

# **Анализ данных в ClickHouse**

Александр Саушев

# Вопрос 1

Получить статистику по дням. Просто посчитать число всех событий по дням, число показов, число кликов, число уникальных объявлений и уникальных кампаний.

## Запрос

```
SELECT date,  
       COUNT(event) AS all_events,  
       countIf(event = 'view') AS views,  
       countIf(event = 'click') AS clicks,  
       uniqExact(ad_id) AS unique_ads,  
       uniqExact(campaign_union_id) AS unique_campaigns  
FROM ads_data  
GROUP BY date
```

## Результат

	date	all_events	views	clicks	unique_ads	unique_campaigns
1	2019-04-01	22073	21782	291	150	149
2	2019-04-02	47117	46572	545	344	336
3	2019-04-03	59483	59023	460	360	352
4	2019-04-04	275735	275092	643	407	396
5	2019-04-05	519707	427386	92321	465	442
6	2019-04-06	75885	60967	14918	220	212

# Вопрос 2

Разобраться, почему случился такой скачок 2019-04-05? Каких событий стало больше? У всех объявлений или только у некоторых?

## Ответ

Скачок 2019-04-05 случился из-за объявления с ad\_id **112583**. На него пришлись почти 71% всех просмотров за день и больше 98,5% кликов.

## Запрос

```
WITH (SELECT COUNT(event) AS all_events
      FROM ads_data
      WHERE date = '2019-04-05') AS all_events_2019_04_05,
      (SELECT countIf(event = 'view') AS views
      FROM ads_data
      WHERE date = '2019-04-05') AS all_views_2019_04_05,
      (SELECT countIf(event = 'click') AS clicks
      FROM ads_data
      WHERE date = '2019-04-05') AS all_clicks_2019_04_05
SELECT ad_id,
       COUNT(event)
       AS events,
       ROUND(events * 100 / all_events_2019_04_05, 2)
       AS perc_of_events_per_2019_04_05,
       countIf(event = 'view')
       AS views,
       ROUND(views * 100 / all_views_2019_04_05, 2)
       AS perc_of_views_per_2019_04_05,
       countIf(event = 'click')
       AS clicks,
       ROUND(clicks * 100 / all_clicks_2019_04_05, 2)
       AS perc_of_clicks_per_2019_04_05
FROM ads_data
WHERE date = '2019-04-05'
GROUP BY ad_id
ORDER BY events DESC
```

# Результат





	ad_id	events	perc_of_events...	views	perc_of_views...	clicks	perc_of_clicks...
1	112583	393828	75.78	302811	70.85	91017	98.59
2	107729	29745	5.72	29724	6.95	21	0.02
3	28142	20903	4.02	20872	4.88	31	0.03
4	38892	8437	1.62	7986	1.87	451	0.49
5	107837	8341	1.6	8338	1.95	3	0
6	37720	4944	0.95	4861	1.14	83	0.09
7	45008	3145	0.61	3145	0.74	0	0
8	18425	2313	0.45	2274	0.53	39	0.04
9	46629	2305	0.44	2302	0.54	3	0
10	42518	1942	0.37	1885	0.44	57	0.06
11	22490	1441	0.28	1440	0.34	1	0
12	98325	1332	0.26	1306	0.31	26	0.03
13	29881	1219	0.23	1203	0.28	16	0.02
14	44685	1121	0.22	1114	0.26	7	0.01
15	113350	984	0.19	975	0.23	9	0.01
16	111335	962	0.19	931	0.22	31	0.03
17	105076	914	0.18	910	0.21	4	0
18	121309	794	0.15	780	0.18	14	0.02
19	12030	786	0.15	777	0.18	9	0.01
20	39828	782	0.15	780	0.18	2	0

# Вопрос 3

Найти топ 10 объявлений по CTR за все время. CTR — это отношение всех кликов объявлений к просмотрам. Например, если у объявления было 100 показов и 2 клика,  $CTR = 0.02$ . Различается ли средний и медианный CTR объявлений в наших данных?

