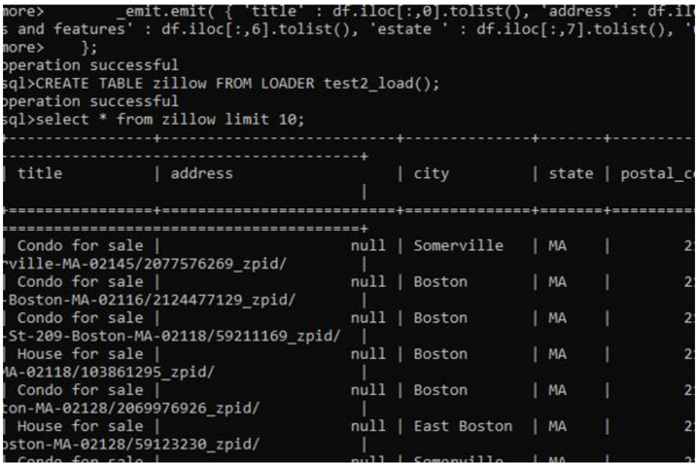
Large scale Data management

Name: Memtsas Athanasios AM: f3352115

Part 1 Monetdb:

The scripts are on finalMonetdb file. Firstly the load function:I tried to make the dataframe to dict, so I can use emit_emit but it didn't work. I made the dictionary brute force since there aren't much features.Then I renamed the 'facts and features' feature to 'fnfs' since sql cant read the name of the variable with spaces.The name I will be using for the table is 'zillow2' since 'zillow' was wrongly placed and registered.This happened for each function and concluded to change the name every time the function isn't working properly.



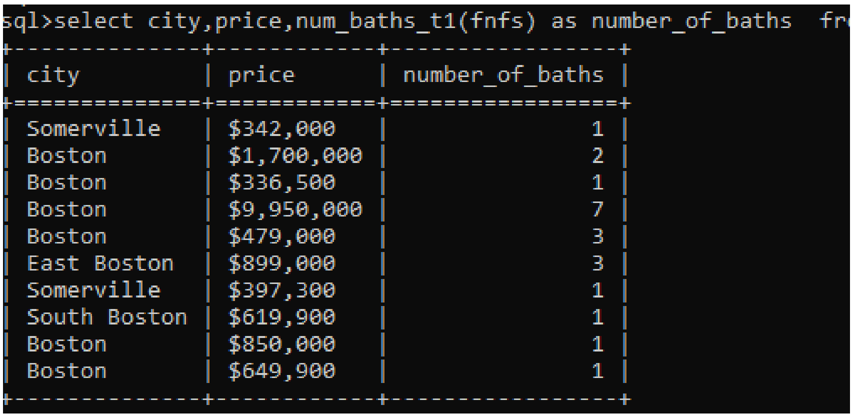
```
more>        _emit.emit( { 'title' : df.iloc[:,0].tolist(), 'address' : df.il
s and features' : df.iloc[:,6].tolist(), 'estate ' : df.iloc[:,7].tolist(), '
more>    };
operation successful
sql>CREATE TABLE zillow FROM LOADER test2_load();
operation successful
sql>select * from zillow limit 10;
+---------------+------------------------+-------------+-------+--------
--------------------------------------+
| title         | address                      | city        | state | postal_c
                                              |
+===============+========================+=============+=======+========
======================================+
| Condo for sale |                        null | Somerville  | MA    |        2
rville-MA-02145/2077576269_zpid/              |
| Condo for sale |                        null | Boston      | MA    |        2
-Boston-MA-02116/2124477129_zpid/             |
| Condo for sale |                        null | Boston      | MA    |        2
-St-209-Boston-MA-02118/59211169_zpid/        |
| House for sale |                        null | Boston      | MA    |        2
MA-02118/103861295_zpid/                      |
| Condo for sale |                        null | Boston      | MA    |        2
ton-MA-02128/2069976926_zpid/                 |
| House for sale |                        null | East Boston | MA    |        2
oston-MA-02128/59123230_zpid/                 |
```

Number of beds (Integers):The main idea is to iterate the structure with all the string's,transform it into tokens and select the first element .Although there are some 'None' values so zero will take its place (this happens for num of baths and sqft too).Finally append for each iteration the value on a list and return it.Finally in the run client we form the sql query and use Num_beds(fnfs) to select those values.

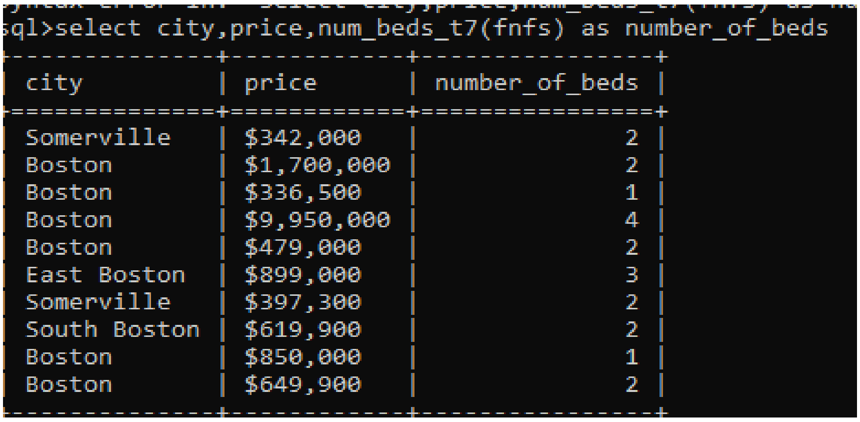
Every question will be structed as above.

