

```
1 --1. Найдите количество вопросов, которые набрали больше 300 очков или как минимум 100 раз
  были добавлены в «Закладки».
2
3 SELECT count (*)
4 FROM stackoverflow.posts
5 WHERE post_type_id = 1 AND (score > 300 OR favorites_count >=100);
6
7 --2.Сколько в среднем в день задавали вопросов с 1 по 18 ноября 2008 включительно?
  Результат округлите до целого числа.
8
9 WITH p AS
10 (SELECT count (post_type_id) AS kol
11 FROM stackoverflow.posts
12 WHERE post_type_id = 1 AND ( creation_date::date BETWEEN '2008-11-01' AND '2008-11-18') )
13
14 SELECT ROUND(AVG(kol)/18)
15 FROM p;
16
17 --3.Сколько пользователей получили значки сразу в день регистрации? Выведите количество
  уникальных пользователей.
18
19 SELECT COUNT( DISTINCT u.id)
20 FROM stackoverflow.users AS u
21 INNER JOIN stackoverflow.badges AS b ON u.id = b.user_id
22 WHERE u.creation_date::date = b.creation_date::date;
23
24 --4.Сколько уникальных постов пользователя с именем Joel Coehoorn получили хотя бы один
  голос?
25
26 SELECT COUNT (DISTINCT p.id)
27 FROM stackoverflow.posts AS p
28 JOIN stackoverflow.votes AS v ON p.id = v.post_id
29 JOIN stackoverflow.users AS u ON p.user_id = u.id
30 WHERE u.display_name LIKE 'Joel Coehoorn' AND v.id >=1;
31
32 --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.
33
34 SELECT *,
35         ROW_NUMBER() OVER(ORDER BY id DESC) AS rank
36 FROM stackoverflow.vote_types
37 ORDER BY id;
38
39 --6. Отберите 10 пользователей, которые поставили больше всего голосов типа Close.
  Отобразите таблицу из двух полей: идентификатором пользователя и количеством голосов.
40 --Отсортируйте данные сначала по убыванию количества голосов, потом по убыванию значения
  идентификатора пользователя.
41
42 SELECT *
43 FROM (
44     SELECT v.user_id,
45            COUNT(v.id) AS cnt
46     FROM stackoverflow.votes AS v
47     WHERE vote_type_id=6
48     GROUP BY v.user_id
49     ORDER BY cnt DESC LIMIT 10
50 ) AS t
51 ORDER BY t.cnt DESC, t.user_id DESC;
52
53 --7. Отберите 10 пользователей по количеству значков, полученных в период с 15 ноября по 15
  декабря 2008 года включительно.
54 --Отобразите несколько полей:
```

```
55 --идентификатор пользователя;
56 --число значков;
57 --место в рейтинге — чем больше значков, тем выше рейтинг.
58 --Пользователям, которые набрали одинаковое количество значков, присвойте одно и то же
    место в рейтинге.
59 --Отсортируйте записи по количеству значков по убыванию, а затем по возрастанию значения
    идентификатора пользователя.
60
61 SELECT *,
62     DENSE_RANK() OVER(ORDER BY t.cnt DESC)
63 FROM (
64     SELECT b.user_id,
65            COUNT(b.id) AS cnt
66     FROM stackoverflow.badges AS b
67     WHERE creation_date::date BETWEEN '2008-11-15' AND '2008-12-15'
68     GROUP BY b.user_id
69     ORDER BY cnt DESC LIMIT 10
70 ) AS t
71
72 ORDER BY t.cnt DESC, t.user_id;
73
74 --8. Сколько в среднем очков получает пост каждого пользователя?
75 --Сформируйте таблицу из следующих полей:
76 --заголовок поста;
77 --идентификатор пользователя;
78 --число очков поста;
79 --среднее число очков пользователя за пост, округлённое до целого числа.
80 --Не учитывайте посты без заголовка, а также те, что набрали ноль очков.
