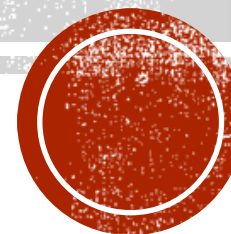


• پروژه انبار داده با شمای ستاره ای

علیرضا عباسی نوخندان ، یونس محمودی (Group 20)



```
Applications Places Terminal Dec 18 18:40 userg20@kaliou: /

Please make sure you have the correct access rights
and the repository exists.

(userg20@kaliou)-[/]
$ sudo git clone git@github.com:vadimtk/ssb-dbgen.git
Cloning into 'ssb-dbgen'...
git@github.com: Permission denied (publickey).
fatal: Could not read from remote repository.

Please make sure you have the correct access rights
and the repository exists.

(userg20@kaliou)-[/]
$ sudo apt-key adv --keyserver keyserver.ubuntu.com --recv E0C56BD4
Warning: apt-key is deprecated. Manage keyring files in trusted.gpg.d instead (see apt-key(8)).
Executing: /tmp/apt-key-gpghome.OR4qzW4g4j/gpg.1.sh --keyserver keyserver.ubuntu.com --recv E0C56BD4
gpg: key C8F1E19FE0C56BD4: public key "ClickHouse Repository Key <milovidov@yandex-team.ru>" imported
gpg: Total number processed: 1
gpg: imported: 1

(userg20@kaliou)-[/]
$ echo "deb http://repo.yandex.ru/clickhouse/deb/stable/ main/" | sudo tee /etc/apt/sources.list.d/clickhouse.
list
deb http://repo.yandex.ru/clickhouse/deb/stable/ main/

(userg20@kaliou)-[/]
$ sudo apt-get update
```

کلید را تعریف و در مسیر
etc/apt/source.list.d/clickhouse
ریپاسیتوری را ادد می کنیم.




```
Applications Places Terminal Dec 18 18:42 userg20@kali: /
d out
W: Some index files failed to download. They have been ignored, or old ones used instead.

(userg20@kali)-[/]
$ sudo apt-get install -y clickhouse-server clickhouse-client
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
 gir1.2-clutter-gst-3.0 gir1.2-gtkclutter-1.0 libamtk-5-0 libamtk-5-common libcamel-1.2-62 libdap27
 libdapclient6v5 libdav1d4 libdataserver-1.2-25 libdataserverui-1.2-2 libepsilon1
 libextutils-pkgconfig-perl libgdal28 libgeos-3.9.1 libgupnp-1.2-0 libidn11 libmusicbrainz5-2
 libmusicbrainz5cc2v5 libnetcdf18 libntfs-3g883 libomp-11-dev libomp5-11 libproj19 libqt5waylandcompositor5
 libquvi-0.9-0.9.3 libquvi-scripts-0.9 libtepl-5-0 libtracker-control-2.0-0 libtracker-miner-2.0-0
 libtracker-sparql-2.0-0 liburcu6 libwireplumber-0.4-0 libx265-192 libxmlb1 libyara4 lua-bitop lua-expat
 lua-json lua-socket python3-editor python3-exif python3-ipython-genutils python3-pylnk python3-stem
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
 clickhouse-common-static
Suggested packages:
 clickhouse-common-static-dbg
The following NEW packages will be installed:
 clickhouse-client clickhouse-common-static clickhouse-server
0 upgraded, 3 newly installed, 0 to remove and 776 not upgraded.
Need to get 208 MB of archives.
After this operation, 680 MB of additional disk space will be used.
Get:1 http://repo.yandex.ru/clickhouse/deb/stable main/ clickhouse-common-static 21.12.2.17 [208 MB]
11% [1 clickhouse-common-static 28.3 MB/208 MB 14%] 3,354 kB/s 53s
```