Number of baths(Floats): The same but instead of token [0] the correct place is [2].

Baths and beds screenshots:

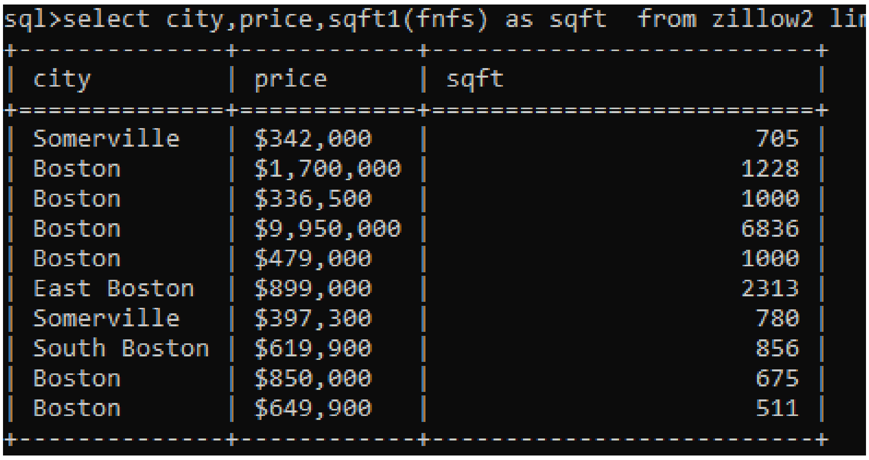


```
sql>select city,price,num_baths_t1(fnfs) as number_of_baths  fr
+-------------+-----------+----------------+
| city        | price     | number_of_baths |
+=============+===========+================+
| Somerville   | $342,000   |               1 |
| Boston       | $1,700,000 |               2 |
| Boston       | $336,500   |               1 |
| Boston       | $9,950,000 |               7 |
| Boston       | $479,000   |               3 |
| East Boston  | $899,000   |               3 |
| Somerville   | $397,300   |               1 |
| South Boston | $619,900   |               1 |
| Boston       | $850,000   |               1 |
| Boston       | $649,900   |               1 |
+-------------+-----------+----------------+
```



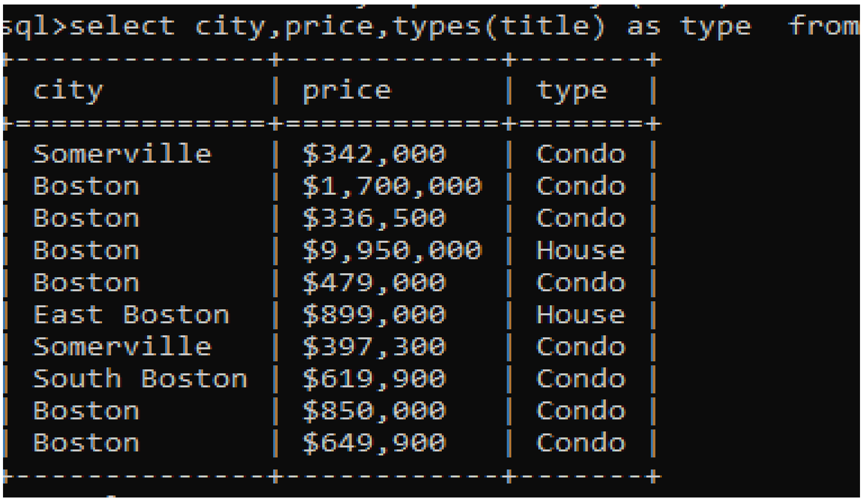
```
sql>select city,price,num_beds_t7(fnfs) as number_of_beds
+--------------+------------+----------------+
| city         | price      | number_of_beds |
+==============+============+================+
| Somerville   | $342,000   |              2 |
| Boston       | $1,700,000 |              2 |
| Boston       | $336,500   |              1 |
| Boston       | $9,950,000 |              4 |
| Boston       | $479,000   |              2 |
| East Boston  | $899,000   |              3 |
| Somerville   | $397,300   |              2 |
| South Boston | $619,900   |              2 |
| Boston       | $850,000   |              1 |
| Boston       | $649,900   |              2 |
+--------------+------------+----------------+
```

Sqft extraction(Floats): Same idea but in place [4] .Also the ‘,’ punctuation is not selected from the [4] token.



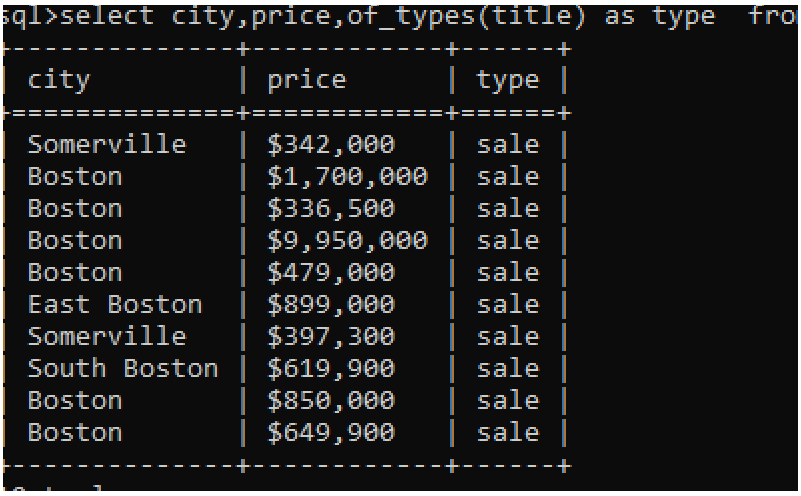
```
sql>select city,price,sqft1(fnfs) as sqft  from zillow2 li
+--------------+------------+-------------------------+
| city         | price      | sqft                    |
+==============+============+=========================+
| Somerville   | $342,000   |                     705 |
| Boston       | $1,700,000 |                    1228 |
| Boston       | $336,500   |                    1000 |
| Boston       | $9,950,000 |                    6836 |
| Boston       | $479,000   |                    1000 |
| East Boston  | $899,000   |                    2313 |
| Somerville   | $397,300   |                     780 |
| South Boston | $619,900   |                     856 |
| Boston       | $850,000   |                     675 |
| Boston       | $649,900   |                     511 |
+--------------+------------+-------------------------+
```

