



Делаем запросы из кода

- Не очень удобно
- Не очень красиво
- Сложно переиспользовать
- При смене базы данных - нам нужно все переписать

Практика

Файл "psysorg2_ex.py"

ORM

Object-Relational Mapper

ORM

- Позволяет использовать настоящие python объекты для работы с данными
- Позволяет "не знать SQL"
- Позволяет не брать во внимание конкретные реализации баз данных
- Помогает нам жить (есть мнение, что мешает)

SQLAlchemy

- Мощнейший инструмент (возможно - самый)
- Один из самых популярных ORM для python
- Прост в использовании
- Может быть декларативен

Модель

- Модель в терминологии ORM - по сути таблица в базе
- Свойство модели - колонка в таблице
- Но модель к тому же - еще и python объект
- Много возможностей

Формы для моделей

- Нам знакомы формы - мы используем их для валидации данных
- Теперь нам нужно валидировать данные для модели
- Но модель уже определяет поля, которые использует. А мы можем создавать формы из моделей в пару строк

Практика

Файлы "sqlalchemy_example/*"

Индексы

- Увеличивают скорость работы запросов
- По-умолчанию создаются для нас ORM, но нам нужно за ней проверять и досоздавать

Отношения

- В SQL отношения устанавливаются через ForeignKey
- Но значение внутри - просто число
- Было бы удобно, если ORM брал число, искал по нему нужные данные и превращал их в объект
- Нам поможет `db.relationship`

Транзакции

- Гарантия того, что наши данные будут целы
- Гарантия атомарности операции