Анализ сервиса вопросов и ответов по программированию

Описание: С помощью SQL посчитать лючевые метрики сервис-системы вопросов и ответов о программировании.

Данные таблиц:

Таблица stackoverflow.badges

Хранит информацию о значках, которые присуждаются за разные достижения. Например, пользователь, правильно ответивший на большое количество вопросов про PostgreSQL, может получить значок postgresql.

Поле	Описание
id	Идентификатор значка, первичный ключ таблицы
name	Название значка
user_id	Идентификатор пользователя, которому присвоили значок, внешний ключ, отсылающий к таблице users
creation_date	Дата присвоения значка

Таблица stackoverflow.post_types

Содержит информацию о типе постов. Их может быть два:

Question — пост с вопросом;
Answer — пост с ответом.

Поле	Описание
id	Идентификатор поста, первичный ключ таблицы
type	Тип поста

Таблица stackoverflow.posts

Содержит информацию о постах.

Поле	Описание

id	Идентификатор поста, первичный ключ таблицы
title	Заголовок поста
creation_date	Дата создания поста
favorites_count	Число, которое показывает, сколько раз пост добавили в «Закладки»
last_activity_date	Дата последнего действия в посте, например комментария
last_edit_date	Дата последнего изменения поста
user_id	Идентификатор пользователя, который создал пост, внешний ключ к таблице users
parent_id	Если пост написали в ответ на другую публикацию, в это поле попадёт идентификатор поста с вопросом
post_type_id	Идентификатор типа поста, внешний ключ к таблице post_types
score	Количество очков, которое набрал пост
views_count	Количество просмотров

Таблица stackoverflow.users

Содержит информацию о пользователях.

Поле	Описание
id	Идентификатор пользователя, первичный ключ таблицы
creation_date	Дата регистрации пользователя

display_name	Имя пользователя
last_access_date	Дата последнего входа
location	Местоположение
reputation	Очки репутации, которые получают за хорошие вопросы и полезные ответы
views	Число просмотров профиля пользователя

Таблица stackoverflow.vote_types

Содержит информацию о типах голосов. Голос — это метка, которую пользователи ставят посту. Типов бывает несколько:

UpMod — такую отметку получают посты с вопросами или ответами, которые пользователи посчитали уместными и полезными.
DownMod — такую отметку получают посты, которые показались пользователям наименее полезными.
Close — такую метку ставят опытные пользователи сервиса, если заданный вопрос нужно доработать или он вообще не подходит для платформы.
Offensive — такую метку могут поставить, если пользователь ответил на вопрос в грубой и оскорбительной манере, например, указав на неопытность автора поста.
Spam — такую метку ставят в случае, если пост пользователя выглядит откровенной рекламой.