Extract type(String): Tokenizing title and append first token in a list to be returned.



```
sql>select city,price,types(title) as type  from
+--------------+------------+-------+
| city         | price      | type  |
+==============+============+=======+
| Somerville   | $342,000   | Condo |
| Boston       | $1,700,000 | Condo |
| Boston       | $336,500   | Condo |
| Boston       | $9,950,000 | House |
| Boston       | $479,000   | Condo |
| East Boston  | $899,000   | House |
| Somerville   | $397,300   | Condo |
| South Boston | $619,900   | Condo |
| Boston       | $850,000   | Condo |
| Boston       | $649,900   | Condo |
+--------------+------------+-------+
```

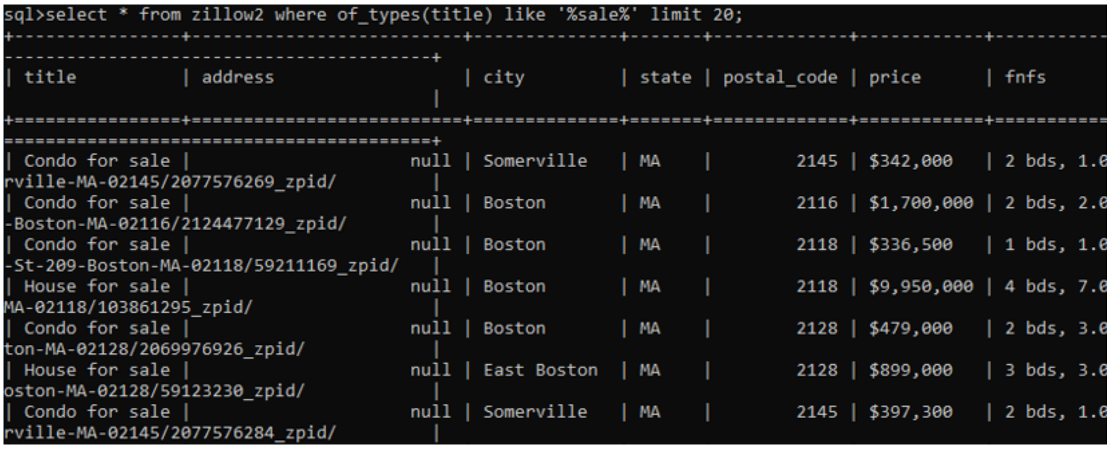
Extract offer(String): Tokenize replace punctuations with ‘’ and iterate each token to check if the value is one of the ones asked.If it is return it, if its not return sale instead of (new construct).



```
sql>select city,price,of_types(title) as type  fro
+--------------+------------+------+
| city         | price      | type |
+==============+============+======+
| Somerville   | $342,000   | sale |
| Boston       | $1,700,000 | sale |
| Boston       | $336,500   | sale |
| Boston       | $9,950,000 | sale |
| Boston       | $479,000   | sale |
| East Boston  | $899,000   | sale |
| Somerville   | $397,300   | sale |
| South Boston | $619,900   | sale |
| Boston       | $850,000   | sale |
| Boston       | $649,900   | sale |
+--------------+------------+------+
```

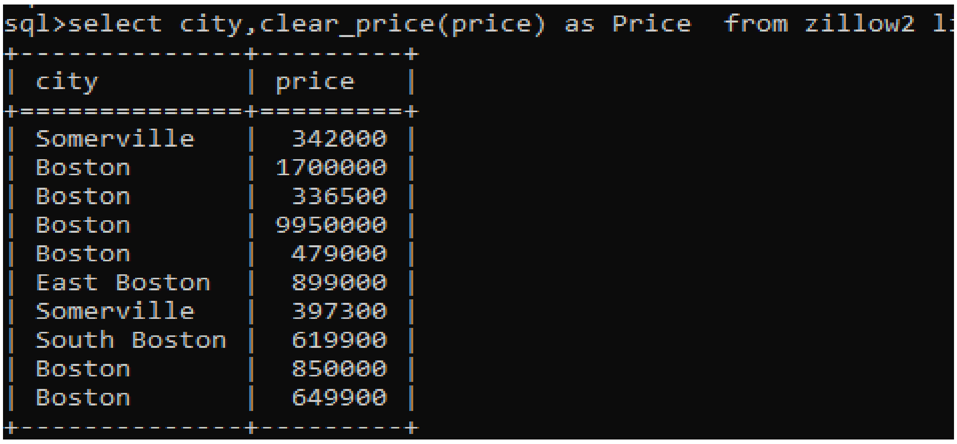
Keep only offers for sale: This can be done without a new function but by using the offer function and check if it contains sale anywhere (‘%sale%’)

Now we could create a view and apply all the filter questions but I didn’t understand if I should so I executed each question independently.