در اینجا کلیک هاوس را نصب می کنیم.




```
Applications Places Terminal Dec 18 18:45 userg20@kaliou: /
Processing triggers for kali-menu (2021.4.1) ...

(userg20@kaliou)-[/]
$ sudo service clickhouse-server start

(userg20@kaliou)-[/]
$ sudo service clickhouse-server status
● clickhouse-server.service - ClickHouse Server (analytic DBMS for big data)
   Loaded: loaded (/etc/systemd/system/clickhouse-server.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2021-12-18 18:44:46 GMT; 9s ago
 Main PID: 4693 (clickhouse-watch)
    Tasks: 207 (limit: 14122)
   Memory: 134.8M
      CPU: 826ms
   CGroup: /system.slice/clickhouse-server.service
           └─4693 clickhouse-watchdog "" "" "" "" "" "" "" --config=/etc/clickhouse-server/config.xml --pid-f
             └─4694 /usr/bin/clickhouse-server --config=/etc/clickhouse-server/config.xml --pid-file=/run/click

Dec 18 18:44:46 kaliou systemd[1]: Started ClickHouse Server (analytic DBMS for big data).
Dec 18 18:44:46 kaliou clickhouse-server[4693]: Processing configuration file '/etc/clickhouse-server/config.x
Dec 18 18:44:46 kaliou clickhouse-server[4693]: Logging trace to /var/log/clickhouse-server/clickhouse-server.
Dec 18 18:44:46 kaliou clickhouse-server[4693]: Logging errors to /var/log/clickhouse-server/clickhouse-server
Dec 18 18:44:46 kaliou clickhouse-server[4694]: Processing configuration file '/etc/clickhouse-server/config.x
Dec 18 18:44:46 kaliou clickhouse-server[4694]: Saved preprocessed configuration to '/var/lib/clickhouse/prepr
Dec 18 18:44:46 kaliou clickhouse-server[4694]: Processing configuration file '/etc/clickhouse-server/users.xml
Dec 18 18:44:46 kaliou clickhouse-server[4694]: Merging configuration file '/etc/clickhouse-server/users.d/def
Dec 18 18:44:46 kaliou clickhouse-server[4694]: Saved preprocessed configuration to '/var/lib/clickhouse/prepr
lines 1-20/20 (END)
```

سرویس کلیک هاوس را استارت و
برای اطمینان حاصل شدن از نصب
کامل و صحیح **clickhouse** باید از
دستور زیر استفاده کنیم:

```
sudo service  
clickhouse-server  
status
```

تا با پیغام زیر روبرو شویم که نشان
clickhouse دهنده نصب درست
است :

Active (running)



پس از نصب کلیک هاوس اکنون
ریپاسیتوری مربوطه را از گیت کلون
می کنیم

```
(userg20@kalious)-[/]  
$ sudo git clone https://github.com/vadimtk/ssb-dbgen.git  
Cloning into 'ssb-dbgen'...  
remote: Enumerating objects: 50, done.  
remote: Total 50 (delta 0), reused 0 (delta 0), pack-reused 50  
Receiving objects: 100% (50/50), 96.51 KiB | 457.00 KiB/s, done.  
Resolving deltas: 100% (12/12), done.
```

```
(userg20@kalious)-[/]  
$
```




```
Applications Places Terminal Dec 18 18:54 userg20@kaliou: /ssb-dbgen
(userg20@kaliou)-[/ssb-dbgen]
$ sudo make
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o build.o build.c
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o driver.o driver.c
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o bm_utils.o bm_utils.c
bm_utils.c: In function 'yes_no':
bm_utils.c:133:9: warning: implicit declaration of function 'gets'; did you mean 'fgets'? [-Wimplicit-function-declaration]
133 |         gets(reply);
    |         ^~~~~
    |         fgets
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o rnd.o rnd.c
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o print.o print.c
print.c: In function 'dbg_print':
print.c:138:48: warning: cast from pointer to integer of different size [-Wpointer-to-int-cast]
138 |         fprintf(target, \"%c \", (char)data);
    |                                   ^
print.c:140:47: warning: cast from pointer to integer of different size [-Wpointer-to-int-cast]
140 |         fprintf(target, \"%c\", (char)data);
    |                                   ^
In file included from print.c:18:
print.c: In function 'pr_line':
dss.h:510:62: warning: cast to pointer from integer of different size [-Wint-to-pointer-cast]
510 | #define PR_INT(f, str)          dbg_print(DT_INT, f, (void *)str, 0, 1)
    |                                   ^
print.c:265:9: note: in expansion of macro 'PR_INT'
265 |         PR_INT(fp_l, o->lineorders[i].linenumber);
    |         ^~~~~
```

