064 使用Flask\_SQLAlchemy创建数据表

* 核心技术

本实例中要实现创建MySQL数据表的功能，所以需要安装使用PyMySQL驱动。安装命令如下：

pip install pymysql

此外，Flask\_SQLAlchemy 不是Python的内置模块，需要安装后才能使用。安装命令如下：

pip install flask\_sqlalchemy

Flask\_SQLAlchemy的使用方法如下：

* 配置数据库连接

配置数据库连接，代码如下：

1. app = Flask(\_\_name\_\_)
2. # 基本配置
3. app.config['SQLALCHEMY\_TRACK\_MODIFICATIONS'] = **True**
4. app.config['SQLALCHEMY\_DATABASE\_URI'] = (
5. 'mysql+pymysql://root:root@localhost/flask'
6. )

上述代码中，第一个root是数据库用户名，第二个root是数据库密码，最后面的flask是数据库名称。

* 定义模型

模型一般是一个Python类，类中的属性对应数据库表中的列。Flask-SQLAlchemy创建的数据库实例为模型提供了一个基类以及一系列辅助类和辅助函数，可用于定义模型的结构。常用的SQLAlchemy字段类型如表1所示。

表1 常用的SQLAlchemy字段类型

|  |  |  |
| --- | --- | --- |
| 类型名 | python中类型 | 说明 |
| Integer | int | 普通整数，一般是32位 |
| SmallInteger | int | 取值范围小的整数，一般是16位 |
| BigInteger | int或long | 不限制精度的整数 |
| Float | float | 浮点数 |
| Numeric | decimal.Decimal | 普通整数，一般是32位 |
| String | str | 变长字符串 |
| Text | str | 变长字符串，对较长或不限长度的字符串做了优化 |
| Unicode | unicode | 变长Unicode字符串 |
| UnicodeText | unicode | 变长Unicode字符串，对较长或不限长度的字符串做了优化 |
| Boolean | bool | 布尔值 |
| Date | datetime.date | 时间 |
| Time | datetime.datetime | 日期和时间 |
| LargeBinary | str | 二进制文件 |

* 操作数据表

操作数据表的命令主要有2个：

* 创建表：db.create\_all()
* 删除表：db.drop\_all()