**Task 1. Анализ и обработка данных**

Во вложении вместе с данным документом вы получили json файл с примером реальных результатов разметки.

В проекте разметчику требовалось прочитать 2 текста и оценить релевантность ответа

Кратко описание полей результатов приводим ниже:

- assignment\_id – Уникальный идентификатор выполненного задания

- item\_id – Идентификатор задания

- file\_name – Имя файла задания, поданного на вход в систему

- item\_type – Тип задания (боевое / контрольное)

- marker\_id – уникальный id разметчика

- price – стоимость задания

- result – ответ разметчика

- result query/answer – задание, которое видел разметчик (скрыто, не имеет значения для выполнения задания)

- result scroll query / answer – если запрос / ответ, который видел разметчик не помещался на экране и отображался scroll, то этот показатель демонстрирует насколько целиком разметчик проскролил страницу вниз. Если значение 0, то значит все задание целиком умещалось на экране без скролла

- result final\_result – ответ разметчика

- start\_date / end\_date – дата и время начала и окончания разметки задания

- status – EXPIRED вышло время на разметку, задание не зачтено, ACCEPTED – выполнено

- consistency – согласованность ответов разметчиков

Все остальные поля во внимание принимать не требуется.

**Ваша задача:**

В свободной форме провести анализ представленных результатов. В ответе написать, какие выводы из этого документа можно сделать. Рассмотрите максимально возможное количество разрезов: качество результатов, дисциплина, работа разметчиков и т.д.

Будет плюсом вместе с анализом предоставить выкладки, как вы к таким выводам пришли (предпочтительно jupyter notebook сохраненный в html формате)

**Task2. Работа с pandas.**

Запишите в dataframe следующую таблицу

|  |  |  |
| --- | --- | --- |
| **Col1** | **Col2** | **Col3** |
| C | 1 |  |
| D | 2 |  |
| B | 3 |  |
| A | 4 |  |
| A | 5 |  |
| B | 6 |  |
| C | 7 |  |
| A | 8 |  |
| A | 9 |  |

Напишите код, который в столбец Col3 запишет True у первых двух строк, у которых Col1 = A

На выходе должны получить следующую таблицу

|  |  |  |
| --- | --- | --- |
| **Col1** | **Col2** | **Col3** |
| C | 1 |  |
| D | 2 |  |
| B | 3 |  |
| A | 4 | True |
| A | 5 | True |
| B | 6 |  |
| C | 7 |  |
| A | 8 |  |
| A | 9 |  |

В качестве результата выполнения задания необходимо приложить код, который приводит нас к требуемому результату. Подумайте, можно ли оптимизировать то, что у вас получилось

**Task 3. Установка и запуск SDK**

Ваша задача установить библиотеку crowd-sdk для работы с AC Tagme и далее в 3 задании продемонстрировать ваши умения разобраться в документации и осуществить ряд типовых действий.

Учетная запись, для подключения:

Логин: [customer\_demo@tagme.ru](mailto:customer_demo@tagme.ru)

Пас: R<S.x2dHqdrR

Ссылка на документацию и инструкцию по установке crowd-sdk

<https://tagme.sberdevices.ru/docs/sdk/installation.html>

В файле crowd.cfg вместо

url:

auth\_url:

Рекомендуем использовать:

stand: "cloud\_prom"

Данное задание не предполагает проверки, если у вас получилось установить crowd-sdk, переходите к следующему заданию.

**Task4. Работа с SDK.**

Все действия по данному заданию выполняются с использованием библиотеки crowd-sdk.

1. Создайте проект, назовите его случайным набором букв и цифр
2. В созданном проекте создайте 3 таски, назовите их по порядковому номеру (1-2-3)
3. В таске 1 измените перекрытие на 82
4. В таске 2 измените приоритет на 82
5. В таске 3 загрузите 3 рандомных текстовых документа (.txt) в качестве заданий
6. На главной странице есть проект с названием «Тестовый проект», необходимо выгрузить результаты разметки по таске «task» в формат xlsx. Обратите внимание, что текст задания у вас также должен быть выгружен в таблицу, если нету, необходимо подгрузить его.
7. Запустите таску «start/stop task»
8. Остановите таску «start/stop task»

В качестве результата выполнения задания необходимо приложить код, выполняющий описанные действия.