پس از مراحل فوق باید با استفاده از دستور **make** در دایرکتوری **dbgen** فایل را ایجاد کنیم.




```

Applications Places Terminal Dec 18 18:53
userg20@kali: /ssb-dbgen

/usr/bin/ld: bm_utils.o: in function `yes_no':
bm_utils.c:(.text+0x4c): warning: the `gets' function is dangerous and should not be used.
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o qgen.o qgen.c
qgen.c: In function `process_options':
qgen.c:275:19: warning: implicit declaration of function `getopt' [-Wimplicit-function-declaration]
275 | while((flag = getopt(cnt, args, "ab:cdhi:n:Nl:o:p:r:s:t:vx")) != -1)
    |                  ^~~~~~
qgen.c: At top level:
qgen.c:388:1: warning: return type defaults to `int' [-Wimplicit-int]
388 | main(int ac, char **av)
    | ^~~~~
In file included from dss.h:48,
                 from qgen.c:20:
qgen.c: In function `main':
config.h:165:25: warning: implicit declaration of function `getpid' [-Wimplicit-function-declaration]
165 | #define DSS_PROC      getpid()
    |                        ^~~~~~
qgen.c:409:54: note: in expansion of macro `DSS_PROC'
409 |         rndm = (long)((unsigned)time(NULL) * DSS_PROC);
    |                                              ^~~~~~
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -c -o varsub.o varsub.c
gcc -O -DDBNAME=\"dss\" -DLINUX -DDB2 -DSSBM -O -o qgen build.o bm_utils.o qgen.o rnd.o varsub.o text.o bcd2.o
permute.o speed_seed.o -lm
/usr/bin/ld: bm_utils.o: in function `yes_no':
bm_utils.c:(.text+0x4c): warning: the `gets' function is dangerous and should not be used.

(userg20@kali) - [/ssb-dbgen]
$

```

سپس پس از مراحل فوق باید با
 استفاده از دستور **make** در
 دایرکتوری **dbgen** فایل را ایجاد
 کنیم.




```
Dec 18 18:56
Applications Places Terminal
userg20@kali: /ssb-dbgen

(userg20@kali)~-[/ssb-dbgen]
$ ./dbgen -s 1 -T c
SSBM (Star Schema Benchmark) Population Generator (Version 1.0.0)
Copyright Transaction Processing Performance Council 1994 - 2000
Open failed for ./customer.tbl at bm_utils.c:405

(userg20@kali)~-[/ssb-dbgen]
$ sudo ./dbgen -s 1 -T c
SSBM (Star Schema Benchmark) Population Generator (Version 1.0.0)
Copyright Transaction Processing Performance Council 1994 - 2000

(userg20@kali)~-[/ssb-dbgen]
$ sudo ./dbgen -s 1 -T l
SSBM (Star Schema Benchmark) Population Generator (Version 1.0.0)
Copyright Transaction Processing Performance Council 1994 - 2000

(userg20@kali)~-[/ssb-dbgen]
$ sudo ./dbgen -s 1 -T p
SSBM (Star Schema Benchmark) Population Generator (Version 1.0.0)
Copyright Transaction Processing Performance Council 1994 - 2000

(userg20@kali)~-[/ssb-dbgen]
$ sudo ./dbgen -s 1 -T s
SSBM (Star Schema Benchmark) Population Generator (Version 1.0.0)
Copyright Transaction Processing Performance Council 1994 - 2000
```

برای وارد کردن داده ها در جداول در خط فرمان دستورات زیر را وارد میکنیم:

```
$ ./dbgen -s 1000 -T c
$ ./dbgen -s 1000 -T l
$ ./dbgen -s 1000 -T p
$ ./dbgen -s 1000 -T s
```

این کدها برای قرار دادن مقادیر و داده ها در جدول است.

عدد **s 1000** - نشان دهنده آن است که جدول ایجاد شده دارای شش میلیارد تاپل است؛ که میتوان برای جلوگیری از اتلاف فضا از اعداد کوچکتری استفاده میکنیم. (ما از عدد **s 1** - استفاده کردیم که میتواند جداولی با حداکثر شش میلیون تاپل ایجاد کند).

