



Modern Database HW03 Report

Assignment	
Author	Alexander Powers

Homework 03 Postgres Coffee Reviews

This homework focused on text searching in postgres. Woot woot!

Coffe Query Setup:

```
-- create prod_review table
CREATE TABLE prod_review (
    productid character(20),
    allreviews text,
    numreviews int,
    processed text,
    metaphone_processed text,
    wordlist tsvector,
    tags varchar []
);
-- start populating prod_review table
INSERT INTO prod_review (productid, numreviews, allreviews)
SELECT productid,
    count(*) AS numreviews,
    STRING_AGG (
        r.summary || ' ' || r.review,
        ','
    )
    ORDER BY r.summary,
        r.review
) AS allreviews
FROM reviews r
GROUP BY productid;
UPDATE prod_review
SET processed = REGEXP_REPLACE(allreviews, '[^\w\s]', ' ', 'g');
UPDATE prod_review
SET metaphone_processed = REGEXP_REPLACE(processed, '[^\w\s]', ' ', 'g');
-- create and configure dictionary
CREATE TEXT SEARCH DICTIONARY simple_english (
    template = snowball,
    language = english,
    stopwords = english
);
CREATE TEXT SEARCH CONFIGURATION simple_english (copy = english);
-- create wordlist
UPDATE prod_review
SET wordlist = to_tsvector('simple_english', processed);
-- create word stats
CREATE TABLE word_stats AS
SELECT *
FROM ts_stat('SELECT wordlist FROM prod_review')
ORDER BY ndoc DESC,
```

```

nentry DESC,
word;
-- creat tags
UPDATE prod_review
SET tags = ARRAY(
    SELECT word
    FROM ts_stat(
        'SELECT wordlist
        FROM prod_review psd
        WHERE productid = ''' || productid || '''
    )
)
ORDER BY nentry DESC,
ndoc DESC,
word
LIMIT 5
);
-- create coffee_prods table
CREATE TABLE coffee_prods (productid character(20), allreviews text);

```

Queries:

1. Query to populate `coffee_prods`

```

-- query for coffee related products store in coffee_prods
INSERT INTO coffee_prods (productid, allreviews)
SELECT productid,
allreviews
FROM prod_review
WHERE to_tsvector(tags::text) @@ to_tsquery('english', '(coffe | coffee | decaf) & ! tea');

```

Resulted in `442` coffee products

2. Task 2 Query

```

-- coffe avg query
SELECT ageGroup,
gender,
SUM(reviewCount) AS review_count,
AVG(score) as avg_rating
FROM coffee_prods
INNER JOIN reviews ON (reviews.productid = coffee_prods.productid)
INNER JOIN user_stats ON (user_stats.userid = reviews.userid)
INNER JOIN users ON (users.userid = user_stats.userid)
GROUP BY ageGroup,
gender;

-- all products avg query
SELECT ageGroup,
gender,
SUM(reviewCount) AS review_count,
AVG(score) as avg_rating
FROM reviews
INNER JOIN user_stats ON (user_stats.userid = reviews.userid)
INNER JOIN users ON (users.userid = user_stats.userid)
GROUP BY ageGroup,
gender;

```

Coffee Result:

Query Results			
group1	gender	review_count	avg_rating
group1	Male	1242	3.4243733034462828
group2	Male	1057	4.07428717423514
group1	Female	1284	4.4128325304136253
group2	Female	1683	4.2653283020577899
group3	Male	1027	3.3882890864657701
group3	Female	1568	4.0823442895851796
6 rows(s)		Total runtime: 58.237 ms	
SQL executed:		Edit SQL Download	

All Products Result:

Query Results					
agegroup	gender	review_count	avg_rating		
group2	Male	9549	4.1803349076057020		
group3	Male	6418	4.13036565536524757		
group1	Female	10250	4.06512765120298167		
group2	Female	33524	4.239367886070098		
group3	Female	10965	4.123943135461568		
group3	Male	9078	4.1203033588613.139		
				6 (row(s))	
				Total runtime: 35.392 ms	
				SQL executed:	
				Edit SQL	Download

3. Task 3 Query

```
-- group by dposted year month
select *
from crosstab(
    'SELECT date_part(''year'', dposted) AS year, date_part(''month'', dposted) AS month, SUM(id) AS review_count
     FROM reviews
    GROUP BY date_part(''year'', dposted), date_part(''month'', dposted)
   ORDER BY 1',
    'select m from generate_series(1,12) m'
) as (
    year int,
    "Jan" int,
    "Feb" int,
    "Mar" int,
    "Apr" int,
    "May" int,
    "Jun" int,
    "Jul" int,
    "Aug" int,
    "Sep" int,
    "Oct" int,
    "Nov" int,
    "Dec" int
);

```

Result:

PostgreSQL 10.12 running on 127.0.0.1:5432 -- You are logged in as user "student01"

phpPgAdmin: PostgreSQL s-l112: student01

Query Results

year	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2007	NULL	NULL	NULL	NULL	NULL	76882						
2008	NULL	NULL	NULL	NULL	2489	NULL	NULL	NULL	NULL	NULL	NULL	NULL
2011	NULL	NULL	190954	94281	183132	8111						
2012	157566	14832	85354	59436	123428	301111	274620	84327	96104	243670	507266	235674
2013	298312	184456	326788	225780	586036	276565	669368	798673	184546	679099	973415	917592
2014	926154	590970	499967	958764	1884382	3621210	1824679	1550927	4726520	2330409	8954991	4656518
2015	5782742	4865146	6456996	5114149	4423546	4844637	5719089	5787324	7022900	7555661	8905850	9090817
2016	7798066	7944842	12952134	8038935	9115537	9199128	17847783	8909732	11495729	15175997	14929852	17201275
2017	41400984	20064021	16797521	29883686	16326439	13659977	21605485	20330424	17643073	18131822	19070099	27178654
2018	21693477	17962989	19397100	19708533	23431638	22268396	28337625	32594757	29248021	32139999	34807290	37533407
2019	43301178	35454966	38140090	39106540	51614218	69556526	74410617	87674000	65134876	74811890	76916512	80878235
2020	88632647	67870591	76263079	76451274	77056596	93029432	104417775	95686483	22777573	NULL	NULL	NULL

12 row(s)

Total runtime: 36.907 ms

SQL executed.

[Edit SQL](#) | [Download](#)