

Hive Assignment:

1. Download vehicle sales data -> https://github.com/shashank-mishra219/Hive-Class/blob/main/sales_order_data.csv

2. Store raw data into hdfs location

```
hadoop fs -mkdir /user/cloudera/hive_assignment
```

```
hadoop fs -put /home/cloudera/ajjoo/sales_order_data.csv  
/user/cloudera/hive_assignment
```

```
hadoop fs -ls /user/cloudera/hive_assignment
```

```
quickstart ~]$ hadoop fs -ls /user/cloudera/hive_assignment  
quickstart ~]$ /home/cloudera/ajjoo/  
e/cloudera/ajjoo/: is a directory  
quickstart ~]$ cd /home/cloudera/ajjoo/  
quickstart ajjoo]$ ls  
csv country wise latest_data.csv csv_file.csv depart_data.csv hive-hcatalog-core-0.14.0.jar json_file.csv sales_data_raw.csv sales_order_data.csv  
quickstart ajjoo]$ hadoop fs -put /home/cloudera/ajjoo/sales_order_data.csv /user/cloudera/hive_assignment  
quickstart ajjoo]$ hadoop fs -ls /user/cloudera/hive_assignment  
ms  
1 cloudera cloudera 360233 2022-12-23 10:24 /user/cloudera/hive_assignment/sales_order_data.csv
```

3. Create a internal hive table "sales_order_csv" which will store csv data
sales_order_csv make sure to skip header row while creating table.

```
show databases;
```

```
create database hive_assignment;
```

```
use hive_assignment;
```

```
show tables;
```

```
set hive.cli.print.header = true;
```

```
cloudera@quickstart:~/ajjoo  
hive> show databases;  
OK  
default  
hive_class_b1  
Time taken: 0.012 seconds, Fetched: 2 row(s)  
hive> create database hive_assignment;  
OK  
Time taken: 0.192 seconds  
hive> use hive_assignment;  
OK  
Time taken: 0.11 seconds  
hive> show tables;  
OK  
Time taken: 0.047 seconds  
hive> set hive.cli.print.header = true;
```

```
create table sales_order_csv(  
ORDERNUMBER int,  
QUANTITYORDERED int,
```

```
PRICEEACH float,  
ORDERLINENUMBER int,  
SALES float,  
STATUS string,  
QTR_ID int,  
MONTH_ID int,  
YEAR_ID int,  
PRODUCTLINE string,  
MSRP int,  
PRODUCTCODE string,  
PHONE string,  
CITY string,  
STATE string,  
POSTALCODE int,  
COUNTRY string,  
TERRITORY string,  
CONTACTLASTNAME string,  
CONTACTFIRSTNAME string,  
DEALSIZE string  
)  
  
row format delimited  
  
fields terminated by ','  
  
tblproperties ("skip.header.line.count" = "1");
```

```

cloudera@quickstart:~/ajjoo
hive> create table sales_order_csv(
  > ORDERNUMBER int,
  > QUANTITYORDERED int,
  > PRICEEACH float,
  > ORDERLINENUMBER int,
  > SALES float,
  > STATUS string,
  > QTR_ID int,
  > MONTH_ID int,
  > YEAR_ID int,
  > PRODUCTLINE string,
  > MSRP int,
  > PRODUCTCODE string,
  > PHONE string,
  > CITY string,
  > STATE string,
  > POSTALCODE int,
  > COUNTRY string,
  > TERRITORY string,
  > CONTACTLASTNAME string,
  > CONTACTFIRSTNAME string,
  > DEALSIZE string
  > )
  > row format delimited
  > fields terminated by ','
  > tblproperties ("skip.header.line.count" = "1");
OK
Time taken: 0.843 seconds

```

4. Load data from hdfs path into "sales_order_csv"

load data inpath 'hdfs:///user/cloudera/hive_assignment/sales_order_data.csv' into table sales_order_csv;