حروف لاتین آخر هر لاین نشان دهنده نام هر جدول است که با حرف اول نام آن جدول مشخص شده اند.




```
Applications Places Terminal Dec 18 19:06
userg20@kali: /ssb-dbggen

C_NATION LowCardinality(String),
C_REGION LowCardinality(String),
C_PHONE String,
C_MKTSEGMENT LowCardinality(String)
)
ENGINE = MergeTree ORDER BY (C_CUSTKEY);

CREATE TABLE customer
(
  `C_CUSTKEY` UInt32,
  `C_NAME` String,
  `C_ADDRESS` String,
  `C_CITY` LowCardinality(String),
  `C_NATION` LowCardinality(String),
  `C_REGION` LowCardinality(String),
  `C_PHONE` String,
  `C_MKTSEGMENT` LowCardinality(String)
)
ENGINE = MergeTree
ORDER BY C_CUSTKEY

Query id: b7b7ddb2-7096-4354-b7d4-0348b04a5dc1

Ok.

0 rows in set. Elapsed: 0.070 sec.

kali:~$
```

ایجاد جدول customer



```
`LO_LINENUMBER` UInt8,  
`LO_CUSTKEY` UInt32,  
`LO_PARTKEY` UInt32,  
`LO_SUPPKEY` UInt32,  
`LO_ORDERDATE` Date,  
`LO_ORDERPRIORITY` LowCardinality(String),  
`LO_SHIPPRIORITY` UInt8,  
`LO_QUANTITY` UInt8,  
`LO_EXTENDEDPRICE` UInt32,  
`LO_ORDTOTALPRICE` UInt32,  
`LO_DISCOUNT` UInt8,  
`LO_REVENUE` UInt32,  
`LO_SUPPLYCOST` UInt32,  
`LO_TAX` UInt8,  
`LO_COMMITDATE` Date,  
`LO_SHIPMODE` LowCardinality(String)  
)  
ENGINE = MergeTree  
PARTITION BY toYear(LO_ORDERDATE)  
ORDER BY (LO_ORDERDATE, LO_ORDERKEY)
```

Query id: 026d7597-29a3-4b88-a9ce-e853a2679e28

Ok.

0 rows in set. Elapsed: 0.085 sec.

kali:~\$

ایجاد جدول customer




```
Applications  Places  Terminal  Dec 18 19:06  userg20@kali: /ssb-dbgm
P_TYPE      LowCardinality(String),
P_SIZE      UInt8,
P_CONTAINER  LowCardinality(String)
)
ENGINE = MergeTree ORDER BY P_PARTKEY;

CREATE TABLE part
(
  `P_PARTKEY` UInt32,
  `P_NAME` String,
  `P_MFGR` LowCardinality(String),
  `P_CATEGORY` LowCardinality(String),
  `P_BRAND` LowCardinality(String),
  `P_COLOR` LowCardinality(String),
  `P_TYPE` LowCardinality(String),
  `P_SIZE` UInt8,
  `P_CONTAINER` LowCardinality(String)
)
ENGINE = MergeTree
ORDER BY P_PARTKEY

Query id: 59e97bd0-3ee9-4c8b-87a0-6244eee30fb0

Ok.

0 rows in set. Elapsed: 0.071 sec.

kali:~$
```

ایجاد جدول part



```
Applications Places Terminal Dec 18 19:06 userg20@kali: /ssb-dbggen

S_ADDRESS String,
S_CITY LowCardinality(String),
S_NATION LowCardinality(String),
S_REGION LowCardinality(String),
S_PHONE String
)
ENGINE = MergeTree ORDER BY S_SUPPKEY;

CREATE TABLE supplier
(
`S_SUPPKEY` UInt32,
`S_NAME` String,
`S_ADDRESS` String,
`S_CITY` LowCardinality(String),
`S_NATION` LowCardinality(String),
`S_REGION` LowCardinality(String),
`S_PHONE` String
)
ENGINE = MergeTree
ORDER BY S_SUPPKEY

Query id: a7bfbb18-64a4-4353-8d5b-520e2d7d43cc

Ok.

0 rows in set. Elapsed: 0.069 sec.

kallious :)
```

