Manipulating Your Database

Xavier Morera

HELPING DEVELOPERS UNDERSTAND SEARCH & BIG DATA

@xmorera www.xaviermorera.com

Manipulating Your Database

- Update data
- Advanced update techniques
- Delete data
- Delete tables

```
engine = db.create_engine('sqlite:///sqlalchemy_sqlite.db')

connection = engine.connect()

engine.execute('select * from post where Id=1').fetchone()

engine.execute('select ViewCount from post where Id=1').fetchone()

engine.execute('Update post set ViewCount = 0 where Id = 1')
```

Updating Data Using SQL

Use a SQL statement

Update the value from one column

```
query = db.update(Post).\
           where(Post.Id == 1).\
           values(ViewCount=1)
result = connection.execute(query)
post_query = my_session.query(Post).filter(Post.Id == 1)
post_query.ViewCount
```

Updating Data Using Update

Select table with update

Specify row with where

Specify values

And execute

```
query = db.update(Post).\
        values(ViewCount=Post.ViewCount + 50)
result = connection.execute(query)
```

Updating Multiple Records

Similar to one record update

- Can use a where to specify multiple records
  - But can also be missing
- Potentially dangerous if no where clause

```
my_post = my_session.query(Post).filter(Post.Id == 1).one()

my_post.Title

my_post.Title = 'Modified Question'

my_session.commit()
```

Updating an Object

Query to get an object

- Modify
- Commit
- Confirm

```
avg_views = db.select([db.func.avg(Post.ViewCount).\
  label('Average_Views')])
query = db.update(Post).values(ViewCount=avg_views)
connection_result = connection.execute(query)
```

Correlated Updates

Update uses the result of another query

- Create the query
- Update, specifying the query

Confirm

```
my_session.query(Post.Id).all()

first_post = my_session.query(Post).first()

first_post.Id

my_session.delete(me)

my_session.commit()
```

Deleting a Record

Use SQL

Use delete from session

- **Specify the object, and remember to commit**

```
my_session.query(Post.Id).all()

my_session.query(Post).filter(Post.Id > 2).delete()
```

Deleting Multiple Records

Using query

- Specify which records

Call delete and commit

- Returns number of records affected

```
metadata = db.MetaData()

metadata.reflect(bind=engine)

metadata.tables

metadata.tables.keys()
```

Deleting a Table

Use reflect to load database information

Get table names

- Use metadata.tables

```
post_table = metadata.tables['post']

post_table.drop(bind=engine)
```

Deleting a Table

Create Table object

Use drop

- Remember to bind to your engine

Confirm

Takeaway

Update data

- SQL statements
- Using update, with where, and values
- Modify an object

Correlated updates

Takeaway

Delete data

- Use session.delete()
- Call delete() on a query

Delete tables

- Use drop() on Table object