**subquery的使用\_高级查询之子查询**

**subquery方法：**

子查询即select语句中还有select。

那么在sqlalchemy中，要实现一个子查询，需以下几个步骤：

**1.** 将子查询按照传统的方式写好查询代码，然后在`query`对象后面执行`subquery`方法，将这个查询变成一个子查询。

**2.** 在子查询中，将以后需要用到的字段通过`label`方法，取个别名。

**3.** 在父查询中，如果想要使用子查询的字段，那么可以通过子查询的返回值上的`c`属性拿到（c=Column）。

**代码演示：**

|  |
| --- |
| **from** sqlalchemy **import** create\_engine,Column,Integer,Float,Boolean,DECIMAL,Enum,\  Date,DateTime,Time,String,Text,func,or\_,and\_,ForeignKey,Table  **from** sqlalchemy.dialects.mysql **import** LONGTEXT  **from** sqlalchemy.ext.declarative **import** declarative\_base  **from** sqlalchemy.orm **import** sessionmaker,relationship,backref  **import** random,time  **from** datetime **import** datetime  HOSTNAME = **'127.0.0.1'**  PORT = **'3306'**  DATABASE = **'first\_sqlalchemy'**  USERNAME = **'root'**  PASSWORD = **'root'**  DB\_URI =**"mysql+pymysql://{username}:{password}@{host}:{port}/{db}?charset=utf8"**.format(username=USERNAME,password=PASSWORD,host=HOSTNAME,port=PORT,db=DATABASE)  engine = create\_engine(DB\_URI)  Base = declarative\_base(engine)  session = sessionmaker(engine)()  **class** User(Base):  \_\_tablename\_\_ = **'user'**  id = Column(Integer,primary\_key=**True**,autoincrement=**True**)  uname = Column(String(50),nullable=**False**)  city = Column(String(50),nullable=**False**)  age = Column(Integer,default=0)  **def** \_\_repr\_\_(self):  **return "<User(username: %s)>"** % self.uname  **def** add\_data():  Base.metadata.drop\_all()  Base.metadata.create\_all()  user1 = User(uname=**'一哥'**,city=**"贵阳"**,age=18)  user2 = User(uname=**'王二'**,city=**"贵阳"**,age=18)  user3 = User(uname=**'张三'**,city=**"北京"**,age=18)  user4 = User(uname=**'赵四'**,city=**"贵阳"**,age=20)  session.add\_all([user1,user2,user3,user4])  session.commit()  # 相亲类网站：同城交友 之珍爱网  #需求： 寻找和 “一哥” 这个人在同一个城市，并且是同年龄的人  #实现思路1：传统方式  **def** oper1():  u = session.query(User).filter(User.uname == **'一哥'**).first()  users = session.query(User).filter(User.city==u.city,User.age==u.age).all()  print(users)  #实现思路2：子查询方式  #原生sql：select `user`.id,`user`.uname,`user`.city,`user`.age from user,  # (select `user`.city,`user`.age from user where uname='一哥') as yige  # where `user`.city=yige.city AND `user`.age=yige.age  **def** oper2():  # stmt = session.query(User.city.label('city'), User.age.label('age')).filter(User.uname == '一哥').subquery()  # result = session.query(User).filter(User.city == stmt.c.city, User.age == stmt.c.age)  # print(result) #查看sql语句  stmt = session.query(User.city.label(**'city'**), User.age.label(**'age'**)).filter(User.uname == **'一哥'**).subquery()  result = session.query(User).filter(User.city == stmt.c.city, User.age == stmt.c.age).all()  print(result) # 查看结果  **if** \_\_name\_\_ == **'\_\_main\_\_'**:  # add\_data()  # oper1()  oper2() |