ایجاد جدول **supplier**




```
Applications Places Terminal Dec 18 19:18 userg20@kaliouss: /ssb-dbgen
(userg20@kaliouss)-[/ssb-dbgen]
$ cd ..

(userg20@kaliouss)-[/]
$ sudo nano /etc/clickhouse-server/users.d/default-password.xml

(userg20@kaliouss)-[/]
$ cd ssb-dbgen/

(userg20@kaliouss)-[/ssb-dbgen]
$ clickhouse-client --password=[REDACTED] --query "INSERT INTO customer FORMAT CSV" < customer.tbl

(userg20@kaliouss)-[/ssb-dbgen]
$ clickhouse-client --password=[REDACTED] --query "INSERT INTO part FORMAT CSV" < part.tbl

(userg20@kaliouss)-[/ssb-dbgen]
$ clickhouse-client --password=[REDACTED] --query "INSERT INTO supplier FORMAT CSV" < supplier.tbl
Code: 552. DB::Exception: Unrecognized option '--password=parkourshad8094'. (UNRECOGNIZED_ARGUMENTS)

(userg20@kaliouss)-[/ssb-dbgen]
$ clickhouse-client --password=[REDACTED] --query "INSERT INTO supplier FORMAT CSV" < supplier.tbl

(userg20@kaliouss)-[/ssb-dbgen]
$ clickhouse-client --password=[REDACTED] --query "INSERT INTO lineorder FORMAT CSV" < lineorder.tbl

(userg20@kaliouss)-[/ssb-dbgen]
$
```

در این مرحله داده را از فایل هایی
با فرمت **CSV** خوانده و در جداول
مربوطه درج می کنیم.



```
Applications Places Terminal Dec 21 11:47 younesious@kali: ~  
CREATE TABLE lineorder_flat  
ENGINE = MergeTree  
PARTITION BY toYear(LO_ORDERDATE)  
ORDER BY (LO_ORDERDATE, LO_ORDERKEY) AS  
SELECT  
  l.LO_ORDERKEY AS LO_ORDERKEY,  
  l.LO_LINENUMBER AS LO_LINENUMBER,  
  l.LO_CUSTKEY AS LO_CUSTKEY,  
  l.LO_PARTKEY AS LO_PARTKEY,  
  l.LO_SUPPKEY AS LO_SUPPKEY,  
  l.LO_ORDERDATE AS LO_ORDERDATE,  
  l.LO_ORDERPRIORITY AS LO_ORDERPRIORITY,  
  l.LO_SHIPPRIORITY AS LO_SHIPPRIORITY,  
  l.LO_QUANTITY AS LO_QUANTITY,  
  l.LO_EXTENDEDPRICE AS LO_EXTENDEDPRICE,  
  l.LO_ORDTOTALPRICE AS LO_ORDTOTALPRICE,  
  l.LO_DISCOUNT AS LO_DISCOUNT,  
  l.LO_REVENUE AS LO_REVENUE,  
  l.LO_SUPPLYCOST AS LO_SUPPLYCOST,  
  l.LO_TAX AS LO_TAX,  
  l.LO_COMMITDATE AS LO_COMMITDATE,  
  l.LO_SHIPMODE AS LO_SHIPMODE,  
  c.C NAME AS C NAME,
```

شمای جداول را از استار به فلت
کانورت می کنیم.




```
Applications Places Terminal Dec 21 11:47 younesious@kali: ~  
1.LO_COMMITDATE AS LO_COMMITDATE,  
1.LO_SHIPMODE AS LO_SHIPMODE,  
c.C_NAME AS C_NAME,  
c.C_ADDRESS AS C_ADDRESS,  
c.C_CITY AS C_CITY,  
c.C_NATION AS C_NATION,  
c.C_REGION AS C_REGION,  
c.C_PHONE AS C_PHONE,  
c.C_MKTSEGMENT AS C_MKTSEGMENT,  
s.S_NAME AS S_NAME,  
s.S_ADDRESS AS S_ADDRESS,  
s.S_CITY AS S_CITY,  
s.S_NATION AS S_NATION,  
s.S_REGION AS S_REGION,  
s.S_PHONE AS S_PHONE,  
p.P_NAME AS P_NAME,  
p.P_MFGR AS P_MFGR,  
p.P_CATEGORY AS P_CATEGORY,  
p.P_BRAND AS P_BRAND,  
p.P_COLOR AS P_COLOR,  
p.P_TYPE AS P_TYPE,  
p.P_SIZE AS P_SIZE,  
p.P_CONTAINER AS P_CONTAINER  
FROM lineorder AS l
```

