

# Dokumentácia k zadaniu 3

**Autor:** Dominik Miškovič  
**Predmet:** Databázové systémy  
**Cvičiaci:** Ing. William Brach  
**Obdobie:** Letný semester 2024

Obsah

<b>1</b>	<b>Endpoint 1</b>	<b>3</b>
1.1	Query . . . . .	3
1.2	Opis fungovania . . . . .	4
1.3	Príklad volania . . . . .	4
<b>2</b>	<b>Endpoint 2</b>	<b>5</b>
2.1	Query . . . . .	5
2.2	Opis fungovania . . . . .	5
2.3	Príklad volania . . . . .	6
<b>3</b>	<b>Endpoint 3</b>	<b>7</b>
3.1	Query . . . . .	7
3.2	Opis fungovania . . . . .	7
3.3	Príklad volania . . . . .	7
<b>4</b>	<b>Endpoint 4</b>	<b>8</b>
4.1	Query . . . . .	8
4.2	Opis fungovania . . . . .	8
4.3	Príklad volania . . . . .	8

# 1 Endpoint 1

## 1.1 Query

```
1  SELECT id, title, type,
2      TO_CHAR(date AT TIME ZONE 'UTC', 'YYYY-MM-DD"T"HH24:MI:SS.FF30F'),
3  CASE
4      WHEN type = 'post' THEN (ROW_NUMBER() OVER ()) - 10
5      ELSE ROW_NUMBER() OVER ()
6  END AS rank
7  FROM (
8      SELECT *,
9      ROW_NUMBER() OVER (PARTITION BY type, post_group ORDER BY date, id) AS rank
10     FROM (
11         SELECT *,
12         LEAD(type) OVER (ORDER BY date) AS next_type,
13         SUM(CASE WHEN type = 'post' THEN 1 ELSE 0 END) OVER (ORDER BY date) AS post_group
14     FROM (
15         SELECT p.id, p.title, p.creationdate AS date, 'post' AS type
16         FROM posts p
17         WHERE p.owneruserid = {user_id}
18
19         UNION ALL
20
21         SELECT b.id, b.name, b.date AS date, 'badge' AS type
22         FROM badges b
23         JOIN (
24             SELECT id, creationdate AS date,
25                 LEAD(creationdate) OVER (ORDER BY creationdate) AS next_post_date
26             FROM posts
27             WHERE owneruserid = {user_id}
28             ) p ON b.date > p.date AND (b.date < p.next_post_date OR p.next_post_date IS NULL)
29             WHERE b.userid = {user_id}
30         ) AS unioned_posts_badges
31     ) AS grouped_posts_badges
32     WHERE NOT (type = 'post' AND next_type = 'post')
33 ) AS ranked_posts_badges
34 WHERE CASE WHEN type = 'badge' THEN rank = 1 ELSE TRUE END
35 ORDER BY date;
36
```

## 1.2 Opis fungovania

### Subquery - unioned\_posts\_badges:

Prvý SELECT získa príspevky používateľa (owneruserid = user\_id). Druhý SELECT získa odznaky používateľa. Je spojený so subquery, ktorý vypočíta next\_post\_date (dátum vytvorenia nasledujúceho príspevku), aby sa stanovil časový rámec na priradenie odznakov k predchádzajúcim príspevkom. Filter (b.date > p.date a (b.date < p.next\_post\_date ...)) zabezpečuje, že odznak je spojený s predchádzajúcim príspevkom len vtedy, ak bol udelený po uverejnení príspevku, pod ktorý spadá.

### Subquery - grouped\_posts\_badges:

LEAD(type) sa pozrie dopredu na nasledujúci riadok na základe usporiadania dátumu a priradí "typ" z tohto nasledujúceho riadku do stĺpca next\_type. SUM(CASE WHEN type = "post" THEN 1 ELSE 0 END) vytvára stĺpec post\_group, ktorý zvyšuje počítadlo zakaždým, keď sa objaví príspevok. Zoskupí sady príspevkov s ich následným odznakom. Následne sa priraduje poradie v rámci každej sady pomocou ROW\_NUMBER() (určené podľa 'type' a 'post\_group'). Toto bude neskôr kľúčové pre filtrovanie príspevkov.