## Топ-10 объявлений по CTR




```
SELECT ad_id,
       countIf(event = 'view') AS views,
       countIf(event = 'click') AS clicks,
       ROUND((clicks / views), 4) AS CTR
FROM   ads_data
GROUP BY ad_id
HAVING views != 0
ORDER BY CTR DESC
LIMIT 10
```

	 ad_id ▾	 views ▾	 clicks ▾	 CTR ▾
1	117164	19	6	0.3158
2	112583	351802	105767	0.3006
3	42507	11	3	0.2727
4	98569	16	3	0.1875
5	46639	253	44	0.1739
6	23599	24	4	0.1667
7	19912	25	4	0.16
8	110414	32	5	0.1562
9	45969	13	2	0.1538
10	20662	26	4	0.1538

# Средний и медианный CTR объявлений

В наших данных средний и медианный CTR различаются. Средний больше медианного почти в 5,5 раз.

```
SELECT ROUND(AVG(CTR), 4) AS avg_CTR,  
       ROUND(medianExact(CTR), 4) AS median_CTR,  
       ROUND(avg_CTR / median_CTR, 2) AS avg_to_median  
FROM (  
  SELECT ad_id,  
         countIf(event = 'view') AS views,  
         countIf(event = 'click') AS clicks,  
         (clicks / views) AS CTR  
  FROM ads_data  
  GROUP BY ad_id  
  HAVING views != 0  
  ORDER BY CTR DESC)
```

	 avg_CTR ▾	 median_CTR ▾	 avg_to_median ▾
1	0.0158	0.0029	5.45

# Вопрос 4

Похоже, в наших логах есть баг: объявления приходят с кликами, но без показов! Сколько таких объявлений, есть ли какие-то закономерности? Эта проблема наблюдается на всех платформах?





## Ответ

Объявлений с таким багом девять, они встречаются на всех платформах.

## Запрос

```
SELECT ad_id,
       countIf(event = 'view') AS views,
       countIf(event = 'click') AS clicks,
       arraySort(groupUniqArray(platform)) AS platforms
FROM
    ads_data
GROUP BY ad_id
HAVING views = 0
```

## Результат

	 ad_id ▾	 views ▾	 clicks ▾	 platforms ▾
1	115825	0	4	['ios','web']
2	26204	0	6	['android','ios','web']
3	45418	0	3	['ios']
4	120431	0	35	['android','ios','web']
5	41500	0	20	['android','ios','web']
6	120796	0	1	['android']
7	120536	0	6	['android','ios']
8	117364	0	7	['android','ios']
9	19223	0	7	['android','ios','web']

# Закономерности

Я посмотрел на все записи проблемных объявлений и обнаружил только одну закономерность — у всех объявлений нет видео. Правда, видео есть только у 20 из 965 объявлений, поэтому это так себе особенность.

```
SELECT ad_id,
       client_union_id,
       campaign_union_id,
       has_video,
       platform,
       target_audience_count,
       ad_cost_type,
       ad_cost
FROM ads_data
WHERE ad_id IN (
  SELECT ad_id
  FROM (
    SELECT ad_id,
           countIf(event = 'view')
             AS views,
           countIf(event = 'click')
             AS clicks,
           arraySort(groupUniqArray(platform))
             AS platforms
    FROM ads_data
    GROUP BY ad_id
    HAVING views = 0 and clicks != 0)
)
```

```
ORDER BY ad_id
```

	ad_id	client_union_id	campaign_union_id	has_video	platform	target_audience_count	ad_cost
1	19223	1630	19223	0	android	61512	CPM
2	19223	1630	19223	0	android	61512	CPM
3	19223	1630	19223	0	android	61512	CPM
4	19223	1630	19223	0	web	61512	CPM
5	19223	1630	19223	0	ios	61512	CPM
6	19223	1630	19223	0	android	61512	CPM
7	19223	1630	19223	0	android	61512	CPM
8	26204	14606	26204	0	android	110586	CPM
9	26204	14606	26204	0	ios	110586	CPM
10	26204	14606	26204	0	ios	110586	CPM
11	26204	14606	26204	0	android	110586	CPM
12	26204	14606	26204	0	web	110586	CPM



# Вопрос 5




Есть ли различия в CTR у объявлений с видео и без?

