生成需要进行 vacuum 表工具

vacuum 需要整理表脚本生成工具

工具目的： 产生根据参数限定后的POSTGRESQL VACUUM 命令

参数 a 为死元组占整体表的百分比

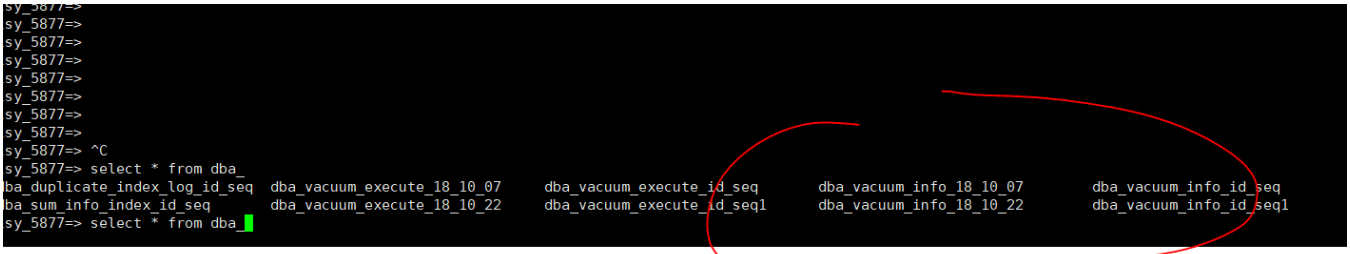
参数b 为多少行表以上才进行记录

同时将死元组和表行数为0的表直接过滤，执行后产生 dba_vacuum_execute ，此表为最终结果表，产生进行 vacuum 的命令

```
lsy_5877=> select * from dba_vacuum_execute_
dba_vacuum_execute_18_10_07  dba_vacuum_execute_18_10_22  dba_vacuum_execute_id_seq    dba_vacuum_execute_id_seq1
lsy_5877=> select * from dba_vacuum_execute_18_10_22 ;
 id |           sql            | schemaname |           tablename
----+--------------------------+------------+-------------------------------
  1 | vacuum (verbose,analyze) | public     | dbi_duty
  2 | vacuum (verbose,analyze) | public     | bm_marketing_plan
  3 | vacuum (verbose,analyze) | public     | dbi_keyword
  4 | vacuum (verbose,analyze) | public     | arch_color
  5 | vacuum (verbose,analyze) | public     | arch_item_class_method
  6 | vacuum (verbose,analyze) | public     | shop_desk_setting
  7 | vacuum (verbose,analyze) | public     | sys_o2o_archm_update
  8 | vacuum (verbose,analyze) | public     | bm_commend_item
  9 | vacuum (verbose,analyze) | public     | sys_item_most_unit_update
 10 | vacuum (verbose,analyze) | public     | arch_payway
 11 | vacuum (verbose,analyze) | public     | takeout_orders_delivery_config
 12 | vacuum (verbose,analyze) | public     | bm_mp_pp_config
 13 | vacuum (verbose,analyze) | public     | dbi_duty_bs
 14 | vacuum (verbose,analyze) | public     | rpt_common_name
 15 | vacuum (verbose,analyze) | public     | biz_code_rule
 16 | vacuum (verbose,analyze) | public     | biz_error_log
 17 | vacuum (verbose,analyze) | public     | bm_maxim
```

dba_vacuum_info 为中间表，其中包含整体的表的，插入，UPDATE ,DELETE 以及行数和死元组，以及最后一次vacuum 和 analyze 的时间信息

```
id            | 116
schemaname    | public
tablename     | dbi_duty
n_tup_ins     | 60
n_tup_upd     | 0
n_tup_del     | 48
n_live_tup    | 12
n_dead_tup    | 48
reltuples     | 12
av_threshold  | 50
last_vacuum   |
last_analyze  |
pct_dead      | 400
create_time   | 2022-11-07 18:10:07.562587
-[ RECORD 2 ]+------------------------------
id            | 117
schemaname    | public
tablename     | bm_marketing_plan
n_tup_ins     | 55
n_tup_upd     | 0
n_tup_del     | 44
n_live_tup    | 11
n_dead_tup    | 44
reltuples     | 11
av_threshold  | 50
last_vacuum   |
last_analyze  |
pct_dead      | 400
create_time   | 2022-11-07 18:10:07.562681
-[ RECORD 3 ]+------------------------------
id            | 118
schemaname    | public
tablename     | dbi_keyword
n_tup_ins     | 50
n_tup_upd     | 0
n_tup_del     | 40
n_live_tup    | 10
n_dead_tup    | 40
reltuples     | 10
av_threshold  | 50
last_vacuum   |
last_analyze  |
```



