

牛人计划-初级项目课（3）



牛客网
NOWCODER



第三课



课程目录

CONTENTS

- 项目结构搭建
- Flask-SQLAlchemy安装
- ORM介绍
- 数据库切换/初始化
- 插入数据
- 查询/分页
- 删除数据
- 更新数据
- 模板继承
- 首页开发



项目结构搭建（代码演示）

```
application
├── manage.py      <-脚本数据
├── runserver.py   <-启动服务器
├── application    <-web目录
│   ├── app.conf  <-配置文件
│   ├── models.py <-数据模型
│   ├── views.py  <-视图
│   └── __init__.py <-模块导出文件
├── static         <-静态文件
└── templates     <-页面模板
    ├── base.html
    ├── index.html
    └── login.html
```



Flask安装和框架（官网演示）

Flask官网：<http://flask-sqlalchemy.pocoo.org/>
中文 <http://www.pythondoc.com/flask-sqlalchemy/>
Flask安装：pip install Flask-SQLAlchemy
MySQLdb：pip install Flask-MySQLdb



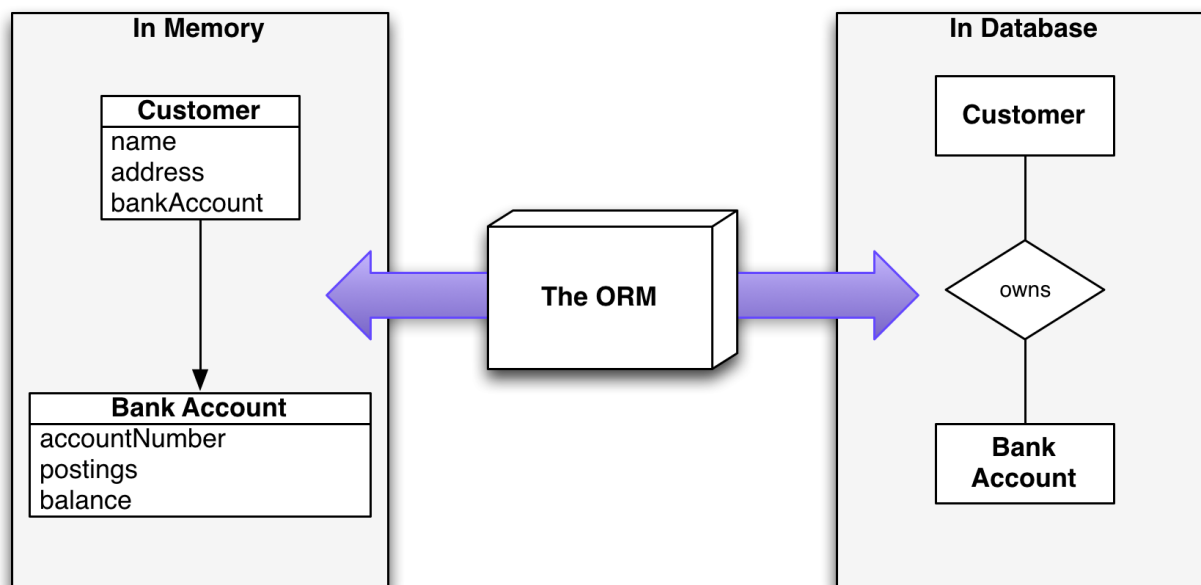
Flask SQLAlchemy



NOWCODER.COM
牛客网-中国最大IT笔试/面试题库

ORM介绍

对象关系映射
(Object-Relational
Mapping) 提供了概
念性的、易于理解的模
型化数据的方法



数据库配置/切换（代码演示）

对象关系映射（Object-Relational Mapping）提供了概念性的、易于理解的模型化数据的方法

```
SQLALCHEMY_DATABASE_URI = 'mysql://root:nowcoder@localhost:3306/test'
```

```
#SQLALCHEMY_DATABASE_URI = 'sqlite:///../nowstagram.db'
```

```
SQLALCHEMY_TRACK_MODIFICATIONS = True
```

```
SQLALCHEMY_ECHO = False
```

```
SQLALCHEMY_NATIVE_UNICODE = True
```

```
SQLALCHEMY_RECORD_QUERIES = False
```