Поле	Описание
id	Идентификатор типа голоса, первичный ключ
name	Название метки

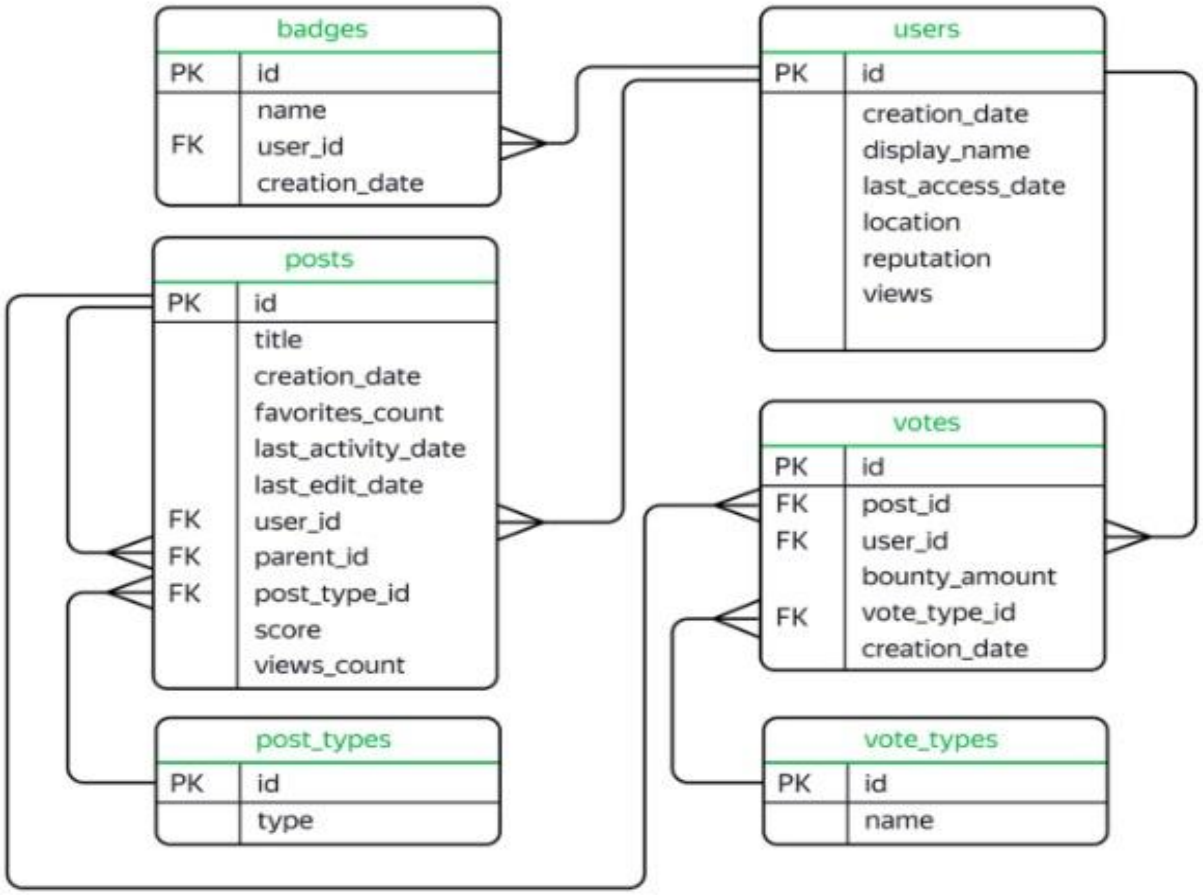
Таблица stackoverflow.votes

Содержит информацию о голосах за посты.

Поле	Описание

id	Идентификатор голоса, первичный ключ
post_id	Идентификатор поста, внешний ключ к таблице posts
user_id	Идентификатор пользователя, который поставил посту голос, внешний ключ к таблице users
bounty_amount	Сумма вознаграждения, которое назначают, чтобы привлечь внимание к посту
vote_type_id	Идентификатор типа голоса, внешний ключ к таблице vote_types
creation_date	Дата назначения голоса

ER-диаграмма



Задание1. Найдите количество вопросов, которые набрали больше 300 очков или как минимум 100 раз были добавлены в «Закладки».

Код

```
SELECT COUNT(post_type_id)
FROM stackoverflow.posts
WHERE  post_type_id=1 AND (score>300 OR favorites_count>=100 );
```

Результат

count
1355

Задание 2. Сколько в среднем в день задавали вопросов с 1 по 18 ноября 2008 включительно? Результат округлите до целого числа.

Код

```
SELECT ROUND(AVG(count_per_day))
FROM (

SELECT distinct COUNT(post_type_id) OVER (PARTITION BY DATE_TRUNC('day', creation_date)::date) AS
count_per_day
FROM stackoverflow.posts

WHERE (creation_date::date BETWEEN '2008-11-1' AND '2008-11-18') AND post_type_id = 1
) as average;
```

Результат

round
383

Задание 3. Сколько пользователей получили значки сразу в день регистрации? Выведите количество уникальных пользователей.

Код

```
SELECT COUNT(DISTINCT user_id)
FROM stackoverflow.badges b
JOIN stackoverflow.users u ON u.id=b.user_id
WHERE b.creation_date::date=u.creation_date::date;
```

Результат

count
7047

Задание 4. Сколько уникальных постов пользователя с именем Joel Coehoorn получили хотя бы один голос?

Код

```
select count(distinct p.id)
from stackoverflow.posts as p
inner join stackoverflow.users as u on u.id= p.user_id
inner join stackoverflow.votes  as v on p.id = v.post_id
where u.display_name  LIKE '%Joel Coehoorn%'
```

Результат

count
12

Задание 5. Выгрузите все поля таблицы vote_types. Добавьте к таблице поле rank, в которое войдут номера записей в обратном порядке. Таблица должна быть отсортирована по полю id.

Код

```
select *, RANK() OVER (ORDER BY id desc)
from stackoverflow.vote_types
order by id
```

Результат

id	name	rank
1	AcceptedByOriginator	15
2	UpMod	14
3	DownMod	13
4	Offensive	12
5	Favorite	11
6	Close	10
7	Reopen	9
8	BountyStart	8
9	BountyClose	7

Задание 6. Отберите 10 пользователей, которые поставили больше всего голосов типа Close. Отобразите таблицу из двух полей: идентификатором пользователя и количеством голосов. Отсортируйте данные сначала по убыванию количества голосов, потом по убыванию значения идентификатора пользователя.

