

## Задача: "Универсальный анализатор данных"

Вы работаете в компании, которая получает данные из разных источников в разных форматах: CSV, JSON и, возможно, в будущем появятся и другие. Вам нужно разработать утилиту, которая сможет автоматически определять тип файла и предоставлять по нему сводную информацию (например, количество записей, список колонок и первые несколько строк).

### Требования к программе:

1. **Гибкость:** Система должна быть легко расширяемой для поддержки новых форматов файлов в будущем (например, XML или YAML), не изменяя основной логики.
2. **Автоматизация:** Программа должна сама определять, какой обработчик использовать, основываясь на расширении файла.
3. **Надежность:** Перед обработкой файла необходимо проверять, существует ли он.
4. **Информативность:** Для каждого файла нужно выводить стандартизированный отчет.

### Описание решения

1. **Абстрактный базовый класс (ABC) `DataAnalyzer`:**
  - Это "чертеж" для всех обработчиков.
  - Он будет содержать абстрактный метод `read_data()`, который каждая дочерняя реализация (для CSV, JSON и т.д.) должна будет переопределить.
  - Также в нем будет общий метод `get_summary()`, который будет работать для любого типа данных, как только они будут прочитаны.
2. **Конкретные классы-наследники `CSVAnalyzer` и `JSONAnalyzer`:**
  - Эти классы будут наследоваться от `DataAnalyzer`.
  - Каждый из них реализует свою версию метода `read_data()`, используя библиотеку **Pandas** для чтения соответствующего формата файла.
3. **Статический метод `validate_path()`:**
  - Это будет вспомогательная функция, размещенная внутри `DataAnalyzer`. Она будет проверять, существует ли файл по

указанному пути. Этот метод не зависит от состояния конкретного объекта или класса

4. **Метод класса `create_analyzer()`:**

- Это будет метод в `DataAnalyzer`. Его задача - принять путь к файлу, посмотреть на его расширение (`.csv` или `.json`) и вернуть правильный объект-анализатор (`CSVAnalyzer` или `JSONAnalyzer`). Он оперирует самим классом для создания нужного экземпляра.