5. Create an internal hive table which will store data in ORC format "sales_order_orc".

create table sales_order_orc

(

ORDERNUMBER int,

QUANTITYORDERED int,

PRICEEACH float,

ORDERLINENUMBER int,

SALES float,

STATUS string,

QTR_ID int,

MONTH_ID int,

YEAR_ID int,

PRODUCTLINE string,

MSRP int,

```
PRODUCTCODE string,  
  
PHONE string,  
  
CITY string,  
  
STATE string,  
  
POSTALCODE int,  
  
COUNTRY string,  
  
TERRITORY string,  
  
CONTACTLASTNAME string,  
  
CONTACTFIRSTNAME string,  
  
DEALSIZE string  
  
)  
  
stored as ORC;
```

```
cloudera@quickstart:~/ajjoo  
hive> create table sales_order_orc  
> (  
> ORDERNUMBER int,  
> QUANTITYORDERED int,  
> PRICEEACH float,  
> ORDERLINENUMBER int,  
> SALES float,  
> STATUS string,  
> QTR_ID int,  
> MONTH_ID int,  
> YEAR_ID int,  
> PRODUCTLINE string,  
> MSRP int,  
> PRODUCTCODE string,  
> PHONE string,  
> CITY string,  
> STATE string,  
> POSTALCODE int,  
> COUNTRY string,  
> TERRITORY string,  
> CONTACTLASTNAME string,  
> CONTACTFIRSTNAME string,  
> DEALSIZE string  
> )  
> stored as ORC;  
OK  
Time taken: 0.881 seconds  
hive> show tables;  
OK  
tab_name  
sales_order_csv  
sales_order_orc  
Time taken: 0.883 seconds, Fetched: 2 row(s)  
hive>
```

6. Load data from "sales_order_csv" into "sales_order_orc".

from sales_order_csv insert overwrite table sales_order_orc select *;

Perform below mentioned queries on "sales_order_orc" table :

a. Calculate total sales per year.

Select sum(sales) as Total_sales ,year_id from sales_order_orc group by year_id;

```
cloudera@quickstart:~$ hive> Select sum(sales) as Total_sales ,year_id from sales_order_orc group by year_id;
Query ID = cloudera_20221223071616_2fbb7714-a92f-47a0-811b-aededacb92df
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0027, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0027/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0027
Hadoop job information for Stage-1: number of mappers: 17 number of reducers: 1
2022-12-23 07:16:27,650 Stage-1 map = 0%, reduce = 0%
2022-12-23 07:16:35,021 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.24 sec
2022-12-23 07:16:43,313 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.94 sec
MapReduce Total cumulative CPU time: 3 seconds 940 msec
Ended Job = job_1671520429232_0027
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.94 sec HDFS Read: 37070 HDFS Write: 70 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 940 msec
OK
total_sales  year_id
3516979.547241211 2003
4724162.593383789 2004
1791486.7086791992 2005
Time taken: 27.444 seconds, Fetched: 3 row(s)
hive>
```

b. Find a product for which maximum orders were placed.

Select p.productline from (Select sum(quantityordered) as maxorder,productline from sales_order_orc group by productline order by maxorder desc limit 1)p;

```
cloudera@quickstart:~$ hive> Select p.productline from (Select sum(quantityordered) as maxorder,productline from sales_order_orc group by productline order by maxorder desc limit 1)p;
Query ID = cloudera_20221223071414_1b436e2d-608a-4de8-b98c-2278fa5ae89a
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0025, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0025/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0025
Hadoop job information for Stage-1: number of mappers: 17 number of reducers: 1
2022-12-23 07:14:28,022 Stage-1 map = 0%, reduce = 0%
2022-12-23 07:14:37,852 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.02 sec
2022-12-23 07:14:47,222 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.6 sec
MapReduce Total cumulative CPU time: 4 seconds 600 msec
Ended Job = job_1671520429232_0025
Launching Job 2 out of 2
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0026, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0026/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0026
Hadoop job information for Stage-2: number of mappers: 17 number of reducers: 1
2022-12-23 07:15:01,089 Stage-2 map = 0%, reduce = 0%
2022-12-23 07:15:06,266 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 1.0 sec
2022-12-23 07:15:13,498 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 2.61 sec
MapReduce Total cumulative CPU time: 2 seconds 610 msec
Ended Job = job_1671520429232_0026
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.6 sec HDFS Read: 28595 HDFS Write: 311 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 2.61 sec HDFS Read: 5247 HDFS Write: 13 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 210 msec
OK
p.productline
Classic Cars
Time taken: 57.676 seconds, Fetched: 1 row(s)
hive>
```

