**PROJECT 4. MasterMind.**

Gorlova Kristina

**Цель:**

Подготовить основу рекомендательной системы.

**Задачи:**

1. Построить таблицу, в которой курсам будет соответствовать по две рекомендации.

2. Оформить и снабдить комментариями написанные в процессе скрипты.

3. Провести А/В тестирование и оценить его результат.

**Формализованные задачи:**

1. Познакомиться с датасетом, подготовить и проанализировать данные с помощью SQL.

2. Обработать данные средствами Python.

3. Составить итоговую таблицу с рекомендациями, снабдив её необходимыми комментариями, и представить отчёт продакт-менеджеру.

4. Проанализировать результаты A/B-теста, проведённого после внедрения фичи, и сделать вывод.

**SQL запрос.**

Необходимо было получить данные, соответствующие клиентам, которые купили более 1 курса. Код, считающий количество таких клиентов выглядит следующим образом:

SELECT

COUNT (DISTINCT purchase\_count.user\_id) /\*выводим количество уникальных пользователей\*/

FROM /\*из данных, полученных на основании подзапроса\*/

(SELECT

DISTINCT user\_id, /\*количество уникальных пользователей\*/

count (DISTINCT resource\_id) purchase /\*количество покупок уникальных курсов\*/

FROM final.carts /\*из таблицы carts\*/

JOIN final.cart\_items ON cart\_items.cart\_id = carts.id /\*соединенной с таблицей cart\_items\*/

WHERE resource\_type = 'Course' AND state = 'successful' /\*фильтрация данных по категории типу продукта 'Course' и статусе покупки 'successful' \*/

GROUP BY 1) AS purchase\_count /\*даем название подзапросу\*/

WHERE purchase\_count.purchase > 1 /\*фильтрация данных по количеству покупок более 1\*/

Для дальнейшей работы с данными необходимо осуществить выгрузку csv-файла по следующему запросу:

WITH

purchase\_count AS /\*даем название подзапросу\*/

(SELECT

DISTINCT user\_id, /\*выводим уникальные user\_id\*/

count (DISTINCT resource\_id) purchase /\*выводим количество уникальных resource\_id\*/

FROM final.carts /\*из таблицы carts\*/

JOIN final.cart\_items ON cart\_items.cart\_id = carts.id /\*соединенной с таблицей cart\_items\*/

WHERE resource\_type = 'Course' AND state = 'successful' /\*фильтрация данных по категории типу продукта 'Course' и статусе покупки 'successful' \*/

GROUP BY 1) /\*группируем по первому столбцу\*/

SELECT DISTINCT /\*выводим уникальные значения\*/

carts.user\_id, /\*user\_id\*/

cart\_items.resource\_id /\*resource\_id\*/

FROM final.carts /\*из таблицы carts\*/

JOIN final.cart\_items ON cart\_items.cart\_id = carts.id /\*соединенной с таблицей cart\_items\*/

JOIN purchase\_count ON purchase\_count.user\_id = carts.user\_id /\* и с подзапросом purchase\_count\*/

WHERE resource\_type = 'Course' AND state = 'successful' AND purchase\_count.purchase > 1 /\*фильтрация данных по категории типу продукта 'Course' и статусе покупки 'successful' и количеству покупок более 1 \*/

order by 1 /\*сортируем по первому столбцу\*/

Также, для работы нам потребуется список уникальных номеров курсов, который можно получить и выгрузить при выполнении следующего кода:

SELECT

DISTINCT resource\_id purchase /\*уникальные номера курсов\*/

FROM final.carts /\*из таблицы carts\*/

JOIN final.cart\_items ON cart\_items.cart\_id = carts.id /\*соединенной с таблицей cart\_items\*/

WHERE resource\_type = 'Course' /\*фильтрация данных по категории типу продукта 'Course' \*/