

Introducing SQLAlchemy and Our Project



Chris Venturini

@ChrisMVenturini www.chrisv.me





SQLAlchemy

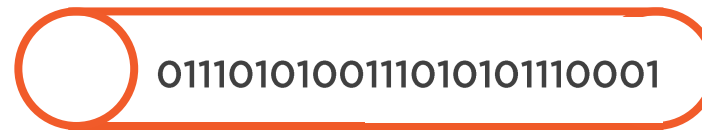
“SQLAlchemy is the Python SQL toolkit and Object Relational Mapper that gives application developers the full power and flexibility of SQL”





0111010100111010101110001

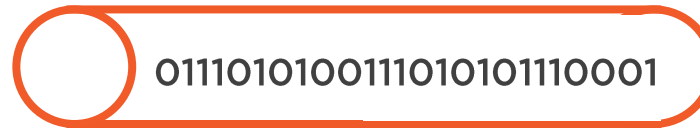




```
new_user_id = Users.create('Jade')
```

	hero_id	name
	Filter	Filter
1	1	Crareek
2	2	Tan
3	3	Huzzt
4	4	Pamble
5	5	Grimm





```
battle_event_type_id =  
    database.find_battle_event_by_name(  
        "Hero_Kill")
```

```
database.insert_battle_participant(  
    battle_paticipant_id,  
    battle_event_type,  
    ...)
```

	battle_event_id	battle_participant_id	battle_event_type_id	rubies_gained
	Filter	Filter	Filter	Filter
1	1	5	3	3
2	2	5	3	5
3	3	1	3	1





```
Table('user', metadata,  
      Column('user_id', Integer, primary_key=True)  
)
```

	hero_id	name
	Filter	Filter
1	1	Crareek
2	2	Tan
3	3	Huzzt
4	4	Pamble
5	5	Grimm

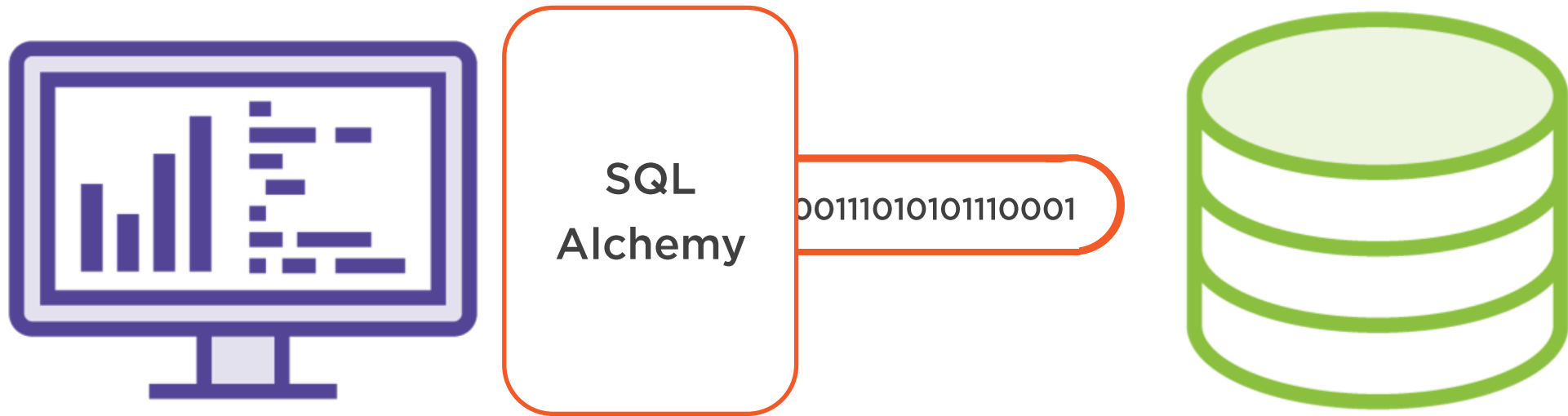




```
Table('user', metadata,  
      Column('user_id', Integer, primary_key=True)  
)
```

```
CREATE TABLE user  
(  
    user_id INT NOT NULL PRIMARY KEY,  
);
```



