Command for importing csv file into the table.

```
\copy land_registry_price_paid_uk
from '/home/deepak/workspace/Exercise/DBMS/pp-monthly-update-new-version.csv'
with (format csv,header);
```

Query 1:Find the Pincode with highest average sale price.

```
answer:with temprarytable(pincode,highest_avg) as (select postcode,avg(price) as price from
land_registry_price_paid_uk
group by postcode order by price desc)
select pincode from temprarytable limit 1;
```

result:

```
 pincode
---------
 W8 5JN
(1 row)
```

Query 2:Find the Highest sale price for each year.

Answer:

```
select extract(year from transfer_date) as year, max( price) as sale_price
from land_registry_price_paid_uk
group by year
order by sale_price;
```

Result:

```
 year | sale_price
------+------------
 1996 |     325000
 1997 |     375000
 1998 |     400000
 1999 |     408000
 2003 |     481500
 2000 |     495000
 1995 |     650000
 2001 |     675000
 2005 |     680000
 2006 |     735000
 2011 |     895001
 2008 |     900000
 2007 |     950000
 2012 |    1050000
```

```
 2004 |    1372500
 2010 |    1650000
 2002 |    1750000
 2009 |    2937987
 2013 |    5881800
 2015 |   11500000
 2014 |   15400000
 2017 |   20930000
 2016 |   23100000
 2018 |   43100000
 2019 |  194800000
(25 rows)
```

Question:3 Find the Date with most sales.

Answer:

```
select transfer_date as date, count(transfer_date)as most_sale  from
land_registry_price_paid_uk
group by date
order by most_sale  desc
limit 1;
```

Result:

```
 date    | most_sale
------------+-----------
 2019-04-26 |      6753
(1 row)
```

Question 4:Find number of sales per pincode per year.

Answer:

```
select postcode, extract(year from transfer_date) as year, count(postcode) from
land_registry_price_paid_uk
group by year,postcode
order by postcode;
```

Question:5 Total sale per month.

Answer:

```
select extract(month from transfer_date) as month, count(extract(month from transfer_date))
from land_registry_price_paid_uk
group by month
order by month;
```

Result:

```
 month | count
-------+-------
     1 |  2082
     2 |  3551
     3 | 12126
     4 | 38629
     5 | 25784
     6 |   631
     7 |   605
     8 |   700
     9 |   985
    10 |  1264
    11 |  2854
    12 |  3379
(12 rows)
```

Question :6 Find Total sale amount per month per pincode by year.

Answer:

```
select postcode, extract(month from transfer_date) as month, count(postcode) from
land_registry_price_paid_uk
group by month,postcode
order by postcode;
```

Question :7 Create a daily cumulative sales report for 2019. Each day should have previous days sale amount added.

Answer:

```
Select transfer_date, price, sum(price) over(order by transfer_date) as runningPrice from
land_registry_price_paid_uk where extract(year from transfer_date)='2019' ;
```

Result :

```
 transfer_date |   price   | runningprice
---------------+-----------+--------------
 2019-01-01    |    480000 |      2336000
 2019-01-01    |    410000 |      2336000
 2019-01-01    |   1386000 |      2336000
 2019-01-01    |     60000 |      2336000
 2019-01-02    |    306995 |      4358740
```

2019-01-02	118995	4358740
2019-01-02	255000	4358740
2019-01-02	250000	4358740
2019-01-02	68000	4358740
2019-01-02	290000	4358740
2019-01-02	108750	4358740
2019-01-02	140000	4358740
2019-01-02	67000	4358740
2019-01-02	168000	4358740
2019-01-02	250000	4358740
2019-01-03	247000	39545200
2019-01-03	280000	39545200
2019-01-03	280000	39545200
2019-01-03	275000	39545200
2019-01-03	308000	39545200
2019-01-03	170000	39545200
2019-01-03	739995	39545200
2019-01-03	454995	39545200
2019-01-03	710000	39545200
2019-01-03	186000	39545200
2019-01-03	460000	39545200
2019-01-03	350000	39545200
2019-01-03	159000	39545200
2019-01-03	151250	39545200
2019-01-03	715000	39545200
2019-01-03	475000	39545200
2019-01-03	347000	39545200
2019-01-03	220000	39545200
2019-01-03	325	39545200