





Сериализация

Семинар 8





Что будет сегодня на семинаре. Наши цели:

-  Разобраться в сериализации и десериализации данных
-  Изучить самый популярный формат сериализации - JSON
-  Познакомиться с чтением и записью таблиц в формате CSV
-  Разобраться с внутренним сериализатором Python - модулем pickle



Вопросы?

Вопросы?







Вопросы?



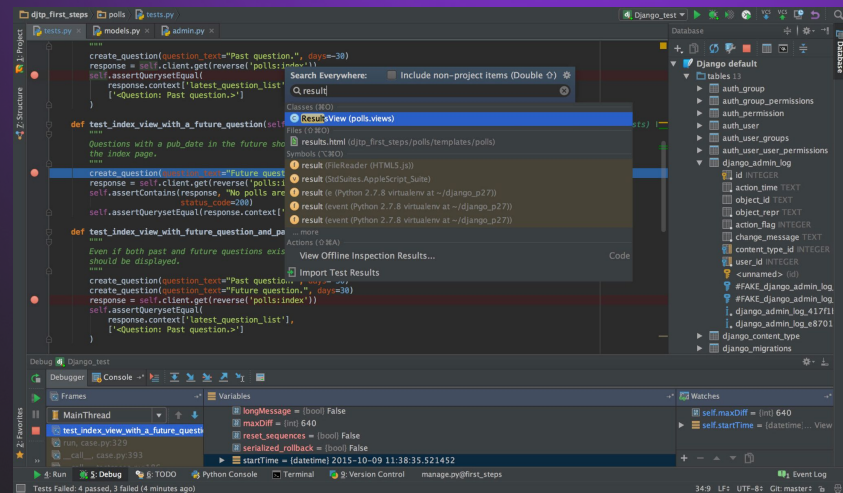


Задание №1

-  Вспоминаем задачу 3 из прошлого семинара. Мы сформировали текстовый файл с псевдо именами и произведением чисел.
-  Напишите функцию, которая создаёт из созданного ранее файла новый с данными в формате JSON.
-  Имена пишите с большой буквы.
-  Каждую пару сохраняйте с новой строки.









Решение в IDE



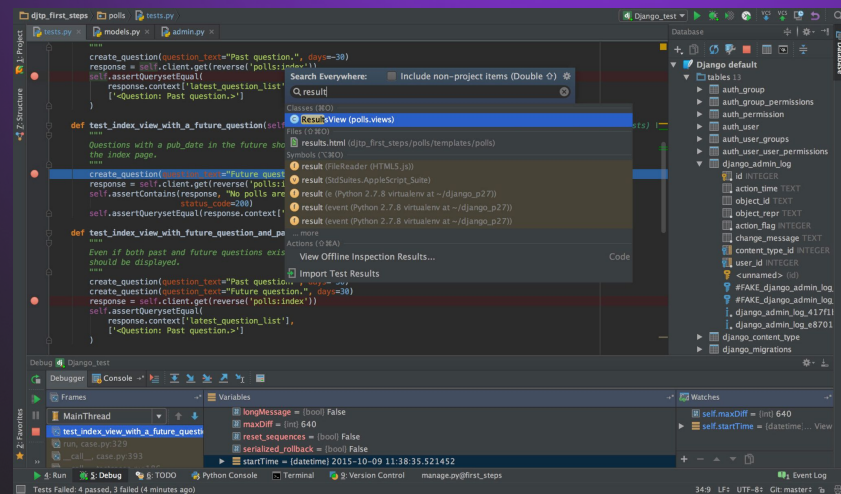


Задание №2

-  Напишите функцию, которая в бесконечном цикле запрашивает имя, личный идентификатор и уровень доступа (от 1 до 7).
-  После каждого ввода добавляйте новую информацию в JSON файл.
-  Пользователи группируются по уровню доступа.
-  Идентификатор пользователя выступает ключём для имени.
-  Убедитесь, что все идентификаторы уникальны независимо от уровня доступа.
-  При перезапуске функции уже записанные в файл данные должны сохраняться.