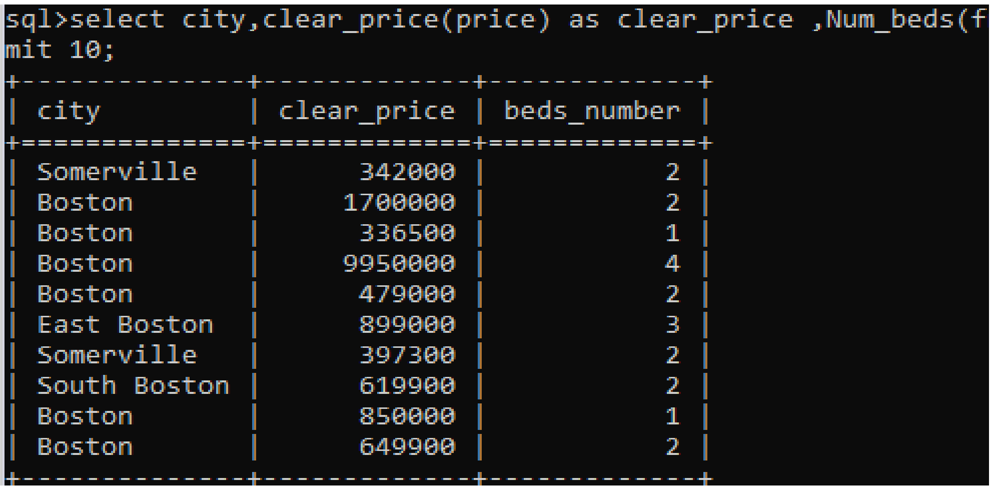
```
sql>select * from zillow2 where of_types(title) like '%sale%' limit 20;
+---------------+-----------------------+-------------+-------+------------+-----------+---------
-----------------------------------------+
| title         | address                    | city        | state | postal_code | price      | fnfs
                                           |
+===============+=======================+=============+=======+============+===========+=========
=========================================+
| Condo for sale |                        null | Somerville  | MA    |        2145 | $342,000   | 2 bds, 1.0
rville-MA-02145/2077576269_zpid/           |
| Condo for sale |                        null | Boston      | MA    |        2116 | $1,700,000 | 2 bds, 2.0
-Boston-MA-02116/2124477129_zpid/          |
| Condo for sale |                        null | Boston      | MA    |        2118 | $336,500   | 1 bds, 1.0
-St-209-Boston-MA-02118/59211169_zpid/     |
| House for sale |                        null | Boston      | MA    |        2118 | $9,950,000 | 4 bds, 7.0
MA-02118/103861295_zpid/                   |
| Condo for sale |                        null | Boston      | MA    |        2128 | $479,000   | 2 bds, 3.0
ton-MA-02128/2069976926_zpid/              |
| House for sale |                        null | East Boston | MA    |        2128 | $899,000   | 3 bds, 3.0
oston-MA-02128/59123230_zpid/              |
| Condo for sale |                        null | Somerville  | MA    |        2145 | $397,300   | 2 bds, 1.0
rville-MA-02145/2077576284_zpid/           |
```

Extract price(Integer): Replace ‘$’ , ‘,’ ,’+’ with ‘’ and transform each sting to int ,then return the list.



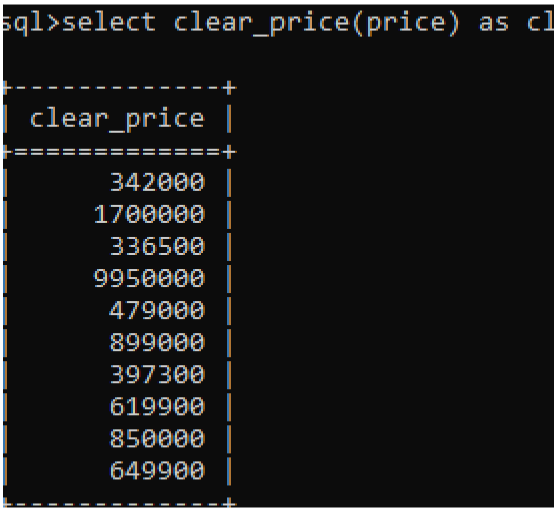
```
sql>select city,clear_price(price) as Price  from zillow2 li
+--------------+---------+
| city         | price   |
+==============+=========+
| Somerville   |  342000 |
| Boston       | 1700000 |
| Boston       |  336500 |
| Boston       | 9950000 |
| Boston       |  479000 |
| East Boston  |  899000 |
| Somerville   |  397300 |
| South Boston |  619900 |
| Boston       |  850000 |
| Boston       |  649900 |
+--------------+---------+
```

Filter more than 10 beds: Single query (with a condition where <)



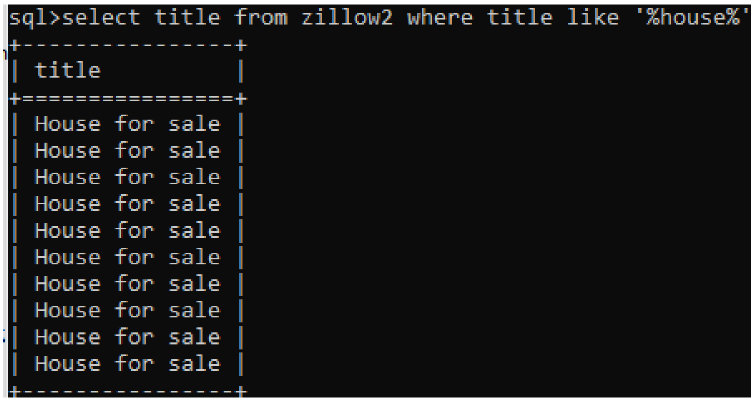
```
sql>select city,clear_price(price) as clear_price ,Num_beds(f
mit 10;
+--------------+-------------+-------------+
| city         | clear_price | beds_number |
+==============+=============+=============+
| Somerville   |      342000 |           2 |
| Boston       |     1700000 |           2 |
| Boston       |      336500 |           1 |
| Boston       |     9950000 |           4 |
| Boston       |      479000 |           2 |
| East Boston  |      899000 |           3 |
| Somerville   |      397300 |           2 |
| South Boston |      619900 |           2 |
| Boston       |      850000 |           1 |
| Boston       |      649900 |           2 |
+--------------+-------------+-------------+
```