Код

```
SELECT v.user_id,
       COUNT (vt.id) as CV
FROM stackoverflow.votes v

JOIN stackoverflow.vote_types vt ON v.vote_type_id=vt.id
WHERE vt.name = 'Close'
GROUP BY v.user_id
ORDER BY CV DESC, v.user_id DESC
LIMIT 10
```

Результат

user_id	cv
20646	36
14728	36
27163	29
41158	24
24820	23
9345	23
3241	23
44330	20
38426	19

Задание 7. Отберите 10 пользователей по количеству значков, полученных в период с 15 ноября по 15 декабря 2008 года включительно. Отобразите несколько полей:

идентификатор пользователя;
число значков;
место в рейтинге — чем больше значков, тем выше рейтинг.

Пользователям, которые набрали одинаковое количество значков, присвойте одно и то же место в рейтинге.
Отсортируйте записи по количеству значков по убыванию, а затем по возрастанию значения идентификатора пользователя.

Код

```
SELECT DISTINCT user_id,
       COUNT(id),
       DENSE_RANK() OVER (ORDER BY COUNT(id)DESC)
FROM stackoverflow.badges
WHERE DATE_TRUNC('day', creation_date)::date BETWEEN '2008-11-15' AND '2008-12-15'
GROUP BY 1
ORDER BY 2 desc,1
LIMIT 10
```

Результат

user_id	count	dense_rank
22656	149	1
34509	45	2
1288	40	3
5190	31	4
13913	30	5
893	28	6
10661	28	6
33213	25	7
12950	23	8

Задание 8. Сколько в среднем очков получает пост каждого пользователя? Сформируйте таблицу из следующих полей:

заголовок поста;
идентификатор пользователя;
число очков поста;
среднее число очков пользователя за пост, округлённое до целого числа.

Не учитывайте посты без заголовка, а также те, что набрали ноль очков.

Код

```
1 select title, user_id, score, round(avg(score) over (partition by user_id))
2 from stackoverflow.posts
3 where title is not null and score != 0
```

Результат

title	user_id	score	round
Diagnosing Deadlocks in SQL Server 2005	1	82	573
How do I calculate someone's age in C#?	1	1743	573
Why doesn't IE7 copy <pre><code> blocks to the clipboard correctly?	1	37	573
Calculate relative time in C#	1	1348	573
Wrapping StopWatch timing with a delegate or lambda?	1	92	573
Practical non-image based CAPTCHA approaches?	1	318	573
Parameterize an SQL IN clause	1	953	573
Escaping Bracket [in a CONTAINS() clause?	1	10	573
Binary Data in MySQL	2	169	76

Задание 9. Отобразите заголовки постов, которые были написаны пользователями, получившими более 1000 значков. Посты без заголовков не должны попасть в список.

Код

```
SELECT title
FROM stackoverflow.posts
WHERE title IS NOT NULL AND score > 0
AND user_id IN (
  select user_id
from stackoverflow.badges
group by user_id
having count(id) > 1000)
```

Результат

title
What's the strangest corner case you've seen in C# or .NET?
What's the hardest or most misunderstood aspect of LINQ?
What are the correct version numbers for C#?
Project management to go with GitHub

Задание 10.
Напишите запрос, который выгрузит данные о пользователях из США (англ. United States). Разделите пользователей на три группы в зависимости от количества просмотров их профилей:

> пользователям с числом просмотров больше либо равным 350 присвойте группу 1;
> пользователям с числом просмотров меньше 350, но больше либо равно 100 — группу 2;
> пользователям с числом просмотров меньше 100 — группу 3.

Отобразите в итоговой таблице идентификатор пользователя, количество просмотров профиля и группу. Пользователи с нулевым количеством просмотров не должны войти в итоговую таблицу.

Код

```
SELECT id, views,
       CASE
           WHEN views >= 350 THEN 1
           WHEN (views < 350 and views>= 100)  THEN  2
           WHEN views < 100 THEN 3
       END  as category
FROM stackoverflow.users
where views > 0 and location LIKE '%United States%'
```

Результат

id	views	category
3	24396	1
13	35414	1
23	757	1
25	3837	1
36	505	1
43	394	1
45	1971	1
50	1616	1
64	866	1

Задание 11. Дополните предыдущий запрос. Отобразите лидеров каждой группы — пользователей, которые набрали максимальное число просмотров в своей группе. Выведите поля с идентификатором пользователя, группой и количеством просмотров. Отсортируйте таблицу по убыванию просмотров, а затем по возрастанию значения идентификатора.

