# [SQLAlchemy](https://habr.com/ru/post/196810/)

В SAQLAlchemy взаимодействие с базой данных происходит с помощью [**сессии**](https://pythonru.com/uroki/13-sessii-vo-flask). К счастью, ее не нужно создавать вручную. Это делает Flask-SQLAlchemy. Доступ к объекту сессии можно получить с помощью db.session. Это объект сессии, которые отвечает за подключение к базе данных. Он же отвечает за процесс транзакции. По умолчанию транзакция запускается и остается открытой до тех пор, пока выполняются коммиты и откаты.

Добавление объектов не записывает их в базу данных, этот процесс лишь готовит их для сохранения при следующем коммите.

my\_obj = Object(name = ‘’, description = ‘’)  
db.session.add(my\_obj) или для нескольких: db.session.add\_all([my\_obj1, my\_obj2])

db.session.new – чтобы посмотреть все объекты сессии

Для сохранения объектов в базе данных нужно вызвать метод commit():

db.session.commit()

Для удаления объектов в БД:

db.session.delete(my\_obj) – чтобы удалить объект

В некоторых случаях может возникнуть необходимость удалить все дочерние записи при том, что родительские записи уже удалены. Для этого нужно в моделях (объектах для БД) изменить свойство поля:

my\_obj = db.relationship(…, cascade='all,delete-orphan')

Чтобы выполнить запрос к базе данных, используется метод query() объекта session. Метод query() возвращает объект flask\_sqlalchemy.BaseQuery, который является расширением оригинального объекта sqlalchemy.orm.query.Query. Объект flask\_sqlalchemy.BaseQuery представляет собой оператор SELECT, который будет использоваться для осуществления запросов к базе данных.

Основные методы для работы с запросами:

|  |  |
| --- | --- |
| all() | Возвращает результат запроса (представленный flask\_sqlalchemy.BaseQuery) в виде списка. |
| count() | Возвращает количество записей в запросе. |
| first() | Возвращает первый результат запроса или None, если в нем нет строк. |
| first\_or\_404() | Возвращает первый результат запроса или ошибку 404, если в нем нет строк. |
| get(pk) | Возвращает объект, который соответствует данному первичному ключу или None, если объект не найден. |
| get\_or\_404(pk) | Возвращает объект, который соответствует данному первичному ключу или ошибку 404, если объект не найден. |
| filter(\*criterion) | Возвращает новый экземпляр flask\_sqlalchemy.BaseQuery с оператором WHERE. |
| limit(limit) | Возвращает новый экземпляр flask\_sqlalchemy.BaseQuery с оператором LIMIT. |
| offset(offset) | Возвращает новый экземпляр flask\_sqlalchemy.BaseQuery с оператором OFFSET. |
| order\_by(\*criterion) | Возвращает новый экземпляр flask\_sqlalchemy.BaseQuery с оператором OFFSET. |
| join() | Возвращает новый экземпляр flask\_sqlalchemy.BaseQuery после создания SQL JOIN. |

Примеры:

db.session.query(Post).all()  
db.session.query(Post).first()  
db.session.query(Post).filter(Post.title == 'Post 1' , Post.id <= 2).all()  
db.session.query(Post).filter(Post.id >= 2).limit(5).all()  
db.session.query(Post).join(Category).all()