AIDVN2202班级训练营-Day05

1、函数-function

- 概念：对一段代码的封装。
- 功能：实现代码的复用。
- 语法：

```
def 函数名([形式参数]):
    ''' 文档字符串 '''
    函数体
    [return 数据]

函数名([实际参数])
```

- 说明：
  - 1、函数定义后只有调用才能执行。
  - 2、如果在一个函数中使用另一个函数的数据，则需要函数将数据返回。

2、文件-file

- 概念：存储数据的一种方式。
- 语法：

```
# 1 打开文件
file_object = open(file, mode='r', encoding='utf-8')

# 2 文件操作
# 读 mode='r'(默认)
data = file_object.read()

# 写 mode='w' - 覆盖
file_object.write(str_data)

# 追加 mode='a'
file_object.write(str_data)

# 3 关闭文件
file_object.close()
```

1、Python网络爬虫

- 技术：用于【**获取数据**】的技术。
- 概念：按照一定规则自动获取互联网上的页面中信息数据的【程序或脚本】。
- 分类：
  - 通用爬虫：搜索引擎使用，遵守robots协议

    - **聚焦爬虫**：爬取指定页面。
- 爬虫流程
    - 1、确定爬取的URL(唯一确定爬取页面)
    - 2、发送请求，获取响应(得到页面源代码)
    - 3、数据清洗（获取想要的数据）
    - 4、数据存储（文件(csv/txt/excel/word)、数据库）
    - 5、数据预处理（异常值、数据标准化）
    - 6、数据分析与挖掘
    - 7、数据可视化（Excel、PowerBI）
    - 8、数据分析报告（数据建模、实施落地）
- 爬虫：
    - 1、url：http://www.baidu.com
    - 2、发送请求、获取响应数据
        - 请求模块：requests

```
import requets

# 发送请求
response = requests.get(url)

# 响应页面数据
data = response.text    # 文本数据
data = response.content    # 二进制数据(视频、音频、压缩包)

# 页面中的数据
print(data) 10:47 回来
```

    - 3、数据解析
        - 模块：lxml

```
from lxml import etree

etree_object = etree.HTML(爬去的页面数据)

etree_object.xpath('xpath表达式')
```

        - **xpath表达式**
            - \\ : 从所有节点（<标签>）中查找（子节点或后代节点）
            - .: 表示当前节点
            - \: 表示子节点
            - @: 获取页面节点的属性值、定位节点
            - text(): 获取节点中的文本数据