شمای جداول را از استار به فلت
کانورت می کنیم.



```
Applications Places Terminal Dec 21 11:47 younesious@kali: ~  
  
s.S_PHONE AS S_PHONE,  
p.P_NAME AS P_NAME,  
p.P_MFGR AS P_MFGR,  
p.P_CATEGORY AS P_CATEGORY,  
p.P_BRAND AS P_BRAND,  
p.P_COLOR AS P_COLOR,  
p.P_TYPE AS P_TYPE,  
p.P_SIZE AS P_SIZE,  
p.P_CONTAINER AS P_CONTAINER  
FROM lineorder AS l  
INNER JOIN customer AS c ON c.C_CUSTKEY = l.LO_CUSTKEY  
INNER JOIN supplier AS s ON s.S_SUPPKEY = l.LO_SUPPKEY  
INNER JOIN part AS p ON p.P_PARTKEY = l.LO_PARTKEY  
  
Query id: 41b14dd5-09a9-4839-878b-452feb5e16ef  
  
Ok.  
  
0 rows in set. Elapsed: 52.263 sec. Processed 6.23 million rows, 267.75 MB (119.27 tho  
usand rows/s., 5.12 MB/s.)
```

شمای جداول را از استار به فلت
کانورت می کنیم.




```
Applications Places Terminal
Dec 18 19:47
userg20@kali: /ssb-dbgen

S_CITY ASC,
P_BRAND ASC;

SELECT
  toYear(LO_ORDERDATE) AS year,
  S_CITY,
  P_BRAND,
  sum(LO_REVENUE - LO_SUPPLYCOST) AS profit
FROM lineorder_flat
WHERE (S_NATION = 'UNITED STATES') AND ((year = 1997) OR (year = 1998)) AND (P_CATEGORY = 'MFGR#14')
GROUP BY
  year,
  S_CITY,
  P_BRAND
ORDER BY
  year ASC,
  S_CITY ASC,
  P_BRAND ASC

Query id: c7b5e7ee-2865-400a-b8c8-208b98373bc1

year S_CITY P_BRAND profit
1997 UNITED ST0 MFGR#141 46368864
1997 UNITED ST0 MFGR#1410 98883339
1997 UNITED ST0 MFGR#1411 108499200
1997 UNITED ST0 MFGR#1412 54712353
1997 UNITED ST0 MFGR#1413 122258439
1997 UNITED ST0 MFGR#1414 72712764
1997 UNITED ST0 MFGR#1415 19257003
```

کویری ۴.۳: این کوئری سال.شهر و
مجموع سود ها را به شرط این که اهل
ایالات متحده باشند و سال ۱۹۹۸ یا
۱۹۹۷ باشد و در دسته بندی
MFGR#14 قرار بگیرند بر اساس
سال و شهر و برند گروه بندی کرده و به
ترتیب صعودی سال سپس در صورت
تساوی سال به ترتیب صعودی شهر و
سپس صعودی برند نشان می دهد.



ApplicationsPlacesTerminal

Dec 18 19:47

userg20@kallious: /ssb-dbgen

1998	UNITED	ST9	MFGR#1425	27782609
1998	UNITED	ST9	MFGR#1426	31270237
1998	UNITED	ST9	MFGR#1427	21856358
1998	UNITED	ST9	MFGR#1428	21441828
1998	UNITED	ST9	MFGR#1429	33590114
1998	UNITED	ST9	MFGR#143	42961444
1998	UNITED	ST9	MFGR#1430	45751199
1998	UNITED	ST9	MFGR#1431	10413405
1998	UNITED	ST9	MFGR#1432	14621151
1998	UNITED	ST9	MFGR#1433	26267990
1998	UNITED	ST9	MFGR#1434	6211183
1998	UNITED	ST9	MFGR#1435	21131164
1998	UNITED	ST9	MFGR#1436	43504
1998	UNITED	ST9	MFGR#1437	20708918
1998	UNITED	ST9	MFGR#1438	21876159
1998	UNITED	ST9	MFGR#1439	11945846
1998	UNITED	ST9	MFGR#144	18980348
1998	UNITED	ST9	MFGR#1440	6681029
1998	UNITED	ST9	MFGR#145	1841436
1998	UNITED	ST9	MFGR#146	20871862
1998	UNITED	ST9	MFGR#147	4759326
1998	UNITED	ST9	MFGR#148	14852692
1998	UNITED	ST9	MFGR#149	14143870

