

올인원 패키지 Online.

Django URLs, Views, Models

PART1 | Frontend, Backend 그리고 DB

PART2 | ORM 이란?

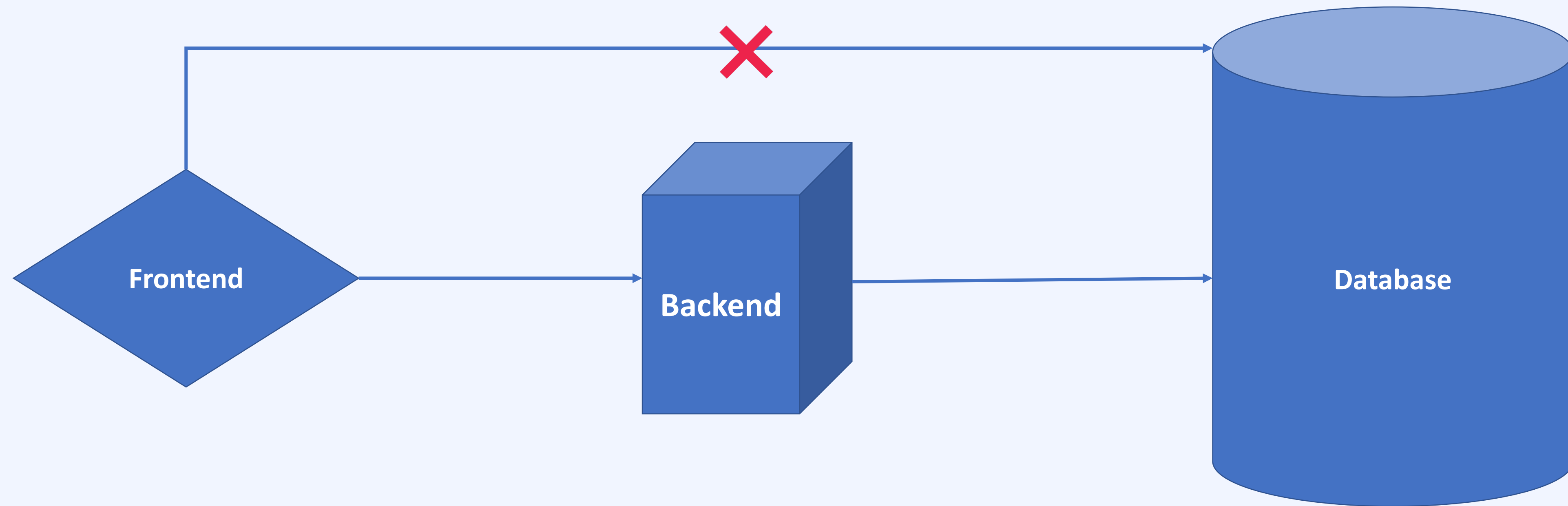
PART3 | Django ORM

Github : <https://github.com/riseryan89/shrinkers>

Commit ID: e1fccd4

1. Frontend, Backend 그리고 DB

1.
Django ORM



2. ORM이란?

Object Relational Mapping, 객체-관계 매핑

- Python 객체와 관계형 데이터베이스 연결
- SQL 쿼리를 자동으로 생성함
- 데이터베이스 데이터 <—매핑—> Python Object
- 객체를 통해 간접적으로 데이터베이스 데이터 핸들링
- Django ORM, SQLAlchemy

장점	단점
<ul style="list-style-type: none">• 직관적이다.• SQL 학습 시간 보다 Business Logic에 더 집중할 수 있다.• 가독성이 올라간다. Zen of Python• 코드 재사용 및 유지보수가 수월함• DBMS 종속성이 줄어든다.• SQL Injection 을 막아준다.	<ul style="list-style-type: none">• Raw 쿼리가 필요한 구간이 반드시 있다.• 프로젝트의 복잡도가 올라가면, ORM 난이도도 올라간다.• 프로시저 사용이 수월하지 않다.

3. Django ORM

공식문서 : <https://docs.djangoproject.com/en/3.2/topics/db/queries/>

.get()
.all()
.filter()
.save()
.delete()
.update()
.exists()
.first()
.count()
.order_by()