Filter price : Singe query and condition on price function created:



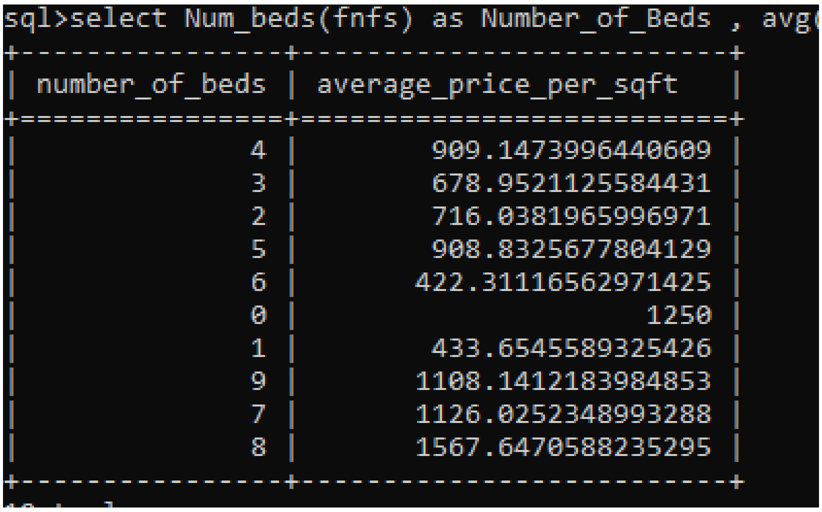
```
sql>select clear_price(price) as cl

+-------------+
| clear_price |
+=============+
|      342000 |
|     1700000 |
|      336500 |
|     9950000 |
|      479000 |
|      899000 |
|      397300 |
|      619900 |
|      850000 |
|      649900 |
+-------------+
```

Keep only Houses records: Single query with condition like '%House%' or %house% because it has some Multi-house values.



```
sql>select title from zillow2 where title like '%house%'
+---------------+
| title         |
+===============+
| House for sale |
| House for sale |
| House for sale |
| House for sale |
| House for sale |
| House for sale |
| House for sale |
| House for sale |
| House for sale |
| House for sale |
+---------------+
```

Average: On the 2 fucntions for price and sqft and substracting them while using group by for number of beds function with condition (like %House for sale%)



```
sql>select Num_beds(fnfs) as Number_of_Beds , avg(
+----------------+--------------------------+
| number_of_beds | average_price_per_sqft   |
+================+==========================+
|              4 |         909.1473996440609 |
|              3 |         678.9521125584431 |
|              2 |         716.0381965996971 |
|              5 |         908.8325677804129 |
|              6 |        422.31116562971425 |
|              0 |                      1250 |
|              1 |         433.6545589325426 |
|              9 |        1108.1412183984853 |
|              7 |        1126.0252348993288 |
|              8 |        1567.6470588235295 |
+----------------+--------------------------+
```

Part 2 Pyspark:

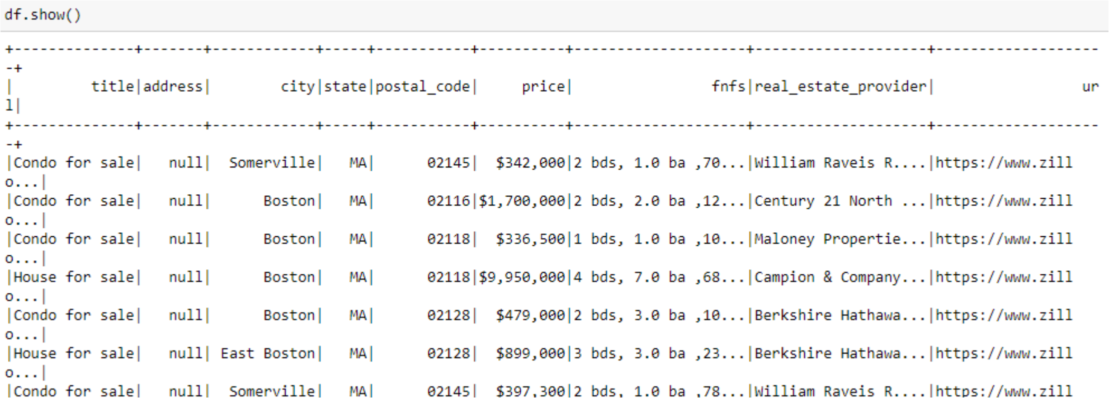
Following the instructions we load the database. Then to make facts and features also understandable for sql queries we import col from pyspark.functions and rename it again to fnfs.

