# Python数据处理 教案

周次：9

课次：1

章节内容：Pandas数据分析

课时：2

## 单元教学目标

- 知识目标：掌握Pandas DataFrame的基本结构和核心数据类型，能够正确识别数值型、分类型等数据类型。   
- 技能目标：能够使用Pandas完成数据的加载、筛选、合并与初步统计分析，如使用loc/iloc索引器、条件筛选及基本统计函数。   
- 素养目标：养成数据清洗的规范意识，能识别并处理缺失值、重复数据，遵守数据处理的伦理规范。

## 教学重点

• 数据框（DataFrame）是核心数据结构，用于存储和操作表格型数据   
• 数据筛选与过滤（如布尔索引、loc/iloc定位、条件过滤）是基础技能   
• 分组聚合操作（如groupby分组、agg聚合函数）是数据分析的核心能力

## 教学难点

• \*\*DataFrame的索引与列操作难点\*\*：学生可能难以理解索引（index）和列（columns）的层级关系，例如如何通过`df.loc[]`或`df.iloc[]`精准选取数据，或如何通过`df.set\_index()`调整索引结构。部分学生容易混淆`loc`与`iloc`的使用场景，导致数据选取逻辑错误。   
  
• \*\*数据合并与重塑的逻辑混淆\*\*：在合并数据（如`pd.merge()`）或重塑数据（如`pd.pivot\_table()`）时，学生可能因对多索引、条件筛选（如`df[df['column'] > 5]`）或合并后的数据结构不清晰，导致操作失误。例如，无法正确使用`merge`函数的`on`或`how`参数，或误判合并后的数据类型。

## 教学活动

新课导入【10分钟】   
教师用真实案例引入：展示某电商平台销售数据，提问“如何快速筛选出Q4的销售数据？”，引发学生兴趣。通过故事导入，结合生活场景激发学习动机，引导学生思考数据处理需求。   
  
讲授新课【80分钟】   
1. \*\*讲授法\*\*（10分钟）   
 - 解释Pandas数据框（DataFrame）的概念，结合案例演示数据框的创建（如`pd.DataFrame({ 'name': ['Alice', 'Bob'], 'age': [25, 30] })`）。   
 - 重点讲解数据框的行、列索引及数据类型。   
  
2. \*\*案例分析法\*\*（15分钟）   
 - 分组分析“销售数据筛选”案例：   
 - 用`loc`筛选Q4数据（`df.loc[df['sale\_date'].str.contains('Q4')`]）。   
 - 用`iloc`筛选特定行（如`df.iloc[1:3]`）。   
 - 学生动手操作，教师巡回指导。   
  
3. \*\*讨论法\*\*（10分钟）   
 - 讨论问题：   
 - 为何`df.loc`比`df['column'].str`更灵活？   
 - 如何避免数据类型错误（如`astype(float)`）。   
  
4. \*\*练习法\*\*（15分钟）   
 - 基础练习：   
 - 用`df.head()`查看前5行数据。   
 - 用`df.tail()`查看后5行数据。   
 - 进阶练习：   
 - 合并两个数据框（`pd.merge(df1, df2, on='id')`）。   
 - 使用`df.sort\_values()`排序数据。   
  
巩固练习【20分钟】   
1. \*\*游戏法\*\*（5分钟）   
 - 设计“数据盲盒”游戏：学生分组抽取数据框，用`loc`筛选指定条件，限时完成任务。   
  
2. \*\*练习法\*\*（10分钟）   
 - 基础练习：   
 - 用`df.columns`查看列名。   
 - 用`df.index`查看行索引。   
 - 进阶练习：   
 - 用`df.loc[df['region']=='Asia']`筛选亚洲数据。   
 - 用`df.groupby('category').sum()`统计分类数据。   
  