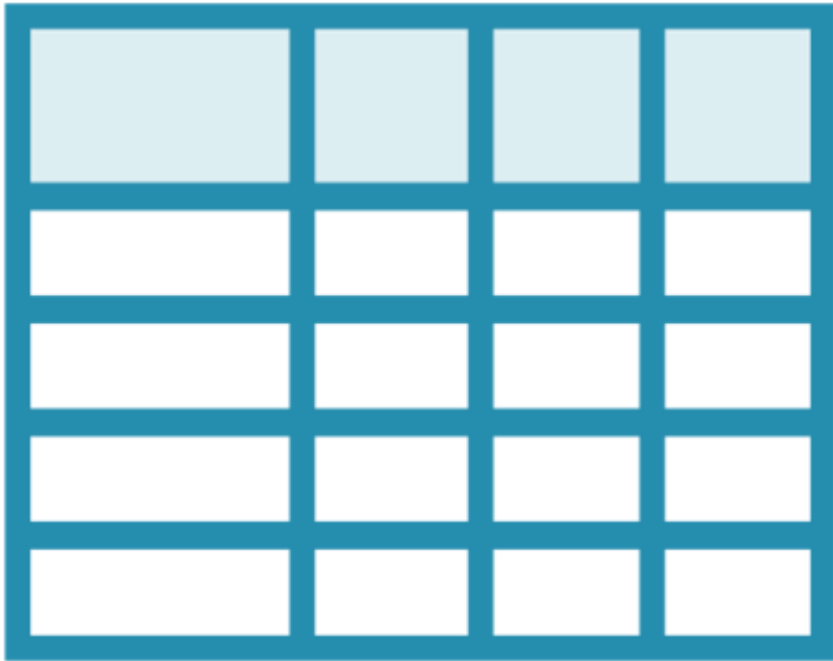


What about the object relationship mapper (ORM)?



SQLAlchemy

Core



ORM



SQLAlchemy

Core

```
user_table = Table('user', metadata,  
    Column('user_id', Integer, primary_key=True),  
    Column('name', String)  
)  
  
select([user_table.columns.name])
```

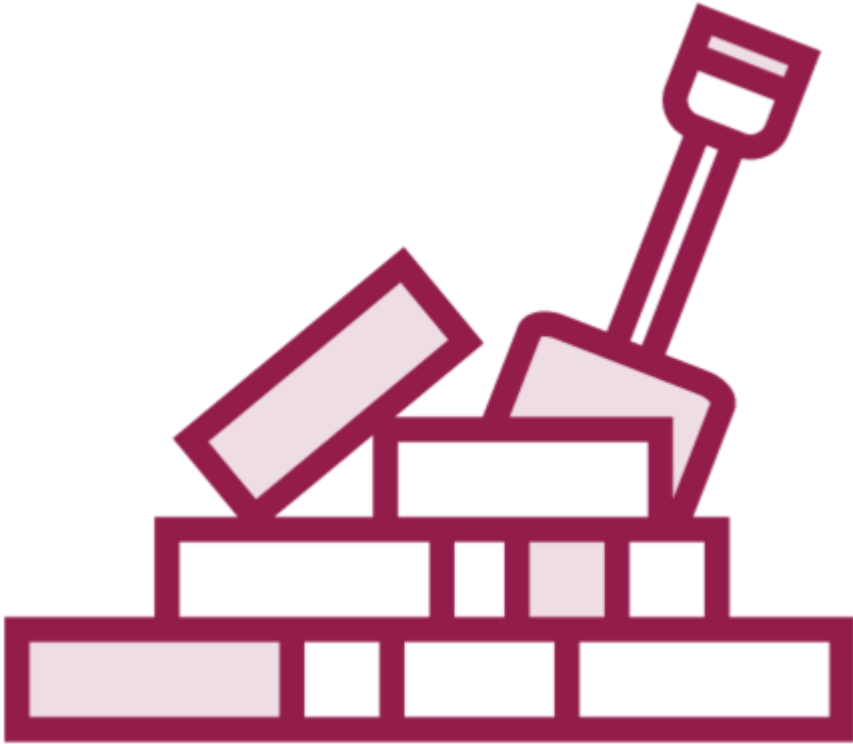
ORM

```
class User(Base):  
    __tablename__ = 'user'  
    id = Column(Integer, primary_key=True)  
    name = Column(String)  
  
    def my_function(self)  
        pass  
  
session.query(User, User.name).all()
```