Now for every question the structure is the same : If I want to build a function and then perform an sql query the steps are:

1. **Initialize function**
2. **Use udf with lambda to apply the function in each row (because the function takes a single value as input)**
3. **Register the udf function so it can be reused**
4. **Create temporary view**
5. **Finally apply the function on an sql query via spark.sql()**
6. **Use show() to see the results**

So we don’t need to think again about the functions we must create, just copy the commands from monetdb functions but without the iteration because we have a single value to process.

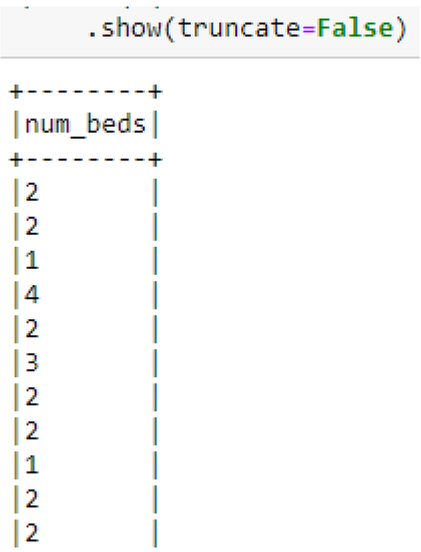
Load:



```
df.show()
```

```
+--------------+-------+------------+-----+-----------+----------+--------------------+--------------------+-------------------
-+
|         title|address|        city|state|postal_code|     price|                fnfs|real_estate_provider|                 ur
l|
+--------------+-------+------------+-----+-----------+----------+--------------------+--------------------+-------------------
-+
|Condo for sale|   null|  Somerville|   MA|      02145|  $342,000|2 bds, 1.0 ba ,70...|William Raveis R....|https://www.zill
o...|
|Condo for sale|   null|      Boston|   MA|      02116|$1,700,000|2 bds, 2.0 ba ,12...|Century 21 North ...|https://www.zill
o...|
|Condo for sale|   null|      Boston|   MA|      02118|  $336,500|1 bds, 1.0 ba ,10...|Maloney Propertie...|https://www.zill
o...|
|House for sale|   null|      Boston|   MA|      02118|$9,950,000|4 bds, 7.0 ba ,68...|Campion & Company...|https://www.zill
o...|
|Condo for sale|   null|      Boston|   MA|      02128|  $479,000|2 bds, 3.0 ba ,10...|Berkshire Hathawa...|https://www.zill
o...|
|House for sale|   null| East Boston|   MA|      02128|  $899,000|3 bds, 3.0 ba ,23...|Berkshire Hathawa...|https://www.zill
o...|
|Condo for sale|   null|  Somerville|   MA|      02145|  $397,300|2 bds, 1.0 ba ,78...|William Raveis R....|https://www.zill
```

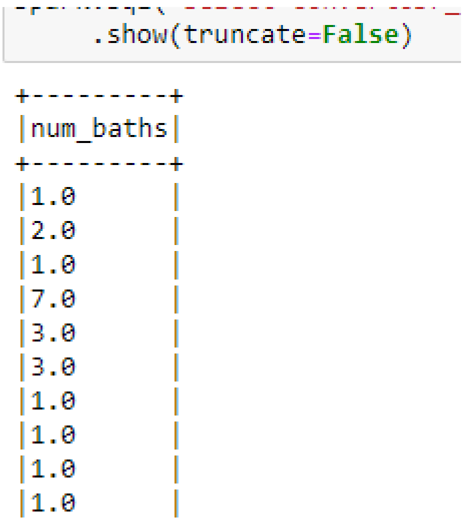
Number of beds (Integers):



```
    .show(truncate=False)
```

```
+--------+
|num_beds|
+--------+
|2       |
|2       |
|1       |
|4       |
|2       |
|3       |
|2       |
|2       |
|1       |
|2       |
|2       |
```

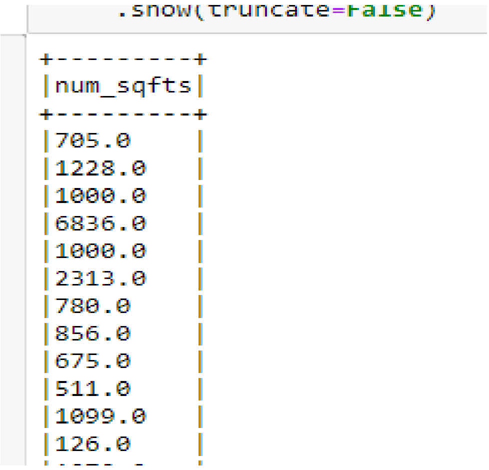
Number of baths(Floats):



```
    .show(truncate=False)
```

```
+---------+
|num_baths|
+---------+
|1.0      |
|2.0      |
|1.0      |
|7.0      |
|3.0      |
|3.0      |
|1.0      |
|1.0      |
|1.0      |
|1.0      |
```

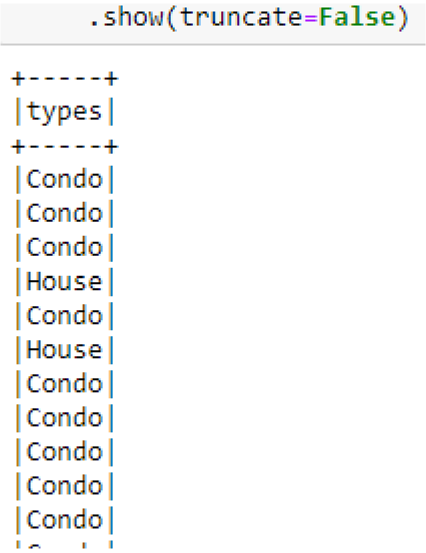
Sqft extraction(Floats):



```
    .show(truncate=False)
```

```
+---------+
|num_sqfts|
+---------+
|705.0    |
|1228.0   |
|1000.0   |
|6836.0   |
|1000.0   |
|2313.0   |
|780.0    |
|856.0    |
|675.0    |
|511.0    |
|1099.0   |
|126.0    |
```

