|  |  |  |
| --- | --- | --- |
| 语句 | 说明 | 截图 |
| >>> db.create\_all() | 创建表  （因为echo了可以看到建表语句） |  |
| >>> user=User(username='fake\_name')  >>> db.session.add(user)  >>>db.session.commit() | 新增数据  (创建，调用add,提交) |  |
| >>>users=User.query.all() | 获取所有行 |  |
| >>>users=User.query.limit(5).all()  >>>users | 返回限定的行数limit(N)  输入users打印 |  |
| >>> users=User.query.order\_by(User.username).all()  >>>users | 正向排序(这里根据username) |  |
| >>> users=User.query.order\_by(User.username.desc()).all()  >>>users | 逆向排序 |  |
| >>> user=User.query.first()  >>> user.username | 返回第一行 |  |
| >>> user=User.query.get(1)  >>> user.username | 获取链式数据的第几个数据  这里可以发现是通过id检索返回的，数值从1开始 |  |
| >>> users=User.query.order\_by(User.username.desc()).limit(5).first()  >>> users | 整合查找（高阶） |  |
| >>> page=User.query.paginate(1,5) | 分页显示  Paginate（x，y）  X:第一页  Y:几行数据  🡺分页操作 |  |
| >>> page.items | 返回这一页中数据对象 |  |
| >>>page.page | 返回第几页 |  |
| >>>page.pages | 返回总页数 |  |
| >>> page.has\_prev,page.has\_next | 有无前后页 |  |
| >>>>>> page.prev(),page.next(0) | 返回前后页对象 |  |
|  |  |  |