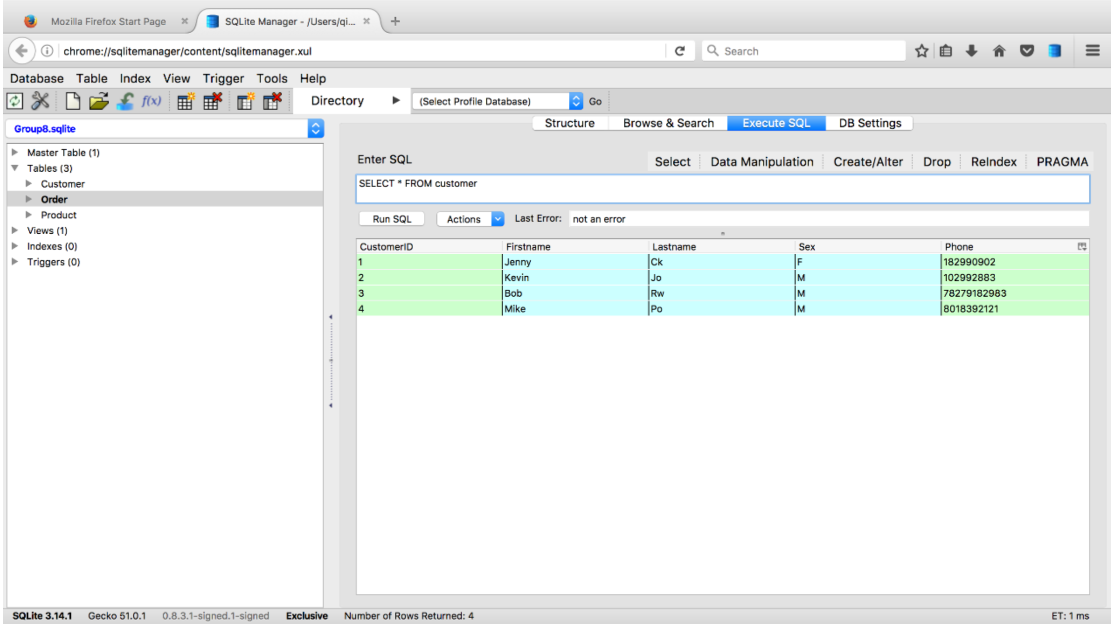
SQLite

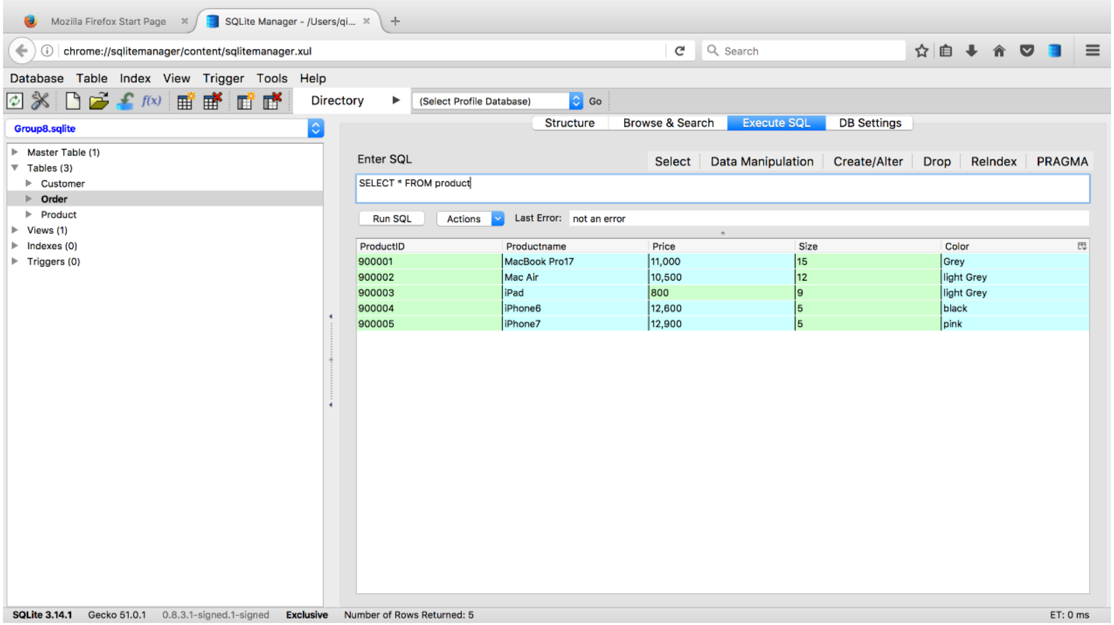


```
SELECT * FROM customer
```

Last Error: not an error

CustomerID	Firstname	Lastname	Sex	Phone
1	Jenny	Ck	F	182990902
2	Kevin	Jo	M	102992883
3	Bob	Rw	M	78279182983
4	Mike	Po	M	8018392121

SQLite 3.14.1 Gecko 51.0.1 0.8.3.1-signed.1-signed Exclusive Number of Rows Returned: 4 ET: 1 ms

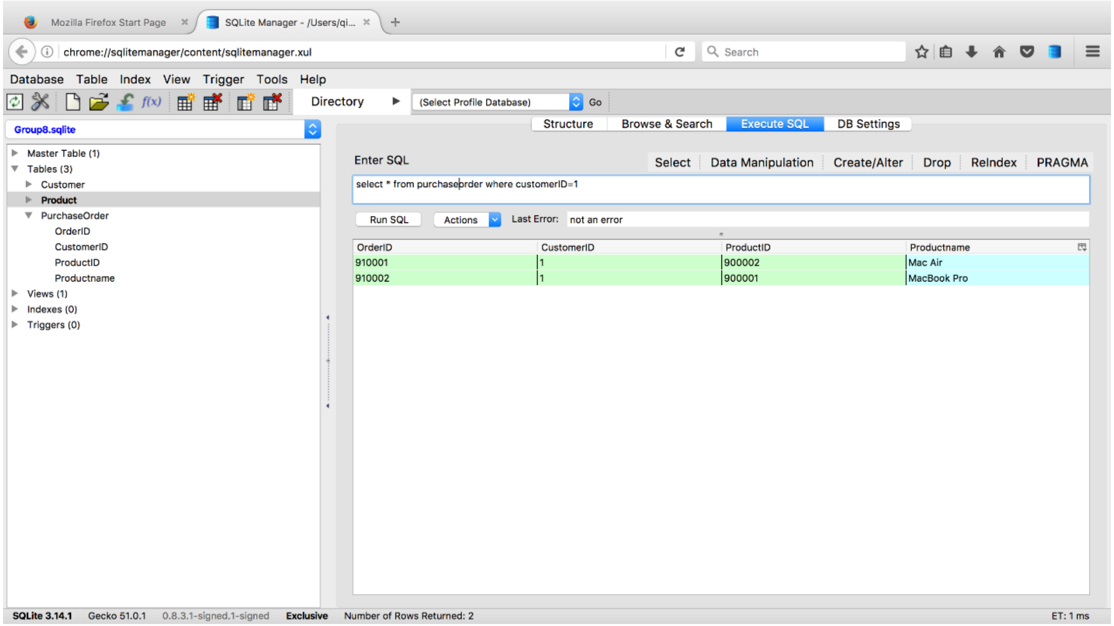


```
SELECT * FROM product
```

Last Error: not an error

ProductID	Productname	Price	Size	Color
900001	MacBook Pro17	11,000	15	Grey
900002	Mac Air	10,500	12	light Grey
900003	iPad	800	9	light Grey
900004	iPhone6	12,600	5	black
900005	iPhone7	12,900	5	pink

SQLite 3.14.1 Gecko 51.0.1 0.8.3.1-signed.1-signed Exclusive Number of Rows Returned: 5 ET: 0 ms



```
select * from purchaseorder where customerID=1
```