代码：

```
CREATE OR REPLACE FUNCTION autovacuum_monitor (a int,b int)
RETURNS INTEGER AS $$
DECLARE
var_sql text;

BEGIN



create table if not exists dba_vacuum_info
(id serial,
schemaname varchar(100),
tablename varchar(100),
n_tup_ins int,
n_tup_upd int,
n_tup_del int,
n_live_tup int,
n_dead_tup int,
reltuples int,
av_threshold int,
last_vacuum timestamp,
last_analyze timestamp,
pct_dead int,
create_time timestamp,
primary key (id)
);

truncate table dba_vacuum_info;


create table if not exists dba_vacuum_execute
(id serial,
sql text,
schemaname varchar(100),
tablename varchar(100),

primary key (id));

truncate table dba_vacuum_execute;
```

```
var_sql = 'insert into dba_vacuum_info (schemaname,tablename,n_tup_ins,n_tup_upd,n_tup_del,n_live_tup,n_dead_tup,reltuples,av_threshold,
last_vacuum,last_analyze,pct_dead,create_time)
SELECT nspname,relname,n_tup_ins,n_tup_upd,n_tup_del,n_live_tup,n_dead_tup,reltuples,av_threshold,last_vacuum,last_analyze,
CASE
WHEN reltuples > 0
THEN round(100.0 * n_dead_tup / (reltuples))
ELSE 0
END AS pct_dead,
clock_timestamp()
FROM (
SELECT N.nspname
,C.relname
,pg_stat_get_tuples_inserted(C.oid) AS n_tup_ins
,pg_stat_get_tuples_updated(C.oid) AS n_tup_upd
,pg_stat_get_tuples_deleted(C.oid) AS n_tup_del
,pg_stat_get_live_tuples(C.oid) AS n_live_tup
,pg_stat_get_dead_tuples(C.oid) AS n_dead_tup
,C.reltuples AS reltuples
,round(current_setting('||'''autovacuum_vacuum_threshold'''||')::INTEGER + current_setting('||'''autovacuum_vacuum_scale_factor'''||')::
NUMERIC * C.reltuples) AS av_threshold
,date_trunc('||'''minute'''||', greatest(pg_stat_get_last_vacuum_time(C.oid), pg_stat_get_last_autovacuum_time(C.oid))) AS last_vacuum
,date_trunc('||'''minute'''||', greatest(pg_stat_get_last_analyze_time(C.oid), pg_stat_get_last_analyze_time(C.oid))) AS last_analyze
FROM pg_class C
LEFT JOIN pg_namespace N ON (N.oid = C.relnamespace)
WHERE C.relkind IN (
'|| '''r'''
||','||'''t'''||'
)
AND N.nspname NOT IN ('||
'''pg_catalog'''||
','||'''information_schema'''||'
)
AND N.nspname not like '||''''pg_toast%'''||
') AS av
where n_dead_tup <> 0 and reltuples <> 0 and round(100.0 * n_dead_tup / (reltuples)) >'|| a || 'and reltuples >' || b ||
' ORDER BY n_dead_tup DESC; ';


execute var_sql;

var_sql = 'insert into dba_vacuum_execute (sql,schemaname,tablename) select' || '''vacuum (verbose,analyze)''' || ',schemaname,tablename
from dba_vacuum_info;';

execute var_sql;

var_sql = 'alter table dba_vacuum_execute rename to ' || '"dba_vacuum_execute_' || to_char(current_timestamp,'HH24_MI_SS')||'";';

execute var_sql;

var_sql = 'alter table dba_vacuum_info rename to ' || '"dba_vacuum_info_' || to_char(current_timestamp,'HH24_MI_SS')||'";';
execute var_sql;


return 1;

END ;
$$ LANGUAGE plpgsql;

select autovacuum_monitor(1,1);
```