694 rows in set. Elapsed: 0.049 sec. Processed 3.59 million rows, 55.62 MB (72.84 million rows/s., 1.13 GB/s.)

kallious :)

کویری ۴.۳: این کوئری سال.شهر و
مجموع سود ها را به شرط این که اهل
ایالات متحده باشند و سال ۱۹۹۸ یا
۱۹۹۷ باشد و در دسته بندی
MFGR#14 قرار بگیرند بر اساس
سال و شهر و برند گروهبندی کرده و به
ترتیب صعودی سال سپس در صورت
تساوی سال به ترتیب صعودی شهر و
سپس صعودی برند نشان می دهد.




```
Applications Places Terminal Dec 18 19:46 userg20@kallious: /ssb-dbgem
ORDER BY
    year,
    P_BRAND;

SELECT
    sum(LO_REVENUE),
    toYear(LO_ORDERDATE) AS year,
    P_BRAND
FROM lineorder_flat
WHERE (P_CATEGORY = 'MFGR#12') AND (S_REGION = 'AMERICA')
GROUP BY
    year,
    P_BRAND
ORDER BY
    year ASC,
    P_BRAND ASC

Query id: 5bb3fb39-cf62-48b7-92a3-bab50c19c43e

sum(LO_REVENUE) year P_BRAND
1703514621 1992 MFGR#121
1831991370 1992 MFGR#1210
1652308986 1992 MFGR#1211
1947617568 1992 MFGR#1212
1872093723 1992 MFGR#1213
2011465404 1992 MFGR#1214
1899457410 1992 MFGR#1215
2024540343 1992 MFGR#1216
2025280305 1992 MFGR#1217
```

کویری ۲.۱:

این کوئری مجموع

LO_REVENUE و تاریخ سفارش

ها و برند را سلکت می کند از

lineorder و بر اساس سال و برند

گروه بندی می کند و ابتدا بر اساس

صعودی سال و در صورت تساوی سال

بر اساس صعودی برند مرتب می کند.



ApplicationsPlacesTerminal

Dec 18 19:46

userg20@kali: /ssb-dbgen

fa

🔍⋮⌵⌵

کویری ۲.۱:

این کوئری مجموع

LO_REVENUE و تاریخ سفارش

ها و برند را سلکت می کند از

lineorder و بر اساس سال و برند

گروه بندی می کند و ابتدا بر اساس

صعودی سال و در صورت تساوی سال

بر اساس صعودی برند مرتب می کند.




```
Applications Places Terminal Dec 18 19:47
userg20@kali: /ssb-dbggen

649315833 1998 MFGR#1226
559782565 1998 MFGR#1227
409932980 1998 MFGR#1228
616397671 1998 MFGR#1229
510049047 1998 MFGR#123
680139988 1998 MFGR#1230
632630138 1998 MFGR#1231
708116825 1998 MFGR#1232
668123950 1998 MFGR#1233
661909997 1998 MFGR#1234
569860150 1998 MFGR#1235
578899053 1998 MFGR#1236
513405516 1998 MFGR#1237
394693916 1998 MFGR#1238
655344065 1998 MFGR#1239
528855736 1998 MFGR#124
671943877 1998 MFGR#1240
559553265 1998 MFGR#125
540484842 1998 MFGR#126
599606311 1998 MFGR#127
585492440 1998 MFGR#128
474134892 1998 MFGR#129

280 rows in set. Elapsed: 0.087 sec. Processed 14.75 million rows, 152.65 MB (170.31 million rows/s., 1.76 GB/s.
)

kali:~$
```

کویری ۲.۱:

این کوئری مجموع

LO_REVENUE و تاریخ سفارش

ها و برند را سلکت می کند از

lineorder و بر اساس سال و برند

گروه بندی می کند و ابتدا بر اساس

صعودی سال و در صورت تساوی سال

بر اساس صعودی برند مرتب می کند.

