Домашнее задание к семинару 02 (HW02)

Тема: работа с табличными данными в Pandas, контроль качества данных, базовый EDA и визуализация в Matplotlib.

HW02 относится к семинару **S02** и выполняется в личном репозитории студента, созданном на основе шаблона `aie-student-template`, в папке `homeworks/HW02/`.

---

1. Цель

Закрепить:

- базовые навыки работы с **табличными данными** в `pandas.DataFrame`;
- приёмы **контроля качества данных** (пропуски, дубликаты, подозрительные значения);
- минимальный **Exploratory Data Analysis (EDA)**: описательные статистики, группировки и агрегаты;
- построение простых, но осмысленных графиков в **Matplotlib** (histogram, boxplot, scatter) и их сохранение;
- первый шаг к переносу этих приёмов на данные своего будущего проекта (опционально).

---

2. Задание

2.1. Структура для HW02

1. В корне репозитория должна быть папка `homeworks/` (создать, если её ещё нет).
2. Внутри `homeworks/` создать папку `HW02/`.
3. В папке `homeworks/HW02/` создать основной ноутбук `HW02.ipynb`.
4. (Рекомендуется) Внутри `homeworks/HW02/` создать подпапку `figures/` для сохранения графиков:
    - `homeworks/HW02/figures/`.

> Если структура уже частично создана на семинаре, убедитесь, что имена папок и файлов совпадают с указанными (регистр букв важен).

2.2. Учебный датасет

1. Использовать **учебный табличный датасет**, предоставленный преподавателем:
    - файл и путь к нему будут указаны в материалах курса (например, `S02-hw-dataset` или аналогичный).
2. Скопировать или настроить путь к этому файлу так, чтобы ноутбук `HW02.ipynb` мог его корректно загрузить (относительный путь внутри вашего репозитория).

2.3. Содержание ноутбука `HW02.ipynb` (основная часть)

В ноутбуке `homeworks/HW02/HW02.ipynb` необходимо выполнить следующие шаги.

2.3.1. Загрузка данных и первичный осмотр

1. Импортировать необходимые библиотеки:
   - `pandas` (обязательно),
   - при необходимости `numpy`,
   - `matplotlib.pyplot` для визуализации.
2. Загрузить учебный датасет в `pandas.DataFrame` с помощью `pd.read_csv` (или другого подходящего метода).
3. Вывести:
   - первые строки датасета (`head()`),
   - информацию о столбцах и типах (`info()`),
   - базовые описательные статистики (`describe()` или аналог).

2.3.2. Пропуски, дубликаты и базовый контроль качества

1. Посчитать долю пропусков в каждом столбце (например, через `isna().mean()` или аналог).
2. Проверить наличие полностью дублирующих строк (через `duplicated()`).
3. Найти и вывести «подозрительные» случаи, исходя из смысла датасета. Примеры:
   - отрицательные значения в полях, где их не должно быть (количество, цена и т.п.);
   - нереалистичные значения (например, возраст > 100, нулевой доход при ненулевых количествах и т.п.);
   - другие логические противоречия, характерные для конкретного датасета.
4. Кратко (1-2 абзаца) описать текстом, какие проблемы качества данных были обнаружены.

2.3.3. Базовый EDA: группировки, агрегаты и частоты

1. Посчитать частоты для одной или двух категориальных переменных (например, `value_counts()` для столбца с категорией/страной/классом).
2. Выполнить хотя бы одну осмысленную группировку с агрегатами через `groupby`:
   - например, среднее и сумму по количественным признакам в разрезе категорий.
3. При необходимости ввести дополнительные «коридоры» (bins) или группировки (например, возрастные группы, диапазоны значений и т.п.).
4. Кратко (1-2 абзаца) описать текстом основные наблюдения:
   - какие категории доминируют,
   - как отличаются группы по средним значениям,
   - есть ли неожиданные эффекты.

2.4. Визуализация данных в Matplotlib

В том же ноутбуке `HW02.ipynb` нужно построить как минимум:

1. **Одну гистограмму** для количественного признака:
   - осмысленный выбор числа корзин (`bins`),
   - подписи осей и заголовок.
2. **Один боксплот (boxplot)** для количественного признака:

    - можно как общий, так и по группам (например, по категориям),
    - подписи оси и заголовок.

