

Информация и задания для подбора программиста python.

1. Используемый стек (знание выделенного красным - обязательно, синим - желательно):
 - a. Бэк: Python3, SQLAlchemy, AsyncIO, HttpAIO, UWSGI
 - b. Фронт: ReactJS, Mako, Babel
 - c. БД: MySQL/PostgreSQL, ClickHouse, Tarantool
 - d. Очереди: RabbitMQ
 - e. Телефония: Asterisk, FreeSwitch, Kamailio
2. Знание/умение работать с GIT

Задания:

1. Расписать что не так в функции (Python3):

```
import 'os'
func readFileId(names=[], mode):
    _ = ''
    id = -1
    for n in names:
        with os.open(n, 'w') as f:
            _ += f.read()
        f.close()
    print 'default: ' + id + ', actual: ' + _
    return id ? _ : id
```

2. Парсинг и вывод информации
 - a. Написать парсер для файла https://lk.globtelecom.ru/upload/test_prog1.csv
 - i. Результат должен быть сохранен в БД. Кодировка хранения - UTF8, в работе использовать SQLAlchemy
 - ii. Номера из столбца 1 должны быть нормализованы (нормализованные номера не имеют в своем теле ничего, кроме цифр. Длина номера - 11 символов). Кривые номера при парсинге должны принтануться в консоли, но не записываться в БД.
 - iii. Хранение в 2х таблицах. В одной номера телефонов, во второй - все остальное.
 - b. Написать скрипт, выдающий следующие результаты (на основании информации из БД, наполненной в п.2):
 - i. Общее кол-во записей
 - ii. Кол-во не уникальных номеров телефонов, и сами номера
 - iii. Статистика по людям: сколько человек в какой год родилось, кол-во однофамильцев
3. Поиск совпадающих аудио фрагментов, на основе "Отпечатков пальцев". На основе проекта <https://github.com/worldveil/dejavu> (проект работает на py2), сделать независимый модуль, который будет включен в состав платформы (не требуется оформление python модуля), отвечающий след. требованиям:
 - a. Работа под Py3.5

- b. Т.к. будет производиться обработка и поиск по совпадениям только wav файлов. Соответственно, все остальное (конвертеры, изображения и т.д.) - выкинуть
- c. Организовать хранение данных в БД. Для работы с БД - использовать SQLAlchemy. При построении структуры таблиц, а также при планировании запросов - учитывать, что кол-во образцов файлов, с которыми необходимо будет искать совпадения - будет не менее 200 (соответственно - кол-во отпечатков, ориентировочно - пара миллионов). Запросов к БД, на получение данных (select) - примерно 800/сек.
- d. Ожидается получить на класс (либо функции), содержащий в себе, как минимум, методы:
 - i. Обучальщика.
 - 1. Принимает строку (путь - может указывать на папку или на файл)
 - 2. Если указывает на файл, то обучается только по нему, если на папку, то проходимся по всем wav-файлам (рекурсивно в глубину) и сохраняет в БД все возможные образцы отпечатков + ссылки на оригинальные файлы.
 - 3. Ничего не возвращает
 - ii. Искателя.
 - 1. Принимает - строку (путь на конкретный wav-файл)
 - 2. Его задача - найти отпечаток пальца этого wav-файла в БД.
 - 3. Должно вернуть: отпечаток + имя оригинального файла (из датасета).
- e. Для проверки работоспособности модуля - нужно вырезать из одного из высланных файлов 2-3 секунды аудио-данных и выполнить поиск по созданной базе. (файлы будут присланы отдельно)