## Ответ




Да, у объявлений с видео и средний и медианный CTR выше. Средний CTR отличается несильно (в 1,28 раз), а вот медианный значительно: у объявлений с видео в больше чем в 4 раза выше.

```
SELECT ROUND(AVG(CTR), 4) AS avg_CTR,  
       ROUND(medianExact(CTR), 4) AS median_CTR,  
       ROUND(avg_CTR / median_CTR, 2) AS avg_to_median  
FROM(  
SELECT ad_id,  
       has_video,  
       countIf(event = 'view') AS views,  
       countIf(event = 'click') AS clicks,  
       (clicks / views) AS CTR  
FROM ads_data  
WHERE has_video = 1 -- 0  
GROUP BY ad_id, has_video  
HAVING views != 0  
ORDER BY CTR DESC)
```

С видео


	 avg_CTR ▾	 median_CTR ▾	 avg_to_median ▾
1	0.0201	0.0122	1.65

Без видео

	 avg_CTR ▾	 median_CTR ▾	 avg_to_median ▾
1	0.0157	0.0028	5.61

Чему равняется 95 процентиль CTR по всем объявлениям за 2019-04-04?

```
SELECT quantileExact(0.95)(CTR)
FROM (SELECT ad_id,
              countIf(event = 'view') AS views,
              countIf(event = 'click') AS clicks,
              (clicks / views)        AS CTR
        FROM ads_data
       WHERE date = '2019-04-04'
      GROUP BY ad_id
     HAVING views != 0
    ORDER BY CTR DESC)
```

	 `quantileExact(0.95)(CTR)` ▾
1	0.08333333333333333

# Вопрос 6

Для финансового отчета нужно рассчитать наш заработок по дням. В какой день мы заработали больше всего? В какой меньше?

Мы списываем с клиентов деньги, если произошел клик по CPC объявлению, и мы списываем деньги за каждый показ CPM объявления. Если у CPM объявления цена 200 рублей, то за один показ мы зарабатываем  $200 / 1000$ .

## Ответ

Больше всего мы заработали 2019-04-05 (чуть больше 96 тысяч рублей), а меньше всего — 2019-04-01 (всего 6 656 рублей).

```
SELECT date,
        ROUND(sumIf(ad_cost, ad_cost_type = 'CPC'
                     AND event = 'click'), 2)
        AS cpc_sum_cost,
        ROUND(sumIf(ad_cost / 1000, ad_cost_type = 'CPM'
                     AND event = 'view'), 2)
        AS cpm_sum_cost,
        ROUND(cpc_sum_cost + cpm_sum_cost, 2)
        AS total_cost
FROM ads_data
GROUP BY date
ORDER BY total_cost DESC
```

	date	cpc_sum_cost	cpm_sum_cost	total_cost
1	2019-04-05	9446.2	86676.92	96123.12
2	2019-04-04	3517.3	51471.5	54988.8
3	2019-04-03	4177.9	9968.06	14145.96
4	2019-04-06	854.4	12492.53	13346.93
5	2019-04-02	5994.4	7291.39	13285.79
6	2019-04-01	3321.1	3334.61	6655.71

# Вопрос 7

На какой платформе больше всего показов? Сколько процентов показов приходится на каждую из платформ (колонка platform)?

## Ответ

Больше всего показов на «Андроиде». На него приходится половина всех показов.

```
WITH (SELECT countIf(event = 'view')
      FROM ads_data) AS total_views
SELECT platform,
       countIf(event = 'view')
       AS views_by_platform,
       ROUND(views_by_platform * 100 / total_views, 2)
       AS perc_of_total_views
FROM ads_data
GROUP BY platform
ORDER BY views_by_platform DESC
```

	platform	views_by_platform	perc_of_total_views
1	android	445722	50.03
2	ios	267117	29.99
3	web	177983	19.98

# Вопрос 8

А есть ли такие объявления, по которым сначала произошел клик, а только потом показ?