3. **Один scatter plot** (диаграмма рассеяния) для пары количественных признаков:

    - подписи обеих осей,
    - заголовок,
    - при желании можно добавить цвет/легенду для различения категорий.

4. Сохранить **минимум один** из построенных графиков в папку `homeworks/HW02/figures/`:

    - использовать `plt.savefig(...)` или аналог;
    - убедиться, что файл действительно появляется в репозитории и может быть открыт отдельно от ноутбука.

Желательно снабдить графики краткими текстовыми комментариями:

- что именно показано;
- какие выводы можно сделать.

2.5. Опциональная часть: мостик к проекту

Опциональная (но рекомендованная) часть для тех, кто уже определился с темой проекта и имеет или может сгенерировать данные.

1. В папке `project/` создать (если ещё нет) папку `notebooks/`.
2. Создать ноутбук `project/notebooks/eda_v1.ipynb`.
3. В этом ноутбуке:
    - загрузить небольшой фрагмент будущих проектных данных (или реалистичный синтетический пример, если «боевые» данные пока недоступны);
    - выполнить минимум 3 простых проверки качества:
        - пропуски,
        - дубликаты,
        - подозрительные значения;
    - сделать 2-3 базовых графика (hist/boxplot/scatter) для ключевых признаков;
    - добавить 3-5 коротких текстовых наблюдений о данных.

Эта часть даёт задел для будущей работы над проектом, но не является критически обязательной для зачёта HW02 (см. критерии).

---

3. Требования к структуре и именованию

Обязательная структура к дедлайну:

- в корне репозитория: папка `homeworks/`;
- внутри неё: папка `HW02/`;
- внутри папки `HW02/`:
    - основной ноутбук `HW02.ipynb`,
    - (рекомендуется) папка `figures/` с хотя бы одним сохранённым графиком.

Требования:

- названия папок и файлов должны быть **строго такими**, как указано выше (регистр букв имеет значение);
- код в `HW02.ipynb` должен использовать учебный датасет, указанный преподавателем;
- ноутбук должен корректно открываться и выполняться без ошибок при наличии стандартного окружения (`pandas`, `numpy`, `matplotlib`).

Опциональная часть:

- при наличии `project/notebooks/eda_v1.ipynb` он не заменяет `HW02.ipynb`, а дополняет его и учитывается как плюс при оценке.

---

4. Критерии зачёта

Домашнее задание считается зачтённым, если:

1. В публичном репозитории студента присутствует папка `homeworks/HW02/` с файлом `HW02.ipynb`.
2. Ноутбук `HW02.ipynb` содержит:
   - корректную загрузку учебного датасета в `pandas.DataFrame`;
   - базовый первичный осмотр (`head`, `info`, `describe` или аналог);
   - анализ пропусков и дубликатов;
   - минимум два примера проверки качества данных (диапазоны, логические противоречия и т.п.);
   - минимум одну осмысленную группировку с агрегатами (через `groupby` или аналог);
   - минимум три графика: histogram, boxplot и scatter plot (можно больше).
3. Хотя бы один график сохранён в файл в структуре репозитория (например, в `homeworks/HW02/figures/`), и этот файл присутствует в Git.
4. Код выполняется без ошибок при последовательном запуске всех ячеек ноутбука.

Дополнительно поощряется (но не обязательно для зачёта):

- аккуратные и понятные подписи осей, легенд и заголовков;
- наличие кратких текстовых комментариев и выводов по результатам EDA;
- осмысленные проверки качества данных, привязанные к предметной области датасета;
- наличие и заполненность ноутбука `project/notebooks/eda_v1.ipynb` с первым EDA по проекту.

---

5. Сроки и порядок сдачи

- Работа выполняется **индивидуально**.
- Дедлайн выполнения HW02 объявляется преподавателем отдельно (в чате/на портале курса).
- Фактом сдачи работы считается наличие к указанному дедлайну:
  - публичного репозитория студента;
  - структуры `homeworks/HW02/`;
  - файла `HW02.ipynb` с выполненными заданиями по учебному датасету (в актуальной версии ветки `main` или другой заранее оговорённой ветки).
- Наличие опционального ноутбука `project/notebooks/eda_v1.ipynb` учитывается как плюс, но не компенсирует отсутствие или критическую неполноту `HW02.ipynb`.