# 第四阶段 Python数据分析

# Python 数据分析库4

## 网校相关视频

## 明确本次课知识点，明确重点难点

### 【知识点目标】

* Pandas中的运算
* 数据规整化

### 【重点】

* Pandas中的运算
* 数据规整化

### 【难点】

* Pandas中的运算
* 数据规整化

## 复习巩固作业讲解

## 本次课程任务讲解

### 【知识点1】Pandas中的运算

1. Pandas中的运算
   1. 简单运算

在分析数据时，不可避免地要对数据进行运算。当对两个数据集进行算术运算时，遇到的核心问题有两个：一是两个数据集之间如何进行匹配进而运算；二是如何处理不匹配的数据。

* + 1. Series之间的运算
    2. Series与DataFrame之间的运算
    3. DataFrame之间的运算

Pandas的两种数据类型(Series与DataFrame)比较有特色的部分是index (DataFrame 还多了列名columns),因此两个数据集最直接的匹配方式是按照index与columns进行匹 配运算。如果两个数据集的加减乘除运算分别用“ + ”、"\*”、“/”来实现，那么Series 与Series之间是index匹配运算，无法匹配的则用NaN进行填充；Series与DataFrame之 间是Series的index与DataFrame的columns之间匹配,运算的方式是将DataFrame每一 行都与Series进行匹配运算，无法匹配的用NaN填充；DataFrame与DataFrame之间是 同时对index与columns进行匹配，匹配成功的元素进行运算，无法匹配的用NaN填充。

* 1. 函数运用和映射

DataFrame是二维的数据集，有时我们需要对数据集的行与列进行函数操作，虽然可以逐行或者逐列地对该数据集进行分析，但 Pandas包里有更为简便的解决方式一一调用apply()函数。该函数基本形式为：

*DataFrame.apply(func, axis=0)*

其中func是要对每行或者每列应用的函数，axis=0为默认值，是指将func应用在 DataFrame的每个列上，当axis=l时，则应用在行上。

### 【知识点2】数据规整化

1. 缺失值的预处理
   1. 缺失值的判断

isnull()

notnull()

返回布尔值的数据集

* 1. 选出非缺失值

通过选择列中的非空索引来选出非缺失值

1. 缺失值的填充

*DataFrame.fillna(value=None, method=None, axis=None, inplace=False, limit=None)*

1. 缺失值的选择删除（删除缺失值所在的行/列）

*DataFrame.dropna(axis=0, how=’any’, thresh=None)*

1. 删除重复数据

*DataFrame.duplicated()*

*DataFrame.drop\_duplicates()*

1. 层次化索引

层次化索引是 pandas的一项重要功能，即在DataFrame的一个轴上拥有多个(两个以上)索引级别。抽象点说，我们可以用低维度形式处理高维度数据。

1. stack

stack:将数据的列“旋转”为行

1. unstack

unstack:将数据的行“旋转”为列

## 自主学习作业讲解

无

## 课程总结

#### Serires基本操作

#### DataFrame对象基本操作

#### 数据规整化

## 下次自主学习任务布置