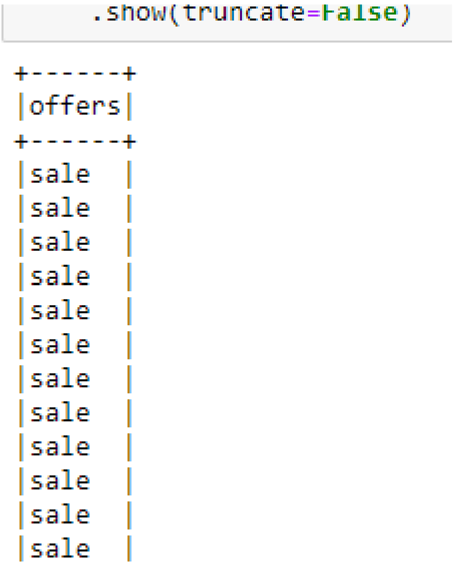
Extract type(String):



```
     .show(truncate=False)
```

```
+-----+
|types|
+-----+
|Condo|
|Condo|
|Condo|
|House|
|Condo|
|House|
|Condo|
|Condo|
|Condo|
|Condo|
|Condo|
```

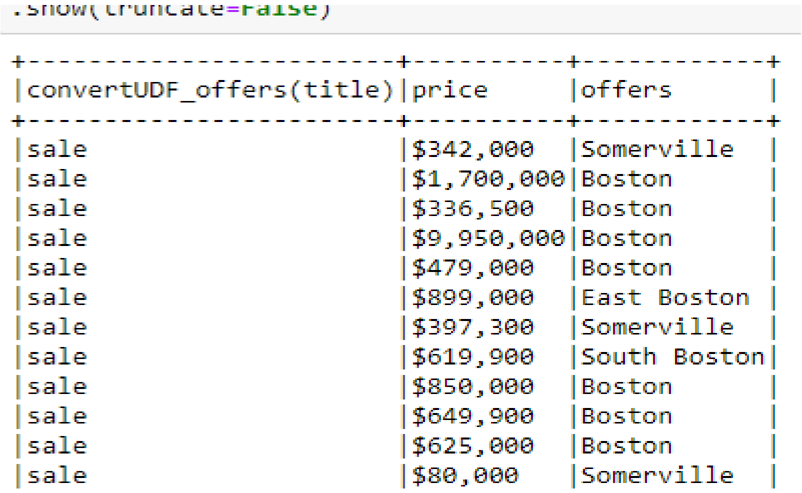
Extract offer(String):



```
     .show(truncate=False)
```

```
+------+
|offers|
+------+
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
|sale  |
```

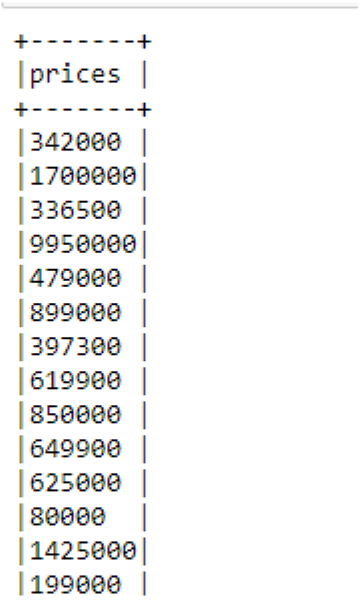
Keep only offers for sale:



```
.show(truncate=False)
```

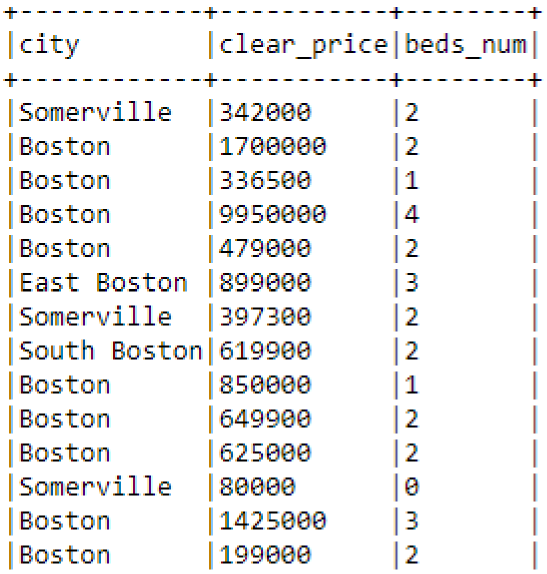
```
+------------------------+----------+------------+
|convertUDF_offers(title)|price     |offers      |
+------------------------+----------+------------+
|sale                    |$342,000  |Somerville  |
|sale                    |$1,700,000|Boston      |
|sale                    |$336,500  |Boston      |
|sale                    |$9,950,000|Boston      |
|sale                    |$479,000  |Boston      |
|sale                    |$899,000  |East Boston |
|sale                    |$397,300  |Somerville  |
|sale                    |$619,900  |South Boston|
|sale                    |$850,000  |Boston      |
|sale                    |$649,900  |Boston      |
|sale                    |$625,000  |Boston      |
|sale                    |$80,000   |Somerville  |
```

Extract price(Integer)



