特点和核心: 1、轻量、简洁、扩展性强 Flask框架的基本认识 2、werkzeug和Jinja2 各个项目有独立的运行空间, 彼此互 不影响,python解释器彼此互不影 响; mkvirtualenv -p python3 环境的配置和安装 rmvirtualenv workon deactivate pip freeze > requirements.txt 依赖包 pip install -r requirements.txt __name__参数的作用是确定程序所 在的位置; 装饰器路由的实现: Rule, Map, MapAdapter 调试模式: 动态加载代码, 不用重启 服务器,会调试错误信息;在生产模 基本程序的实现 式下不能开启 加载配置文件: app.config.from_object(配置对象) app.config.from_pyfile(初始化文件) app.config.from_envvar(环境变量) 视图函数不允许重名,url可以重 复; redirect函数接收的参数为固定的 url,不建议直接使用,扩展性不 重定向 强,需要配合url_for实现重定向,接 收的参数为函数名,相对于url,函 数名更不容易发生变化 return后面可以返回自定义的不符合 http协议的状态码,实现前后端的数 据交互,也可以返回符合http协议的 状态码 状态码,相当于修改了框架封装好的 默认响应报文中的状态码 abort函数接收的参数为符合http协 议的状态码,作用配合errorhandler 异常处理 修饰的函数必须传入参数,参数为错 误信息, 实现自定义的错误页面 基于键值对的字符串,用来实现数据 的传输 使用json模块中的dumps(把字典转 **JSON** 成json), loads(把json转字典) 建议使用Flask封装的jsonify方法, 不仅可以返回json数据,可以指定响 应的数据类型 <args>固定的参数,必须传给视图 函数, 转换器限制参数的数据类型, 内置6种转换器,自定义转换器可以 限制参数的长度(了解) 查询字符串: url中? 后面、=传值、 &号分隔; request.args.get(等号前面的key) form: request.form.get(表单中字 段的key) files: request.files.get(表单中的 key文件的名称) cookies: request.cookies.get(cookie的key) postman工具的使用; 两种请求前执行 (before_first_request在第一次请求 前执行, before_request在每次请求 前执行),两种请求后执行 请求钩子 (after_request必须传入response响 应对象,并且返回response, teardown_request必须传入错误信 息) 字典、键值对的字符串,实现会话保 持,可以设置过期时间,保存在浏览 cookie 器中,不安全 保存在服务器中,把session_id存入 状态保持 session 浏览器中,相对安全; http协议底层是TCP/IP协议,是无 状态的,每次连接都要三次握手、四 次挥手; 封装了客户端和服务器交互过程中的 请求上下文 信息, session表示用户信息, request表示请求的参数信息 封装了程序运行过程中的一些配置信 息,比如调用的函数、模块、加载的 应用上下文 工具类、文件等, current_app生命 周期最长, 用来记录项目日志; g对 象可以在请求过程用来临时存储数据 在终端以命令的形式运行项目, 动态 的指定ip和port,后面配合flask-Flask_script

migrate扩展实现数据库表的创建(迁

移)

视图