됨
- 중복 배제

## HTML form element

---

- 사용자 정보 입력 방식(text, button, checkbox, file, hidden, image, password, radio, reset, submit)과 서버전송 담당
- action : 전송될 URL
- method: 데이터 전달 방식

## HTML input element

---

- 사용자 데이터 입력 받음
- TYPE 속성에 따라 동작 방식 변화
- name(핵심속성, key값: 서버에 접근할 수 있는 방법)

## HTTP

---

- Hyper Text Transfer Protocol
- 웹 데이터 교환의 기초
- 검색해서 정보를 얻는다 request.method.GET

# request.method.GET

---

- 데이터의 정보를 조회하고 가져올때 사용
- body가 아닌 Query String Parameter(url)에 붙어서 전송된다.

## Django URLs

---

- dispatche: 발송자로서의 역할
- path()와 app이 많아지면 유지보수가 안좋기 때문에, 각 app에 urls.py를 따로 작성

## Naming url patterns

---

- path안에 name인자로 이름 지정
- 사용시, url tag를 사용하여 절대경로를 사용하지 않고 특정 경로의 의존성을 줄일 수 있음

```
{% url 'index'%}
```

## Name space

---

- 서로 다른 app의 같은 이름을 가진 이름 공간(폴더)를 만들어서 구분해준다.

```
{% url 'article:index'%}
```

- 기본적으로 경로에 있는 templates 폴더만 검색하기 때문에 app\_name> templates> app\_name 형태로 구분