81
82 SELECT title,
83        user_id,
84        score,
85        ROUND(AVG(score) OVER (PARTITION BY user_id) )AS avg_score
86 FROM stackoverflow.posts
87 WHERE title IS NOT NULL AND score!=0
88 ORDER BY user_id;
89
90 --9.Отобразите заголовки постов, которые были написаны пользователями, получившими более
    1000 значков.
91 --Посты без заголовков не должны попасть в список.
92
93 WITH t AS
94 (SELECT b.user_id
95  FROM stackoverflow.badges AS b
96  GROUP BY b.user_id
97  HAVING COUNT (b.id) > 1000)
98
99 SELECT p.title
100 FROM stackoverflow.posts AS p
101 LEFT JOIN stackoverflow.users AS u ON p.user_id=u.id
102 JOIN t ON u.id= t.user_id
103 WHERE title IS NOT NULL;
104
105 --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:
106 --пользователям с числом просмотров больше либо равным 350 присвойте группу 1;
107 --пользователям с числом просмотров меньше 350, но больше либо равно 100 — группу 2;
108 --пользователям с числом просмотров меньше 100 — группу 3.
109 --Отобразите в итоговой таблице идентификатор пользователя, количество просмотров профиля и
    группу.
110 --Пользователи с нулевым количеством просмотров не должны войти в итоговую таблицу.
111
```

```
112 SELECT DISTINCT u.id,
113     u.views,
114     CASE
115         WHEN u.views >=350 THEN 1
116         WHEN u.views >=100 THEN 2
117         WHEN u.views < 100 THEN 3
118     END AS category
119 FROM stackoverflow.users AS u
120 WHERE u.location LIKE '%United States%' AND
121 views !=0;
122
123 --11.Дополните предыдущий запрос. Отобразите лидеров каждой группы – пользователей, которые
    набрали максимальное число просмотров в своей группе.
124 --Выведите поля с идентификатором пользователя, группой и количеством просмотров.
125 --Отсортируйте таблицу по убыванию просмотров, а затем по возрастанию значения
    идентификатора.
126
127 WITH t AS
128 (SELECT DISTINCT u.id,
129     u.views,
130     CASE
131         WHEN u.views >=350 THEN 1
132         WHEN u.views >=100 THEN 2
133         WHEN u.views < 100 THEN 3
134     END AS category
135 FROM stackoverflow.users AS u
136 WHERE u.location LIKE '%United States%' AND
137 views !=0),
138 maximum AS (
139 SELECT *,
140     MAX(views) OVER (PARTITION BY t.category) AS maxi
141 FROM t )
142
143 SELECT maximum.id,
144     maximum.category,
145     maximum.maxi
146 FROM maximum
147 WHERE maximum.views = maximum.maxi
148 ORDER BY maximum.maxi DESC, maximum.id;
149
150 --12.Посчитайте ежедневный прирост новых пользователей в ноябре 2008 года. Сформируйте
    таблицу с полями:
151 --номер дня;
152 --число пользователей, зарегистрированных в этот день;
153 --сумму пользователей с накоплением.
154
155 WITH t AS (
156 SELECT EXTRACT(DAY FROM creation_date::date) AS days,
157     count(id) AS total_user
158 FROM stackoverflow.users
159 WHERE creation_date::date BETWEEN '2008-11-01' AND '2008-11-30'
160 GROUP BY EXTRACT(DAY FROM creation_date::date))
161
162 SELECT *,
163     SUM(total_user) OVER (ORDER BY days)
164 FROM t;
165
166 --13. Для каждого пользователя, который написал хотя бы один пост, найдите интервал между
    регистрацией и временем создания первого поста. Отобразите:
167 --идентификатор пользователя;
168 --разницу во времени между регистрацией и первым постом.
169
```

```
170 WITH p AS
171 (SELECT DISTINCT user_id,
172      MIN(creation_date) OVER (PARTITION BY user_id) AS min_dt
173 FROM stackoverflow.posts
174 )
175
176 SELECT p.user_id,
177      (p.min_dt - u.creation_date) AS diff
178 FROM stackoverflow.users AS u
179 JOIN p ON u.id = p.user_id;
180
```