

# Python第三轮考核

---

- 想说的话
- 知识点
- 推荐教程
- 初学者作业
- 有基础者作业
- 人工智能作业
- 作业要求
- 预习下一轮
- 考核截止日期
- 提交方式

## 想说的话

---

1. Q: 说好的分流在哪里?

A: 临近期末, 大家都很忙, 组织面试也十分困难, 所以拖到下一轮再分流和面试, 本轮任务也很简单。

2. Q: 我想学人工智能, 为什么要学后端方面的知识?

A: 人工智能不是单打独斗, 也需要合作应用, 很多时候训练好的模型在别人调用时, 需要一些简单的接口, 因此需要学习少量的后端知识, 从而知道怎么发送接口, 但是无需深入了解。

3. Q: Python后端有前途吗?

A: Python后端在几年前还是很热门的, 但是现在已经日渐式微, 在求职方面比不过Java和Go, 但由于搭建速度快还是有些中小公司在产品初期使用, 因此如果有想深入后端的, 可以去多看看Java和Go, 或者转前端测试。但是如果你只想做一个人的小项目的话, Python开发速度快, 工具多, 是不错的选择。

4. Q: 为什么学习Flask或者FastApi而不是更强的Django?

A: 现在前后端分离是主流, 前端通过接口和后端通讯, 而不是后端直接渲染, 强调低耦合。而Django在前后端分离的项目上, 对比Flask以及FastApi没有优势, 反而显得太重了。因此选择Flask或者FastApi进行学习

5. Q: 人工智能面试会考什么?

A: 在本轮, 我们会要求有想要参加人工智能的同学写一些通过之前的资料学习的感悟和思考, 然后根据这些感悟有针对性的提问, 检验大家的学习成果。当然搞人工智能需要你有一个好一点的显卡(如果你能搞到室友的显卡也不是不可以)。

## 知识点

---

1. 注册GitHub账号, 学习git的使用
2. HTTP请求, URL, 请求头, 请求体, 状态码与REST风格API
3. flask框架/fastapi框架的使用
4. 接口测试工具的使用(apifox(推荐), apipost, postman)
5. 数据库的增删改查
6. 接口文档编写(接口测试工具的文档功能/swagger)
7. 简单网页的制作(可选)

## 推荐教程

1. git和GitHub的使用 [https://blog.csdn.net/weixin\\_53315561/article/details/126802065](https://blog.csdn.net/weixin_53315561/article/details/126802065)  
(pycharm的git其实也很方便，可以了解一下)
2. flask文档 <https://dormousehole.readthedocs.io/en/latest/>
3. flask视频教程 <https://www.bilibili.com/video/BV1v7411M7us> (为什么推荐这个教程，因为这个教程只讲接口，不用了解模板和渲染的纯后台，更适合现在前后端分离趋势)
4. fastapi文档 <https://fastapi.tiangolo.com/zh/> (fastapi的文档已经写的很完善了，一步步入手，可以先看文档，不会再去搜)
5. 接口测试工具
  - apifox (<https://www.apifox.cn/>)
  - apipost (<https://www.apipost.cn/>)
  - postman (<https://www.postman.com/>)
6. REST风格api (<https://juejin.cn/post/7025222833798119454>) 状态码请按照文中的反例来写，状态码全传200，通过返回体中的code字段进行区分请求是否成功，同时返回msg，成功返回data。至于为什么状态码全传200可以看知乎

```
1 HTTP/1.1 200 ok
2 Content-Type: application/json
3 Server: example.com
4 #失败
5 {
6     "code": 404,
7     "msg": "该活动不存在",
8 }
9 #成功
10 {
11     "code": 200,
12     "msg": "success",
13     "data": {
14         ...
15     }
16 }
```

7. 多种传参方式，Query传参和Body传参，具体请看相应框架的文档
8. GitHub是一个很好的学习平台，可以看别人的代码的架构了解自己的不足
9. pycharm可以在创建项目时选择Flask项目和FastApi项目，让项目快速启动起来，可以尝试

## 初学者作业

编写一个TODO LIST，使用flask框架/fastapi框架完成以下 API，执行操作数据表的操作，并编写接口文档！

增:

- 添加一条新的待办事项

改:

- 将一条/所有待办事项设置为已完成

- 将 一条/所有已完成事项设置为待办

查:

- 查看 所有已完成/所有待办/所有事项 （需分页）
- 输入关键字查询事项 （需分页）
- 通过id查询事项

删:

- 删除 一条/所有已完成/所有待办/所有事项

#### 要求数据表数据与数据类型

id	主键,int
标题	varchar
内容	varchar/text
完成状态	varchar
添加时间	int/timestamp
截止时间	int/timestamp

要求:

1. 以上的改，查，删的“一条”请都通过id实现
2. 设置合理的路径，能从路径看出所实现的功能
3. 接口尽量满足 RESTful API 规范
4. 返回的接口格式请和上面所说的类似，返回JSON格式数据

## 有基础者作业

在完成初学者作业的基础上有选择的加以下功能

- 数据表再多一个type类型，区分事项的紧急度，类型为枚举类型
- 使用ORM
- 跨域功能
- 使用Redis缓存
- 做一个简单的前端todolist页面，不要求美观，只要求展现数据和进行增删改查功能，要求使用ajax通讯。可以使用jQuery和bootstrap(上面三个都是很多年前的技术，想搞全栈的直接去学前端框架)
- 编写单元测试

## 人工智能作业

1. 完成初学者作业（查询可以不分页）
2. 结合之前学习的和人工智能有关的内容，根据之前的学习情况梳理一份学习笔记，用Markdown格式写，最好写的通俗易懂，加上自己的理解，字数不限，越多越好，并上传。

## 作业要求

1. 不要抄袭哦
2. 将作业提交到GitHub上，提交push到的github的网址

## 考核截止日期

---

2023/1/12

## 提交方式

---

只需要传GitHub地址，选人工智能的还要传学习笔记。

上传地址 <https://www.wjx.cn/vm/O5mXWjX.aspx#>