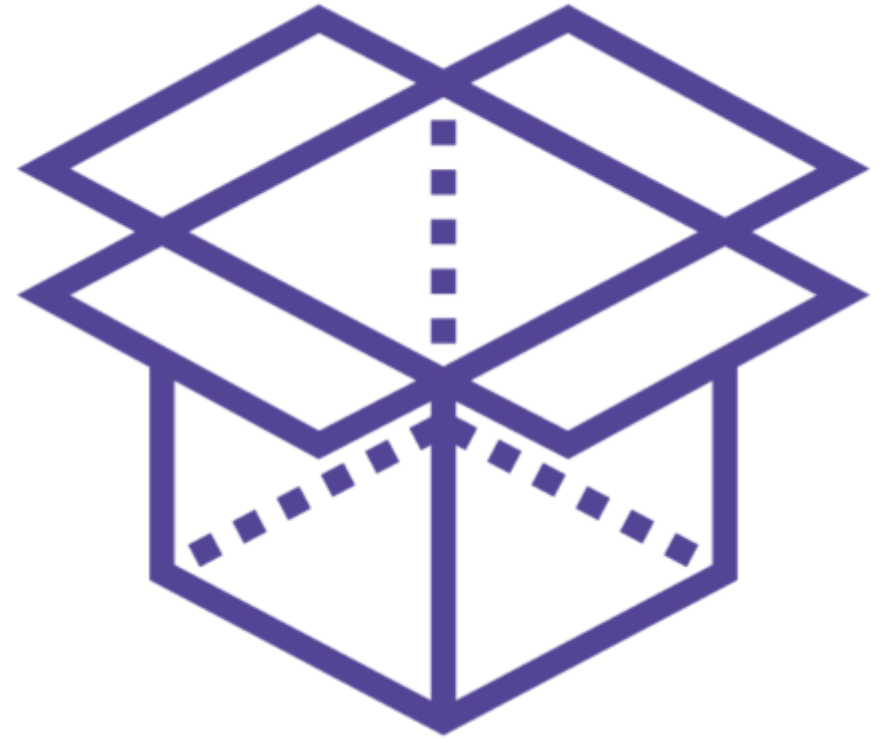


SQLAlchemy

Core

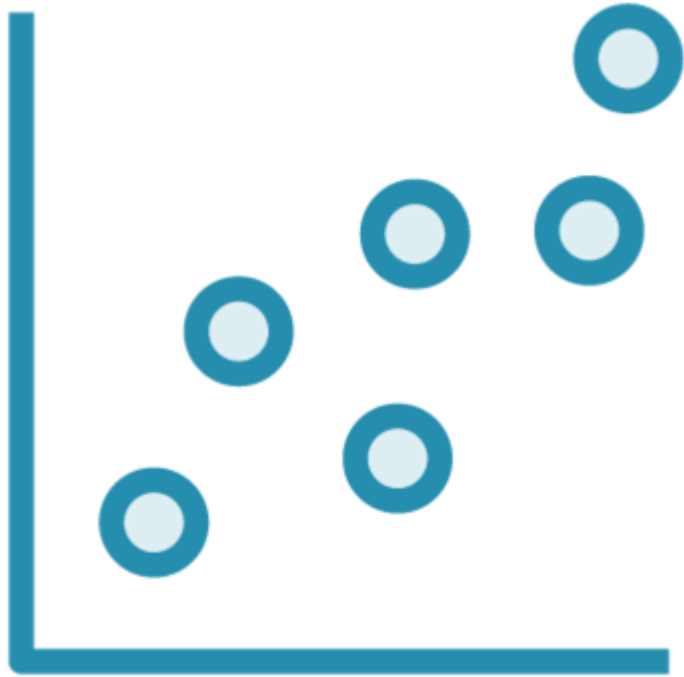


ORM



SQLAlchemy

Core



ORM



HEROES of the Endless War

Jade



Grimm



Mako