归纳总结【10分钟】   
教师总结关键点：   
- 数据框的核心操作（创建、筛选、合并、排序）。   
- 强调`loc`与`iloc`的区别及适用场景。   
- 提醒学生注意数据类型转换（如`astype(int)`）。   
- 布置课后任务：分析学生日常数据（如学生成绩表）并用Pandas处理。

## 教学资源

• 教学设备：投影仪/白板（用于展示图表/代码），学生计算机（需安装Python和Pandas库），网络设备（确保学生可访问在线资源），打印机（用于打印教学资料）   
• 工具推荐：Python编程环境（建议安装Jupyter Notebook或PyCharm），Pandas库（需安装至学生电脑），数据可视化工具（如Matplotlib/Seaborn）   
• 参考资料：   
- 书籍：《Python数据科学手册》（第3版）、《Pandas Cookbook》   
- 在线资源：   
 - Pandas官方文档（https://pandas.pydata.org/）   
 - Kaggle学习路径（https://www.kaggle.com/learn/python）   
 - Coursera《Python for Data Analysis》课程（https://www.coursera.org/）   
- 练习平台：Exercism（https://exercism.org/）、Codecademy（https://www.codecademy.com/）   
- GitHub仓库：Pandas官方项目（https://github.com/pandas-dev/pandas）   
• 实践项目建议：   
 - 使用Kaggle数据集练习数据清洗与分析   
 - 通过Jupyter Notebook完成从数据导入到可视化的一整套流程   
 - 模拟真实场景（如电商销售数据的汇总分析）

## 教学反思

• 教学效果方面：需评估学生是否能独立完成数据读取、清洗、筛选等基础操作，是否掌握Pandas核心数据结构（如DataFrame、Series）的使用，以及是否理解数据处理流程的逻辑顺序。   
• 学生反馈方面：关注学生对Pandas功能的实用性评价（如“案例贴近实际”“操作步骤清晰”），同时留意对复杂操作（如多级索引、数据合并）的掌握程度及反馈意愿。   
• 改进建议：   
• 增加实战案例：通过真实数据集（如电商销售数据）引导学生完成完整数据处理流程，强化应用能力。   
• 分步骤教学：将复杂操作拆解为阶段性任务（如先清洗再分析），降低学习门槛。   
• 强化互动：通过提问、小组讨论或代码调试环节，提升学生参与度并及时发现问题。   
• 增设练习时间：预留课堂时间让学生实践操作，结合即时反馈（如代码错误提示）巩固知识。   
• 提供资源支持：补充Pandas进阶技巧（如数据可视化）及常见问题解答文档，辅助学生自主学习。   
• 优化教学节奏：根据学生反馈调整课时分配，确保核心知识点（如数据类型转换）与进阶内容（如性能优化）合理平衡。

## 教学评价

• 过程性评价   
• 课堂参与度：观察学生在实验课中的发言频率、问题解答积极性及小组协作表现，满分10分。   
• 作业完成情况：检查学生是否按时提交作业，作业内容是否符合要求（如代码规范、注释完整性），满分10分。   
• 实验操作能力：评估学生在Pandas基础操作（如数据读取、清洗、筛选）中的动手能力与问题解决能力，满分10分。   
• 小组合作质量：观察小组成员分工是否明确、沟通是否顺畅，是否完成团队任务，满分10分。   
  
• 结果性评价   
• 作业/项目评分：根据代码正确性、逻辑性、文档完整性等综合评分，满分20分。   
• 任务完成度：评估学生是否完整实现Pandas数据分析目标（如数据可视化、统计分析），满分20分。   
• 创新性：考察学生是否在数据处理中引入新方法（如自定义函数、数据合并）或优化流程，满分10分。   
• 项目展示表现：通过答辩或演示评估学生对知识点的掌握程度与表达能力，满分10分。   
  
• 评价方式   
• 过程性评价：课堂观察记录、实验日志、小组讨论反馈、作业提交情况。   
• 结果性评价：作业评分表、项目代码审查、任务完成度核查、答辩评分表。