### Subquery - ranked\_posts\_badges:

Tu sa podľa post\_group priradia badgom ranky na základe dátumu ich vytvorenia. Filter NOT (type = 'post' AND next\_type = 'post') odstráni riadky, v ktorých po príspevku nasleduje iný príspevok. Je potrebné vyfiltrovať len sekvencie post -> badge.

### Vonkajší Select:

Vyberú sa príslušné stĺpce a vypočítaný 'rank'. Filter CASE WHEN type = 'badge' THEN rank = 1 ELSE TRUE END zabezpečuje, aby sa vybrali iba odznaky s rankom 1, čiže odznaky udelené ako prvé po danom poste. ORDER BY date zabezpečuje, aby bol konečný výstup chronologicky zoradený.

## 1.3 Príklad volania

```
http://localhost:8000/v3/users/120/badge_history
```

```
{
  "items": [
    {
      "id": 7744,
      "title": "How do I make Firefox remember its window size?",
      "type": "post",
      "created_at": "2009-07-18T03:33:08.597+00",
      "position": 1
    },
    {
      "id": 5453,
      "title": "Student",
      "type": "badge",
      "created_at": "2009-07-18T03:47:30.730+00",
      "position": 1
    },
    {
      "id": 8957,
      "title": null,
      "type": "post",
      "created_at": "2009-07-20T02:27:58.430+00",
      "position": 2
    },
    {
      "id": 6095,
      "title": "Teacher",
      "type": "badge",
      "created_at": "2009-07-20T02:32:30.713+00",
      "position": 2
    }
  ]
}
```

## 2 Endpoint 2

### 2.1 Query

```
1  SELECT post_id, post_title, author_displayname, text,
2         TO_CHAR(p_created_at AT TIME ZONE 'UTC', 'YYYY-MM-DD"THH24:MI:SS.FF30F'),
3         TO_CHAR(created_at AT TIME ZONE 'UTC', 'YYYY-MM-DD"THH24:MI:SS.FF30F'),
4         diff::text,
5         (AVG(diff) OVER (PARTITION BY post_id ORDER BY created_at))::text
6  FROM (
7      SELECT p.id AS post_id, p.title AS post_title,
8             u.displayname AS author_displayname,
9             c.text as text, p.creationdate AS p_created_at,
10            c.creationdate AS created_at,
11            CASE
12                WHEN LAG(c.creationdate) OVER (PARTITION BY c.postid ORDER BY c.creationdate)
13                 IS NULL THEN c.creationdate - p.creationdate
14                ELSE c.creationdate - LAG(c.creationdate) OVER (PARTITION BY c.postid
15                        ORDER BY c.creationdate)
16            END AS diff
17  FROM comments c
18  LEFT JOIN posts p ON c.postid = p.id
19  LEFT JOIN users u ON c.userid = u.id
20  JOIN post_tags pt ON p.id = pt.post_id
21  JOIN tags t ON pt.tag_id = t.id
22  WHERE t.tagname = '{tag}'
23         AND p.commentcount > {count}
24  ) AS sub
25  ORDER BY post_id, created_at;
```

### 2.2 Opis fungovania

#### Subquery:

Začína sa výberom údajov z tabuľky "comments". Spojí sa s tabuľkami "posts" a "users" na získanie názvov príspevkov a informácií o autoroch, ako aj s tabuľkami "post\_tags" a "tags" na filtrovanie výsledkov na základe zadaného tagu. WHERE t.tagname = 'tag' AND p.commentcount > count filter zabezpečí, že sa query zameria len na príspevky s určeným tagom a na tie, ktoré prekročia zadanú hranicu počtu komentárov. LAG(c.creationdate) skúma dátum vytvorenia predchádzajúceho komentára v rámci toho istého príspevku (PARTITION BY c.postid) a umožňuje vypočítať časový rozdiel. Potom sa vypočíta rozdiel medzi dátumom vytvorenia aktuálneho komentára a dátumom vytvorenia príspevku (ak ide o prvý komentár) alebo dátumom vytvorenia predchádzajúceho komentára.