## Ответ

Объявление, клик по которым случился раньше просмотра, есть. Всего их 12 штук.

```
SELECT ad_id,  
       minIf(time, event = 'view') AS first_view_time,  
       minIf(time, event = 'click') AS first_click_time  
FROM ads_data  
GROUP BY ad_id  
HAVING first_view_time > first_click_time  
AND first_click_time != '1970-01-01 00:00:00'
```

	ad_id	first_view_time	first_click_time
1	18681	2019-04-05 02:45:35	2019-04-05 00:18:20
2	23599	2019-04-05 05:48:26	2019-04-05 00:05:26
3	32386	2019-04-03 00:03:25	2019-04-03 00:03:23
4	33033	2019-04-05 03:33:11	2019-04-05 00:10:28
5	36758	2019-04-04 01:23:42	2019-04-04 01:21:18
6	38224	2019-04-04 00:09:24	2019-04-04 00:02:30
7	44283	2019-04-04 00:11:36	2019-04-04 00:09:24
8	44766	2019-04-05 02:02:57	2019-04-05 00:54:49
9	46639	2019-04-02 00:02:06	2019-04-02 00:01:55
10	98569	2019-04-04 07:55:00	2019-04-04 03:09:35
11	107798	2019-04-02 00:21:14	2019-04-02 00:20:02
12	114886	2019-04-02 00:40:49	2019-04-02 00:31:51

## Второй вариант

После ревью мне посоветовали попробовать решить это задание с помощью `argMin()` — и я попробовал. `argMin()` возвращает на 9 записей больше, чем первый вариант, потому что в нем учитываются объявления, у которых есть клики, но нет просмотров.

```
SELECT ad_id,  
       argMin(event, time) AS first_action  
FROM ads_data  
GROUP BY ad_id  
HAVING first_action = 'click'  
ORDER BY ad_id
```

	ad_id	first_action
1	18681	click
2	19223	click
3	23599	click
4	26204	click
5	32386	click
6	33033	click
7	36758	click
8	38224	click
9	41500	click
10	44283	click
11	44766	click
12	45418	click
13	46639	click
14	98569	click
15	107798	click
16	114886	click
17	115825	click
18	117364	click
19	120431	click
20	120536	click
21	120796	click

А вот как я это выяснил:

```
SELECT ad_id,  
       minIf(time, event = 'view') AS first_view_time,  
       minIf(time, event = 'click') AS first_click_time,  
       argMin(event, time) AS first_action  
FROM ads_data  
GROUP BY ad_id  
HAVING first_action = 'click'  
ORDER BY first_view_time
```

	ad_id	first_view_time	first_click_time	first_action
1	115825	1970-01-01 00:00:00	2019-04-01 01:30:30	click
2	26204	1970-01-01 00:00:00	2019-04-01 02:33:00	click
3	45418	1970-01-01 00:00:00	2019-04-01 06:43:08	click
4	120431	1970-01-01 00:00:00	2019-04-01 03:55:10	click
5	41500	1970-01-01 00:00:00	2019-04-01 07:10:30	click
6	120796	1970-01-01 00:00:00	2019-04-01 18:57:47	click
7	120536	1970-01-01 00:00:00	2019-04-01 00:17:08	click
8	117364	1970-01-01 00:00:00	2019-04-01 06:19:03	click
9	19223	1970-01-01 00:00:00	2019-04-01 11:37:39	click
10	46639	2019-04-02 00:02:06	2019-04-02 00:01:55	click
11	107798	2019-04-02 00:21:14	2019-04-02 00:20:02	click
12	114886	2019-04-02 00:40:49	2019-04-02 00:31:51	click
13	32386	2019-04-03 00:03:25	2019-04-03 00:03:23	click
14	38224	2019-04-04 00:09:24	2019-04-04 00:02:30	click
15	44283	2019-04-04 00:11:36	2019-04-04 00:09:24	click
16	36758	2019-04-04 01:23:42	2019-04-04 01:21:18	click
17	98569	2019-04-04 07:55:00	2019-04-04 03:09:35	click
18	44766	2019-04-05 02:02:57	2019-04-05 00:54:49	click
19	18681	2019-04-05 02:45:35	2019-04-05 00:18:20	click
20	33033	2019-04-05 03:33:11	2019-04-05 00:10:28	click
21	23599	2019-04-05 05:48:26	2019-04-05 00:05:26	click