

Лабораторная работа № 8

«Работа с Pandas»

Подготовка к л.р.

Как создать CSV-файлы:

1. Откройте текстовый редактор (например: Notepad на Windows или TextEdit на macOS).
2. Скопируйте данные из каждого блока и вставьте их в новый файл.
3. Сохраните файл с соответствующим именем и расширением .csv (например: sales_data.csv).
4. Убедитесь в том, что в процессе сохранения выбран формат "Все файлы" или "CSV".

Примеры:

```
Date,Product,Quantity,Price
2023-01-01,Apple,10,0.5
2023-01-01,Banana,5,0.3
2023-01-02,Orange,8,0.4
2023-01-02,Apple,15,0.5
2023-01-03,Banana,20,0.3
2023-01-03,Grapes,12,1.0
2023-01-04,Apple,7,0.5
2023-01-04,Banana,10,0.3
2023-01-05,Grapes,5,1.0
```

```
StudentID,Name,Major
1,Alice Smith,Computer Science
2,Bob Johnson,Business
3,Charlie Brown,Mathematics
4,Diana Prince,Computer Science
5,Ethan Hunt,Biology
```

```
StudentID,Course,Grade
1,Math,A
1,Physics,B+
2,Business,A-
2,Economics,B+
3,Math,C+
4,Chemistry,A
4,Biology,A-
5,Biology,B+
```

```
Date,Temperature,Humidity,Windspeed
2023-02-01,10.5,80,15
2023-02-02,12.0,75,10
2023-02-03,-2.0,90,20
2023-02-04,-1.5,85,25
2023-02-05,8.0,70,5
2023-02-06,-5.0,95,30
2023-02-07,-1.0,88,12
2023-02-08,-2.5,92,18
```

Задание 1

Вам дан CSV-файл `sales_data.csv`, содержащий данные о продажах в некотором магазине. Файл содержит следующие столбцы:

1. Date: Дата продажи (в формате YYYY-MM-DD).
2. Product: Название продукта.
3. Quantity: Количество проданных единиц товара.
4. Price: Цена за единицу товара.

Задачи:

1. Загрузите данные из файла `sales_data.csv` в `DataFrame` Pandas.
2. Вычислите общую выручку за каждую дату. Создайте новый столбец `Revenue`, который будет содержать выручку для каждой строки (`Quantity * Price`).
3. Найдите дату с наибольшей общей выручкой.
4. Определите топ-3 самых продаваемых продукта по количеству проданных единиц.
5. Выведите `DataFrame`, содержащий только те строки, где количество проданных единиц товара больше 10.

Подсказка: Используйте функции `pd.read_csv()`, `groupby()`, `sum()`, `sort_values()`, `head()` и операторы сравнения.

Задание 2

Вам даны два CSV-файла: `students.csv` и `grades.csv`.

`students.csv` содержит информацию о студентах:

1. StudentID: Уникальный идентификатор студента.
2. Name: Имя студента.
3. Major: Специальность студента.

`grades.csv` содержит информацию об оценках студентов:

1. StudentID: Уникальный идентификатор студента.
2. Course: Название курса.
3. Grade: Оценка студента по данному курсу.

Задачи:

1. Загрузите данные из файлов `students.csv` и `grades.csv` в `DataFrame` Pandas.
2. Объедините два `DataFrame` в один, используя `StudentID` в качестве ключа для объединения.
3. Вычислите средний балл для каждого студента.
4. Найдите студента с самым высоким средним баллом.
5. Выведите `DataFrame`, содержащий только тех студентов, которые учатся на специальности "Computer Science".

Подсказка: Используйте функции `pd.read_csv()`, `merge()`, `groupby()`, `mean()`, `sort_values()`, `head()` и операторы сравнения.

Задание 3

Вам дан CSV-файл `weather_data.csv`, содержащий данные о погоде в некотором городе. Файл содержит следующие столбцы:

1. Date: Дата (в формате YYYY-MM-DD).
2. Temperature: Температура в градусах Цельсия.
3. Humidity: Влажность (в процентах).
4. WindSpeed: Скорость ветра (в км/ч).

Задачи:

1. Загрузите данные из файла `weather_data.csv` в `DataFrame` Pandas.
2. Вычислите среднюю температуру, влажность и скорость ветра за весь период.
3. Найдите дату с самой высокой температурой.
4. Найдите даты, когда температура была выше средней, а влажность ниже средней.

Подсказка: Используйте функции `pd.read_csv()`, `mean()`, `max()`, `groupby()` и операторы сравнения.