#### Vonkajší Select:

Vyberú sa relevantné stĺpce pre konečný výstup. AVG(diff) OVER (PARTITION BY post\_id ORDER BY created\_at) sa použije na výpočet priebežného priemeru časových rozdielov (diff) v rámci každého setu príspevkov (PARTITION BY post\_id), zoradených podľa času vytvorenia komentára (ORDER BY created\_at). Konečný výsledok sa zoradí podľa času vytvorenia príspevku a komentára.

## 2.3 Příklad volania

http://localhost:8000/v3/tags/networking/comments?count=40

```
▼ "items": [  
  ▼ {  
    "post_id": 1034137,  
    "title": "Did I just get hacked?",  
    "displayname": "Jonno",  
    "text": "Yeah that doesn't look too good. I'm not an expert in Linux by any means, but somethings definitely tried to execute to log in as root and failed. Are there any other logs in your auth.log? Any other means of remote admin? I've seen Mac's like an SSH attempt. Looks like the IPs it was downloading from are hosted in China somewhere.",  
    "post_created_at": "2016-02-01T10:21:48.690+00",  
    "created_at": "2016-02-01T10:25:02.610+00",  
    "diff": "00:03:13.92",  
    "avg": "00:03:13.92"  
  },  
  ▼ {  
    "post_id": 1034137,  
    "title": "Did I just get hacked?",  
    "displayname": "David Schwartz",  
    "text": "The attack actually came from China.",  
    "post_created_at": "2016-02-01T10:21:48.690+00",  
    "created_at": "2016-02-01T10:30:45.310+00",  
    "diff": "00:05:42.7",  
    "avg": "00:04:28.31"  
  },  
  ▼ {  
    "post_id": 1034137,  
    "title": "Did I just get hacked?",  
    "displayname": "vaid",  
    "text": "Yes but what is a Microsoft owned IP doing trying to breach a device across the internet?",  
    "post_created_at": "2016-02-01T10:21:48.690+00",  
    "created_at": "2016-02-01T10:37:58.037+00",  
    "diff": "00:07:12.727",  
    "avg": "00:05:23.115667"  
  },  
]
```

## 3 Endpoint 3

### 3.1 Query

```
1  SELECT id, displayname, body, text, score, position
2  FROM(
3      SELECT ROW_NUMBER() OVER (PARTITION BY p.id ORDER BY c.creationdate) as position,
4      c.id,c.text,c.score, p.body, u.displayname
5      FROM comments c
6      JOIN posts p ON p.id = c.postid
7      JOIN post_tags pt ON p.id = pt.post_id
8      JOIN tags t ON pt.tag_id = t.id
9      JOIN users u ON u.id = c.userid
10     WHERE t.tagname = '{tag}'
11     ORDER BY p.creationdate
12     ) AS sub
13 WHERE position = {position}
14 LIMIT {limit};
15
```

### 3.2 Opis fungovania

#### Subquery:

Začína sa výberom údajov z tabuľky "comments". Spojí sa s tabuľkami "posts" a "users" na získanie názvov príspevkov a informácií o autoroch, ako aj s tabuľkami "post\_tags" a "tags" na filtrovanie výsledkov na základe zadaného tagu. ROW\_NUMBER() priradí každému komentáru v rámci príspevku poradové číslo ("pozíciu"). To nám hovorí, či je to 1., 2., 3. komentár atď. pre daný príspevok. WHERE t.tagname = '{tag}' filter zabezpečí, že sa query zameria len na príspevky so zadaným tagom.