Решение в IDE





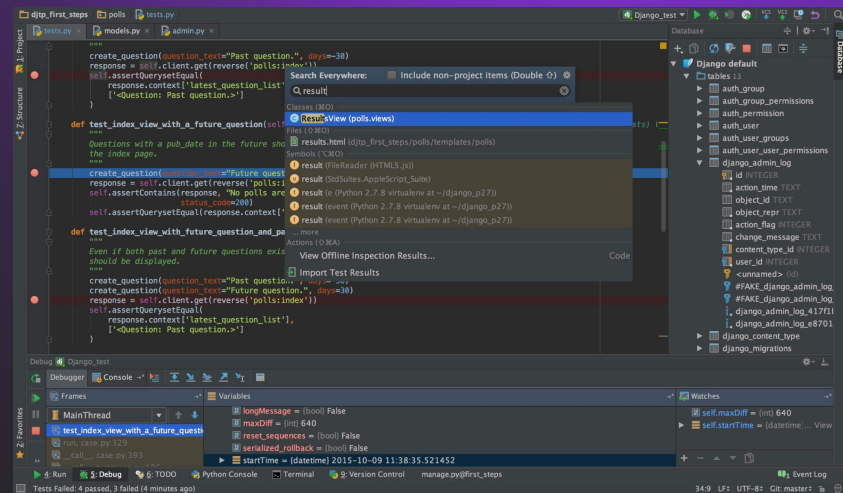
Задание №3



Напишите функцию, которая сохраняет созданный в прошлом задании файл в формате CSV.









Решение в IDE



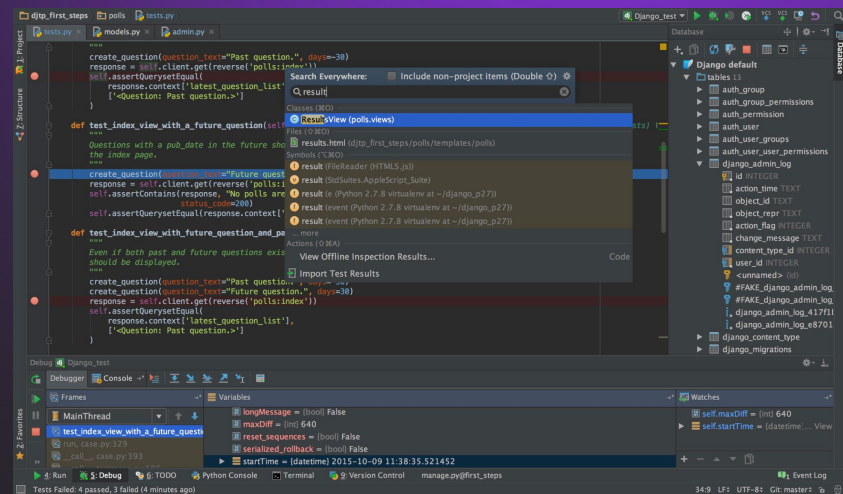


Задание №4

-  Прочитайте созданный в прошлом задании csv файл без использования `csv.DictReader`.
-  Дополните `id` до 10 цифр незначащими нулями.
-  В именах первую букву сделайте прописной.
-  Добавьте поле хеш на основе имени и идентификатора.
-  Получившиеся записи сохраните в `json` файл, где каждая строка `csv` файла представлена как отдельный `json` словарь.
-  Имя исходного и конечного файлов передавайте как аргументы функции.



Решение в IDE





Перерыв?

Голосуйте в чате



Перерыв...

<<7:00->>



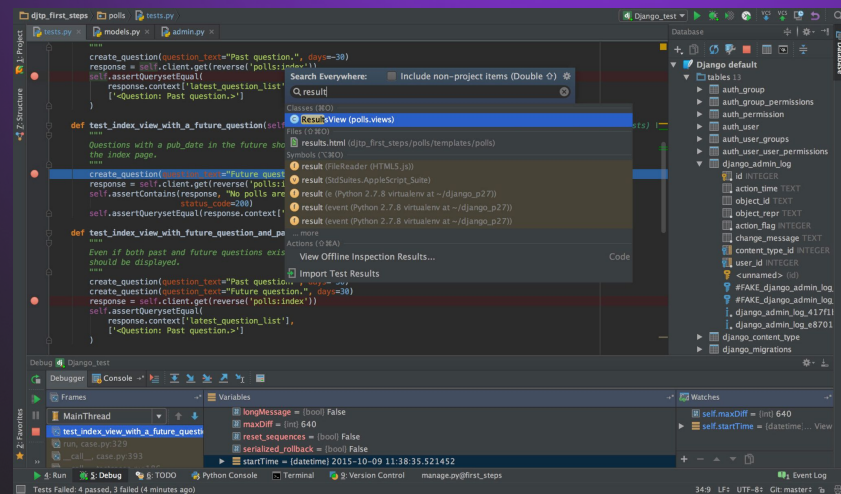
Задание №5



Напишите функцию, которая ищет json файлы в указанной директории и сохраняет их содержимое в виде одноимённых pickle файлов.