Код

```sql
WITH tab_1 as
(SELECT id, views,
       CASE
           WHEN views >= 350 THEN 1
           WHEN (views < 350 and views>= 100)  THEN  2
           WHEN views < 100 THEN 3
       END  as category
FROM stackoverflow.users
where views > 0 and location LIKE '%United States%')

SELECT id,
       category,
       views
FROM
  (SELECT *,
          rank() over(PARTITION BY category
                      ORDER BY views DESC) AS ranks
   FROM tab_1) AS tab_2
WHERE ranks = 1
ORDER BY views DESC, id
```

Результат

id	category	views
16587	1	62813
9094	2	349
9585	2	349
15079	2	349
33437	2	349
3469	3	99
4829	3	99
19006	3	99
22732	3	99

Задание 12. Посчитайте ежедневный прирост новых пользователей в ноябре 2008 года. Сформируйте таблицу с полями: номер дня; число пользователей, зарегистрированных в этот день; сумму пользователей с накоплением.

Код

```
select distinct extract(DAY from CAST(creation_date as date)) ,
count(id) over (partition by extract(DAY from creation_date::date) ), count(id) over (order by
extract(DAY from CAST(creation_date as date))) as cum_sum

from stackoverflow.users
where cast(creation_date as date) between '2008-11-01' and '2008-11-30'
```

Результат

date_part	count	cum_sum
1	34	34
2	48	82
3	75	157
4	192	349
5	122	471
6	132	603
7	104	707
8	42	749
9	45	794

Задание 13. Для каждого пользователя, который написал хотя бы один пост, найдите интервал между регистрацией и временем создания первого поста. Отобразите: идентификатор пользователя; разницу во времени между регистрацией и первым постом.

Код

```
WITH pre AS (SELECT u.id AS u_id,
       u.creation_date AS u_creation,
       FIRST_VALUE(p.creation_date) OVER (PARTITION BY p.user_id ORDER BY p.creation_date) AS
first_post
FROM stackoverflow.users u
JOIN stackoverflow.posts p ON u.id=p.user_id)

SELECT DISTINCT u_id,
       first_post - u_creation
FROM pre
```

Результат

u_id	?column?
27088	22 days, 10:32:25
4666	4 days, 13:51:01
43473	0:00:00
761503	0:00:00
9293	8 days, 0:39:24
22972	51 days, 12:36:52
17941	0:20:35
10950	19 days, 13:44:24
12151	2 days, 18:53:29

Часть 2

Задание 1. Выведите общую сумму просмотров постов за каждый месяц 2008 года. Если данных за какой-либо месяц в базе нет, такой месяц можно пропустить. Результат отсортируйте по убыванию общего количества просмотров.

Код

```
select distinct (date_trunc('month', creation_date)::date),
sum(views_count) over (partition by date_trunc('month', creation_date)::date  )
from stackoverflow.posts
where extract(year from creation_date::date) = 2008
order by sum desc
```

Результат

date_trunc	sum
2008-09-01	452928568
2008-10-01	365400138
2008-11-01	221759651
2008-12-01	197792841
2008-08-01	131367083
2008-07-01	669895

Задание 2. Выведите имена самых активных пользователей, которые в первый месяц после регистрации (включая день регистрации) дали больше 100 ответов. Вопросы, которые задавали пользователи, не учитывайте. Для каждого имени пользователя выведите количество уникальных значений user_id. Отсортируйте результат по полю с именами в лексикографическом порядке.

Код

```
select display_name, am1
from (select display_name, count(distinct user_id) as am1, count(distinct post_id) as am2
from (select utab.display_name,user_id, id as post_id, creation_date as post_date,utab.user_date,
utab.user_new_date
      from stackoverflow.posts as ptab
      join (select display_name, id as user_id1,creation_date as user_date,creation_date + INTERVAL
'1 month' as user_new_date
      from stackoverflow.users
      group by id, display_name, creation_date
      order by display_name) as utab on ptab.user_id= utab.user_id1
where post_type_id = 2) as tab1
where post_date::date between user_date::date and user_new_date::date
group by display_name) as tab2
where am2 > 100
order by display_name
```

Результат

display_name	am1
1800 INFORMATION	1
Adam Bellaire	1
Adam Davis	1
Adam Liss	1
aku	1
Alan	8
Amy B	1
anjanb	1
Ben Hoffstein	1

Задание 3. Выведите количество постов за 2008 год по месяцам. Отберите посты от пользователей, которые зарегистрировались в сентябре 2008 года и сделали хотя бы один пост в декабре того же года. Отсортируйте таблицу по значению месяца по убыванию.