#### Vonkajší Select:

Vyberú sa relevantné stĺpce pre konečný výstup. WHERE position = {position} filtruje výsledky tak, aby zahrňali len komentáre s požadovanou hodnotou "pozície", ktorá sa zadala. LIMIT {limit} obmedzí výstup na zadaný počet výsledkov.

### 3.3 Príklad volania

```
http://localhost:8000/v3/tags/linux/comments/2?limit=1
```

```
{
  "items": [
    {
      "id": 745427,
      "displayname": "Oliver Salzburg",
      "body": "<p>I am running Kubuntu Hardy Heron, with a dual monitor setup, and have VirtualBox on it running Windows XP in sear\nthe second monitor. \nHow can this be achieved?</p>\n",
      "text": "http://ubuntuforums.org/showthread.php?t=433359",
      "score": 0,
      "position": 2
    }
  ]
}
```

## 4 Endpoint 4

### 4.1 Query

```
1 SELECT name, STRING_AGG(thread_posts.body, ''),
2 TO_CHAR(creationdate AT TIME ZONE 'UTC', 'YYYY-MM-DD"T"HH24:MI:SS.FF30F')
3 FROM (
4     SELECT p.id, p.parentid,
5           (SELECT displayname FROM users WHERE p.owneruserid = users.id) AS name,
6           p.body, p.creationdate
7     FROM posts p
8     WHERE p.id = {post_id}
9
10    UNION
11
12    SELECT p.id, p.parentid,
13          (SELECT displayname FROM users WHERE p.owneruserid = users.id) AS name,
14          p.body, p.creationdate
15    FROM posts p
16    JOIN (SELECT p.id
17          FROM posts p
18          WHERE p.id = {post_id}
19        ) AS tp ON p.parentid = tp.id
20    ) AS thread_posts
21 GROUP BY thread_posts.creationdate, name
22 ORDER BY creationdate ASC
23 LIMIT {limit};
24
```

### 4.2 Opis fungovania

#### Subquery:

Má dve časti spojené pomocou UNION. Prvý SELECT nájde počiatočný príspevok na základe jeho zadaného ID a získa základné informácie o príspevku. Druhý SELECT nájde odpovede. Spája sa so subquery, ktoré vyberie id počiatočného príspevku. ON p.parentid = tp.id je kľúčové prepojenie, ktoré nájde príspevky, ktorých "parentid" sa zhoduje s ID príspevku z predchádzajúceho kroku.

#### Vonkajší Select:

Vyberú sa relevantné stĺpce pre konečný výstup. STRING\_AGG funkcia spojí obsah "body" všetkých relevantných príspevkov a vytvorí jeden text pre celú konverzáciu. GROUP BY thread\_posts.creationdate, name zoskupí výsledky podľa dátumu vytvorenia a mena. ORDER BY creationdate ASC usporiada výsledky chronologicky

### 4.3 Príklad volania

```
http://localhost:8000/v3/posts/2154?limit=2
```

```
{
  "items": [
    {
      "displayname": "Eugene M",
      "body": "<p>So, I'm a technology guy and sometimes I have to troubleshoot a home network, including my own. I make sure the internet connection. Usually after that point I just reset the router( and possibly the cable modem) and that fixes things n issue I could possibly be fixing by resetting the router.</p>\n\n<p>EDIT: Just to clarify, I was speaking more about reset a reset(paperclip in the hole) is useful. So the more accurate term would probably be restarting </p>\n\n<p>Also, personally I only bother messing around with stuff if I can't make a connection to the internet at all.</p>\n",
      "created_at": "2009-07-15T12:51:57.340+00"
    },
    {
      "displayname": "Ólafur Waage",
      "body": "<p>Every router has it's original firmware stored somewhere on it.</p>\n\n<p>When you reset the router you overwrite fixing the problem is that the config is overwritten with the original one. But in some cases you have an updated router tha",
      "created_at": "2009-07-15T12:54:48.507+00"
    }
  ]
}
```