c. Calculate the total sales for each quarter

Select sum(sales)as total_sales_per_quarter,qtr_id from sales_order_orc group by qtr_id;

```
cloudera@quickstart:~$ hive> select sum(sales) as total_sales_per_quarter, qtr_id from sales_order_orc group by qtr_id;
Query ID = cloudera_20221223071111_0b573e12-dc7d-4b93-be18-a575e4191845
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0024, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0024/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0024
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-12-23 07:12:01,874 Stage-1 map = 0%, reduce = 0%
2022-12-23 07:12:10,204 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.4 sec
2022-12-23 07:12:23,702 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.33 sec
MapReduce Total cumulative CPU time: 3 seconds 330 msec
Ended Job = job_1671520429232_0024
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.33 sec HDFS Read: 37261 HDFS Write: 81 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 330 msec
CS
total_sales_per_quarter qtr_id
2350817.726501465 1
2048120.3029174805 2
1758910.808959961 3
1874780.010925291 4
Time taken: 34.239 seconds, Fetched: 4 row(s)
hive>
```

d. In which quarter sales was minimum?

**select sum(sales) as minimum_sales_quarter, qtr_id from sales_order_orc group by qtr_id
order by minimum_sales_quarter limit 1;**

```
cloudera@quickstart:~$ hive> select sum(sales) as minimum_sales_quarter, qtr_id from sales_order_orc group by qtr_id order by minimum_sales_quarter limit 1;
Query ID = cloudera_20221223070303_1936616d-8f8a-4275-a179-d7c5105dcf8e
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0022, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0022/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0022
Hadoop job information for Stage-1: number of mappers: 17; number of reducers: 1
2022-12-23 07:04:34,508 Stage-1 map = 0%, reduce = 0%
2022-12-23 07:04:45,932 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.33 sec
2022-12-23 07:05:00,202 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 1.33 sec
MapReduce Total cumulative CPU time: 2 seconds 880 msec
Ended Job = job_1671520429232_0022
Launching Job 2 out of 2
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0023, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0023/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0023
Hadoop job information for Stage-2: number of mappers: 17; number of reducers: 1
2022-12-23 07:05:20,607 Stage-2 map = 0%, reduce = 0%
2022-12-23 07:05:29,136 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 1.12 sec
2022-12-23 07:05:38,561 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 2.77 sec
MapReduce Total cumulative CPU time: 2 seconds 770 msec
Ended Job = job_1671520429232_0023
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 2.88 sec HDFS Read: 36349 HDFS Write: 200 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 2.77 sec HDFS Read: 5106 HDFS Write: 20 SUCCESS
Total MapReduce CPU Time Spent: 5 seconds 650 msec
CS
minimum_sales_quarter qtr_id
1758910.808959961 3
Time taken: 111.461 seconds, Fetched: 1 row(s)
hive>
```

e. In which country sales was maximum and in which country sales was minimum.

**SELECT s.country, s.sales FROM(SELECT country,Sales,RANK() OVER (ORDER BY sales)
rnk_min,RANK() OVER (ORDER BY Sales DESC) rnk_max FROM sales_order_orc) s
WHERE rnk_min = 1 OR rnk_max = 1 ORDER BY Sales;**

```

cloudera@quickstart:~$
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-12-23 06:58:31,490 Stage-1 map = 0%, reduce = 0%
2022-12-23 06:58:38,795 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.89 sec
2022-12-23 06:58:46,299 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 5.8 sec
MapReduce Total cumulative CPU time: 5 seconds 800 msec
Ended Job = job_1671520429232_0019
Launching Job 2 out of 3
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0020, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0020/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0020
Hadoop job information for Stage-2: number of mappers: 17; number of reducers: 1
2022-12-23 06:59:03,959 Stage-2 map = 0%, reduce = 0%
2022-12-23 06:59:12,209 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 2.21 sec
2022-12-23 06:59:21,902 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 5.07 sec
MapReduce Total cumulative CPU time: 5 seconds 70 msec
Ended Job = job_1671520429232_0020
Launching Job 3 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0021, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0021/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0021
Hadoop job information for Stage-3: number of mappers: 17; number of reducers: 1
2022-12-23 06:59:35,242 Stage-3 map = 0%, reduce = 0%
2022-12-23 06:59:41,453 Stage-3 map = 100%, reduce = 0%, Cumulative CPU 1.07 sec
2022-12-23 06:59:46,721 Stage-3 map = 100%, reduce = 100%, Cumulative CPU 2.67 sec
MapReduce Total cumulative CPU time: 2 seconds 670 msec
Ended Job = job_1671520429232_0021
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 5.8 sec HDFS Read: 38635 HDFS Write: 85350 SUCCESS
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 5.07 sec HDFS Read: 93872 HDFS Write: 149 SUCCESS
Stage-Stage-3: Map: 1 Reduce: 1 Cumulative CPU: 2.67 sec HDFS Read: 4928 HDFS Write: 26 SUCCESS
Total MapReduce CPU Time Spent: 13 seconds 540 msec
OK
country      s.sales
France 482.13
USA 14082.8
Time taken: 93.147 seconds, Fetched: 2 row(s)
hive>

```

f. Calculate quartelry sales for each city.