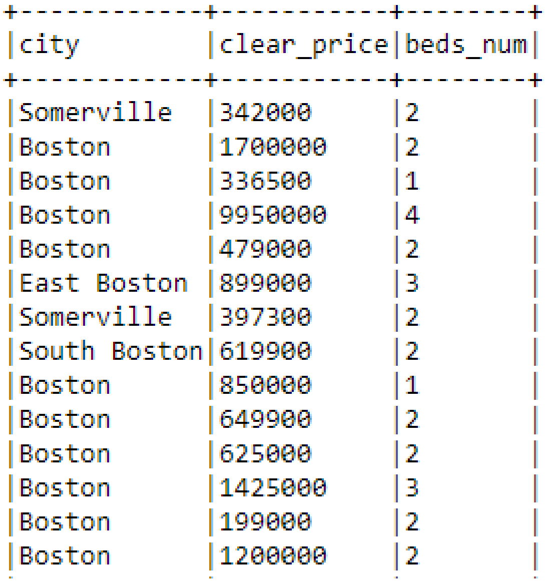
```
+-------+
|prices |
+-------+
|342000 |
|1700000|
|336500 |
|9950000|
|479000 |
|899000 |
|397300 |
|619900 |
|850000 |
|649900 |
|625000 |
|80000  |
|1425000|
|199000 |
```

Filter more than 10 beds:



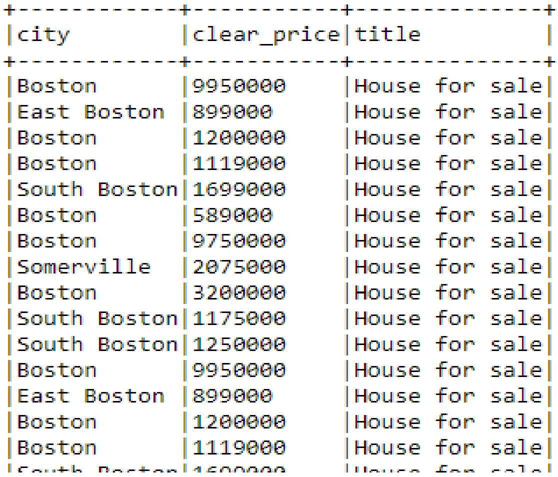
```
+------------+-----------+--------+
|city        |clear_price|beds_num|
+------------+-----------+--------+
|Somerville  |342000     |2       |
|Boston      |1700000    |2       |
|Boston      |336500     |1       |
|Boston      |9950000    |4       |
|Boston      |479000     |2       |
|East Boston |899000     |3       |
|Somerville  |397300     |2       |
|South Boston|619900     |2       |
|Boston      |850000     |1       |
|Boston      |649900     |2       |
|Boston      |625000     |2       |
|Somerville  |80000      |0       |
|Boston      |1425000    |3       |
|Boston      |199000     |2       |
```

Filter price :



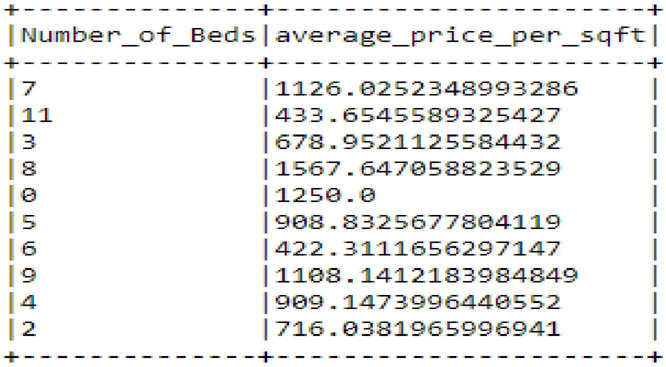
```
+------------+-----------+--------+
|city        |clear_price|beds_num|
+------------+-----------+--------+
|Somerville  |342000     |2       |
|Boston      |1700000    |2       |
|Boston      |336500     |1       |
|Boston      |9950000    |4       |
|Boston      |479000     |2       |
|East Boston |899000     |3       |
|Somerville  |397300     |2       |
|South Boston|619900     |2       |
|Boston      |850000     |1       |
|Boston      |649900     |2       |
|Boston      |625000     |2       |
|Boston      |1425000    |3       |
|Boston      |199000     |2       |
|Boston      |1200000    |2       |
```

Keep only Houses records:



```
+------------+-----------+--------------+
|city        |clear_price|title         |
+------------+-----------+--------------+
|Boston      |9950000    |House for sale|
|East Boston |899000     |House for sale|
|Boston      |1200000    |House for sale|
|Boston      |1119000    |House for sale|
|South Boston|1699000    |House for sale|
|Boston      |589000     |House for sale|
|Boston      |9750000    |House for sale|
|Somerville  |2075000    |House for sale|
|Boston      |3200000    |House for sale|
|South Boston|1175000    |House for sale|
|South Boston|1250000    |House for sale|
|Boston      |9950000    |House for sale|
|East Boston |899000     |House for sale|
|Boston      |1200000    |House for sale|
|Boston      |1119000    |House for sale|
|South Boston|1699000    |House for sale|
```

Average:



```
+--------------+----------------------+
|Number_of_Beds|average_price_per_sqft|
+--------------+----------------------+
|7             |1126.0252348993286    |
|11            |433.6545589325427     |
|3             |678.9521125584432     |
|8             |1567.647058823529     |
|0             |1250.0                |
|5             |908.8325677804119     |
|6             |422.3111656297147     |
|9             |1108.1412183984849    |
|4             |909.1473996440552     |
|2             |716.0381965996941     |
+--------------+----------------------+
```