dialect+driver://username:password@host:port/database

<http://flask-sqlalchemy.pocoo.org/2.1/config/#connection-uri-format>



数据模型（代码演示）

```
class User(db.Model):
```

```
    __tablename__ = 'user' # 指定表名字
```

```
    __table_args__ = {'mysql_collate': 'utf8_general_ci'}
```

```
    id = db.Column(db.Integer, primary_key=True, autoincrement=True)
```

```
    username = db.Column(db.String(80), unique=True)
```

```
    password = db.Column(db.String(32))
```

```
    def __init__(self, username, password):
```

```
        self.username = username
```

```
        self.password = password
```

```
    def __repr__(self):
```

```
        return ('<User %d %s>' % (self.id, self.username)).encode('gbk')
```

数据类型

Integer

String (size)

Text

DateTime

Float

Boolean

PickleType

LargeBinary



插入数据（代码演示）

```
for i in range(0, 100):
```

```
    db.session.add(User('牛客' +str(i), 'a'+str(i)))
```

```
    for j in range(0, 3): #每人发三张图
```

```
        db.session.add(Image(get_image_url(), i + 1))
```

```
        for k in range(0, 3):
```

```
            db.session.add(Comment('这是一条评论'+str(k), 1+3*i+j, i+1))
```

```
db.session.commit()
```



一对多和查询数据（代码演示）

一对多

```
images = db.relationship('Image', backref='user', lazy='dynamic')
```

<http://flask-sqlalchemy.pocoo.org/2.1/models/#one-to-many-relationships>

查询函数	
all()	filter_by(id>2)
order_by(db.desc(User.id))	get(primary_key)
limit(count)	first_or_404()
offset(count)	first()
paginate	get_or_404

```
User.query.paginate(page=1, per_page=10).items
```

<http://flask-sqlalchemy.pocoo.org/2.1/api/#flask.ext.sqlalchemy.BaseQuery>



删除数据（代码演示）

- `query.delete()`

```
Comment.query.filter_by(id = i + 1).delete()
```

- `db.session.delete()`

```
for i in range(51, 100, 2):  
    comment = Comment.query.get(i + 1)  
    db.session.delete(comment)
```



更新数据（代码演示）

- `query.update()`

```
User.query.filter_by(id>2).update({ 'username' : '新名字' +str(i)})
```

- `db.session.commit()`

```
for i in range(1, 100, 2):
```

```
    # 通过设置属性
```

```
    u = User.query.get(i + 1)
```

```
    u.username = 'd' + str(i*i)
```



模板继承（代码演示）

base.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  {% block head %}
  <link rel="stylesheet" href="style.css" />
  <title>{% block title %}{% endblock %} - My Webpage</title>
  {% endblock %}
</head>
<body>
  <div id="content">{% block content %}{% endblock %}</div>
  <div id="footer">
    {% block footer %}
    &copy; Copyright 2008 by <a href="http://domain.invalid/">you</a>.
    {% endblock %}
  </div>
</body>
</html>
```

child.html

```
{% extends "base.html" %}
{% block title %}Index{% endblock %}
{% block head %}
  {{ super() }}
  <style type="text/css">
    .important { color: #336699; }
  </style>
{% endblock %}
{% block content %}
  <h1>Index</h1>
  <p class="important">
    Welcome to my awesome homepage.
  </p>
{% endblock %}
```



首页/个人页/详情页开发（代码演示）



课后作业

- a) MySQL安装
- b) Jinja2模板语言复习
- c) 首页、个人页、详情页开发
- d) Flask-SQLAlchemy 一对多、多对多练习
- e) Flask-SQLAlchemy增删改查练习



Thanks