Select sum(sales) as quarterly_sales, qtr_id, city from sales_order_orc group by qtr_id, city;

```

cloudera@quickstart:~$
hive> select sum(sales) as quarterly_sales, qtr_id, city from sales_order_orc group by qtr_id, city;
Query ID = cloudera_20221223065656_12dc29a6-4d78-4e81-b620-9elce60ce909
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1671520429232_0018, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0018/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0018
Hadoop job information for Stage-1: number of mappers: 17; number of reducers: 1
2022-12-23 06:56:23,831 Stage-1 map = 0%, reduce = 0%
2022-12-23 06:56:31,297 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.52 sec
2022-12-23 06:56:49,462 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 3.39 sec
MapReduce Total cumulative CPU time: 3 seconds 390 msec
Ended Job = job_1671520429232_0018
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 3.39 sec HDFS Read: 39341 HDFS Write: 5283 SUCCESS
Total MapReduce CPU Time Spent: 3 seconds 390 msec
OK
quarterly_sales qtr_id city
56181.320068359375 1 Bergamo
31606.72021484375 1 Boras
31474.7802734375 1 Brickhaven
16110.479558398488 1 Brisbane
15800.089721679688 1 Bruxelles
37850.07958984375 1 Burbank
13529.570190429688 1 Burlingame
21782.699951171875 1 Cambridge
14628.16015625 1 Charleroi
26906.68017578125 1 Cowes
38784.470458984375 1 Dublin
51373.49072265625 1 Espoo
48698.82922363281 1 Frankfurt
50432.549860546875 1 Gensve
3987.199951171875 1 Glendale
8775.159912109375 1 Graz
26422.819458007812 1 Helsinki
58971.110107421875 1 København
28170.1298828125 1 Lille
6477.219970703125 1 London
23889.320068359375 1 Los Angeles
9748.999755859375 1 Lule
101339.13977050781 1 Lyon
357668.489929192 1 Madrid
55245.02014160156 1 Makati City

```

h. Find a month for each year in which maximum number of quantities were sold.

select month_id,year_id,QUANTITYORDERED from (select
month_id,year_id,QUANTITYORDERED ,dense_rank() over(partition by year_id order
by QUANTITYORDERED desc) as rnks from sales_order_csv)s where s.rnks =1;

```

cloudera@quickstart~
hive> select month_id,year_id,QUANTITYORDERED from (select month_id,year_id,QUANTITYORDERED ,dense_rank() over(partition by year_id order by QUANTITYORDERED desc) as rnks from sales_order_
cav)s where s.rnks=1;
Query ID = cloudera_20221223065151_b2c233c7-0b82-4dd2-bfda-4775005dbdda
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reducers=<number>
Starting Job = job_1671520429232_0017, Tracking URL = http://quickstart.cloudera:8088/proxy/application_1671520429232_0017/
Kill Command = /usr/lib/hadoop/bin/hadoop job -kill job_1671520429232_0017
Hadoop job information for Stage-1: number of mappers: 17 number of reducers: 1
2022-12-23 06:51:19,967 Stage-1 map = 0%, reduce = 0%
2022-12-23 06:51:29,129 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 1.89 sec
2022-12-23 06:51:44,063 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 4.52 sec
MapReduce Total cumulative CPU time: 4 seconds 520 msec
Ended Job = job_1671520429232_0017
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 4.52 sec HDFS Read: 372950 HDFS Write: 392 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 520 msec
OK
month_id      year_id quantityordered
11            2003      50
3             2003      50
11            2003      50
5             2003      50
9             2003      50
2             2003      50
11            2003      50
10            2003      50
2             2003      50
2             2003      50
5             2003      50
11            2003      50
11            2003      50
10            2003      50
6             2003      50
9             2003      50
5             2003      50
10            2003      50
4             2003      50
6             2003      50
9             2003      50
10            2003      50
11            2003      50
10            2003      50

```