

Лабораторная работа 3. Обработка данных.

Вариант 1–2

1. Написать скрипт, который разобьёт исходный csv файл на файл X.csv и Y.csv, с одинаковым количеством строк. Первый будет содержать даты, второй - данные.
2. Написать скрипт, который разобьёт исходный csv файл на N файлов, где каждый отдельный файл будет соответствовать одному году. Файлы называются по первой и последней дате, которую они содержат. (если файл содержит данные с первого января 2001 по 31 декабря 2001, то файл назвать 20010101_20011231.csv) 0.
3. Написать скрипт, который разобьёт исходный csv файл на N файлов, где каждый отдельный файл будет соответствовать одной неделе. Файлы называются по первой и последней дате, которую они содержат. 0.
4. Написать скрипт, содержащий функцию, принимающую на вход дату (тип datetime) и возвращающий данные для этой даты (из файла) или None если данных для этой даты нет. Функция должна быть представлена в четырёх версиях в зависимости от типа входных файлов, из которых будут прочитаны данные (пункты 0–3).

Вариант 3–10

1. Написать скрипт для формирования текстового файла-аннотации собранного датасета. Файл-аннотация должен представлять собой csv-файл, в котором в первой колонке будет указан абсолютный путь к файлу, во второй колонке относительный путь относительно вашего Python-проекта, третья колонка будет содержать текстовое название класса (метку класса), к которому относится данный экземпляр.
2. Написать скрипт для копирования датасета в другую директорию таким образом, чтобы имена файлов содержали имя класса и его порядковый номер. То есть из dataset/class/0000.jpg должно получиться dataset/class_0000.jpg. Для того чтобы осталась возможность определить принадлежность экземпляра к классу создать файл-аннотацию (как в пункте 1).
3. Написать скрипт, создающий копию датасета таким образом, чтобы каждый файл из сходного датасета получил случайный номер от 0 до 10000, и датасет представлял собой следующую структуру dataset/номер.jpg.
4. Написать скрипт, содержащий функцию, получающую на входе метку класса и возвращающую следующий экземпляр (путь к нему) этого класса. Экземпляры идут в любом порядке, но не повторяются. Когда экземпляры заканчиваются, функция возвращает None.