Решение в IDE



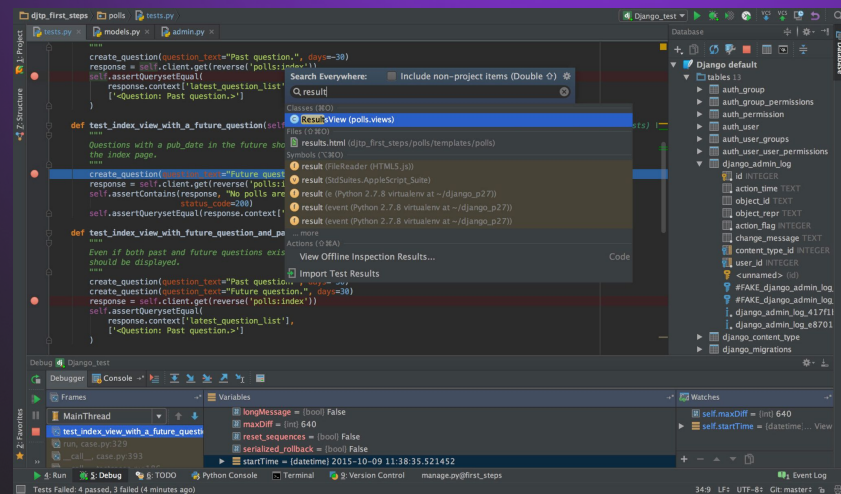


Задание №6

-  Напишите функцию, которая преобразует pickle файл хранящий список словарей в табличный csv файл.
-  Для тестированию возьмите pickle версию файла из задачи 4 этого семинара.
-  Функция должна извлекать ключи словаря для заголовков столбца из переданного файла.



Решение в IDE





Задание №7



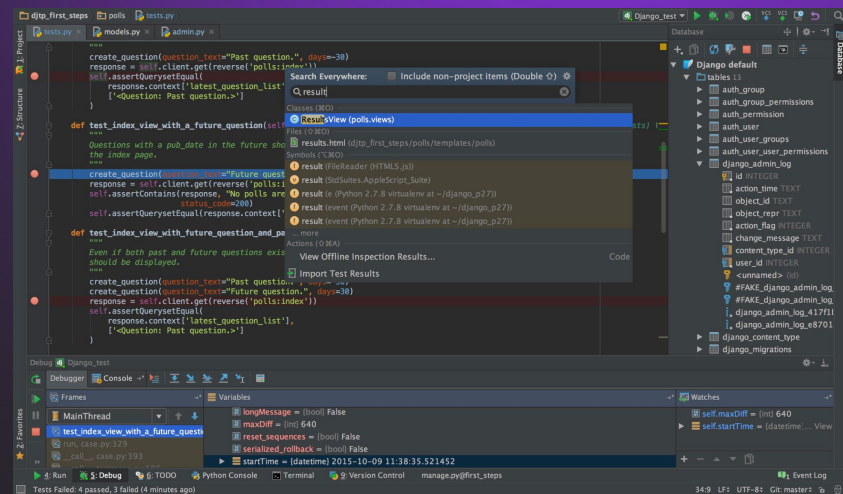
Прочитайте созданный в прошлом задании csv файл без использования `csv.DictReader`.



Распечатайте его как pickle строку.



Решение в IDE





Вопросы?

Вопросы?



Вопросы?





Домашнее задание

Задание



Решить задачи, которые не успели решить на семинаре.



Напишите функцию, которая получает на вход директорию и рекурсивно обходит её и все вложенные директории. Результаты обхода сохраните в файлы json, csv и pickle.

- Для дочерних объектов указывайте родительскую директорию.
- Для каждого объекта укажите файл это или директория.
- Для файлов сохраните его размер в байтах, а для директорий размер файлов в ней с учётом всех вложенных файлов и директорий.



Соберите из созданных на уроке и в рамках домашнего задания функций пакет для работы с файлами разных форматов.



Подведем итоги



Что было
сложного на
семинаре?





Напишите три вещи в
комментариях, которым
вы научились сегодня.





Как настроение?





Спасибо за работу!