

## 第 2 天的课后作业

### 基本概念（必做）

1. 解释 Python 脚本程序的"`_name_`"的作用
2. 请简述 Python 解释器搜索模块位置的顺序。
3. 请简述模块的概念。
4. 请简述导入模块的方法。导入模块的方式有哪几种，官方不推荐哪种？
5. 如何让你写的模块可以被系统上任何一个 `py` 文件导入。

### 编程操作 1（必做）

1. 继续完成进击的小鸟游戏，需求描述：
  - 提交 2 个文件夹，1 个是源程序文件夹，1 个打包发布后的程序文件夹
  - 源程序文件夹要求：
    1. 图片、音频、程序要分开
    2. 鼓励大家根据自己的喜好来修改游戏规则、以及相关图片和音频。
    3. 添加说明文档（`readme.md`）
  - 打包发布的要求：
    1. 使用 `pyinstaller` 完成打包，并提交
    2. 请同学们提前做好测试，确保 `exe` 能独立运行。

打包发布的参考文档：

<https://blog.csdn.net/chenzhanhai/article/details/85334919>

或者

<https://blog.csdn.net/fei347795790/article/details/88760265>

### 编程操作 2（必做）

写一个用户注册和登录验证程序，文件名 `account.json`，内容如下

```
{“expire_date”: “2021-01-01”, “status”: 0, “password”: “abc”}
```

■ 用户注册：

1. 输入用户名称（需要检查用户名称是否已经被注册）
2. 两次输入用户密码（用户密码需要至少 6 位，至少包含数字字母和下划线），检查两次输入的用户密码是否不一致。
3. 输入用户有效期。
4. 结果保存到 `account.json`，`account` 是 用户名称。

■ 用户登录：

1. 根据用户输入的用户名&密码，找到对应的 `json` 文件，把数据加载出来进行验证
2. 用户名为 `json` 文件名，密码为 `password` 所对应的值。
3. 判断账号是否过期，获取当前日期与 `expire_date` 进行对比。
4. 不要明文存储密码，要采用 `hash` 来加密
5. 登陆成功后，打印“登陆成功”，三次登陆失败，`status` 值改为 1，并且锁定账号。

## 编程操作 3（选做）

### 购物车

- 启动程序后，分别从文件加载商品(`goods.txt`)和用户信息(`user.txt`)
- 输入用户名密码后，检查用户密码是否正确，不正确重复登录，3 次失败，则退出程序。
- 登录后，用户选择以下操作：（查看商品 (g)、查看余额 (u)、购买商品(b)、退出购买(q)）
  1. 输入 q，则退出。
  2. 输入 g，则查看所有商品信息。
  3. 输入 u，则查看用户余额信息。
  4. 输入 b，则进入购买流程：
  5. 允许用户根据商品编号购买商品。
  6. 用户选择商品后，检测余额是否够，够就直接扣款，不够就提醒。
  7. 可随时退出，退出后，打印已购买商品和余额，并更新文件信息。

```
goods = [{"name": "电脑", "price": 1999}, {"name": "鼠标", "price": 10}, {"name": "书籍",  
"price": 50}, {"name": "空调", "price": 2000},]
```

```
users = [ {"name": "张三", "password": 'zhangsan', 'age':11,'balance': 1000.0}, {"name": "李四",  
"password": 'lisi', 'age':22,'balance': 2000.0}, {"name": "王五", "password": 'wangwu',  
'age':33,'balance': 2000.0},]
```