Код

```sql
WITH sept_users AS
    (SELECT u.id,
        u.creation_date::date AS reg_date,
        p.creation_date::date AS post_date
    FROM stackoverflow.users u
    JOIN stackoverflow.posts p ON p.user_id=u.id
    WHERE (u.creation_date::date BETWEEN '2008-09-01' AND '2008-09-30')
        AND (p.creation_date::date BETWEEN '2008-12-01' AND '2008-12-31'))
SELECT DATE_TRUNC('month', pt.creation_date)::date AS month,
    COUNT(distinct pt.id)
FROM sept_users su
JOIN stackoverflow.posts pt ON pt.user_id=su.id
GROUP BY month
ORDER BY month DESC;
```

Результат

month	count
2008-12-01	17641
2008-11-01	18294
2008-10-01	27171
2008-09-01	24870
2008-08-01	32

Задание 4. Используя данные о постах, выведите несколько полей:

идентификатор пользователя, который написал пост;
дата создания поста;
количество просмотров у текущего поста;
сумму просмотров постов автора с накоплением.

Данные в таблице должны быть отсортированы по возрастанию идентификаторов пользователей, а данные об одном и том же пользователе — по возрастанию даты создания поста.

Код

```
1 select user_id, creation_date,views_count,
2 sum(views_count) over (partition by user_id order by creation_date)
3 from stackoverflow.posts
4 order by user_id, creation_date
```

Результат

user_id	creation_date	views_count	sum
1	2008-07-31 23:41:00	480476	480476
1	2008-07-31 23:55:38	136033	616509
1	2008-07-31 23:56:41	0	616509
1	2008-08-04 02:45:08	0	616509
1	2008-08-04 04:31:03	0	616509
1	2008-08-04 08:04:42	0	616509
1	2008-08-10 08:28:52	0	616509
1	2008-08-11 19:23:47	0	616509
1	2008-08-12 00:30:43	0	616509

Задание 5. Сколько в среднем дней в период с 1 по 7 декабря 2008 года включительно пользователи взаимодействовали с платформой? Для каждого пользователя отберите дни, в которые он или она опубликовали хотя бы один пост. Нужно получить одно целое число — не забудьте округлить результат.

Код

```
with a as
(select distinct user_id, count(creation_date::date) over (partition by user_id) as
counting
from stackoverflow.posts
where creation_date::date between '2008-12-01' and '2008-12-07'
group by user_id, creation_date::date)


select round(avg(a.counting))
from a
```

Результат

round
2

Задание 6. На сколько процентов менялось количество постов ежемесячно с 1 сентября по 31 декабря 2008 года? Отобразите таблицу со следующими полями:

номер месяца;
количество постов за месяц;
процент, который показывает, насколько изменилось количество постов в текущем месяце по сравнению с предыдущим.

Если постов стало меньше, значение процента должно быть отрицательным, если больше — положительным. Округлите значение процента до двух знаков после запятой.
Напомним, что при делении одного целого числа на другое в PostgreSQL в результате получится целое число, округлённое до ближайшего целого вниз. Чтобы этого избежать, переведите делимое в тип numeric.

Код

```
with a as
(select extract(month from creation_date::date) as month ,count(id) as count_month
from stackoverflow.posts
where creation_date::date between '2008-09-01' and '2008-12-31'
 group by month)

select *,
ROUND((100 * CAST(count_month AS numeric) / LAG(count_month, 1) OVER (ORDER by month) -
100),2) AS percentage
from a
```

Результат

month	count_month	percentage
9	70371	
10	63102	-10.33
11	46975	-25.56
12	44592	-5.07

Задание 7. Выгрузите данные активности пользователя, который опубликовал больше всего постов за всё время. Выведите данные за октябрь 2008 года в таком виде: номер недели; дата и время последнего поста, опубликованного на этой неделе.

Код

```
with user_1 as
(select distinct user_id,
       count(id)
from stackoverflow.posts
group by user_id
order by count(id) desc
limit 1)

select distinct extract (week from creation_date::date),
       max(creation_date) over (partition by extract (week from creation_date::date))
from stackoverflow.posts p
     join user_1 u on p.user_id = u.user_id
where creation_date between '01-10-2008' and '2008-11-01'
```

Результат

date_part	max
40	2008-10-05 09:00:58
41	2008-10-12 21:22:23
42	2008-10-19 06:49:30
43	2008-10-26 21:44:36
44	2008-10-31 22:16:01