Last Error: not an error

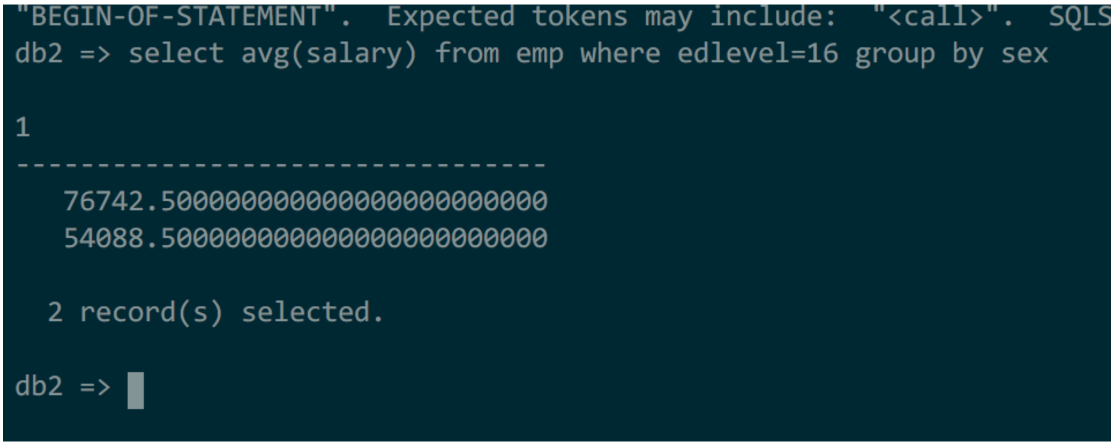
OrderID	CustomerID	ProductID	Productname
910001	1	900002	Mac Air
910002	1	900001	MacBook Pro

Number of Rows Returned: 2

In 3rd normal form, Productname should not be included in the PurchaseOrder table as it is already in the product table. However, we chose to include Productname in the PurchaseOrder table just for performance consideration so that when running many queries against the order table, we don’t have to join two tables repeatedly.

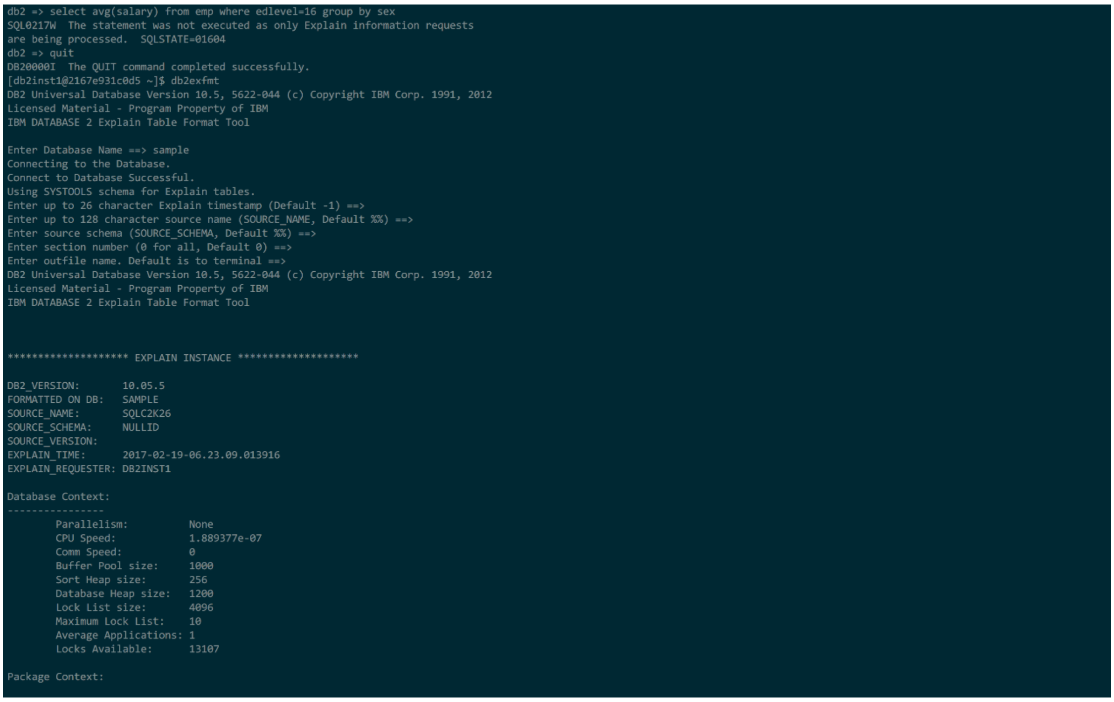
DB2 Express C

Run a sample query (use where clause and Group by):

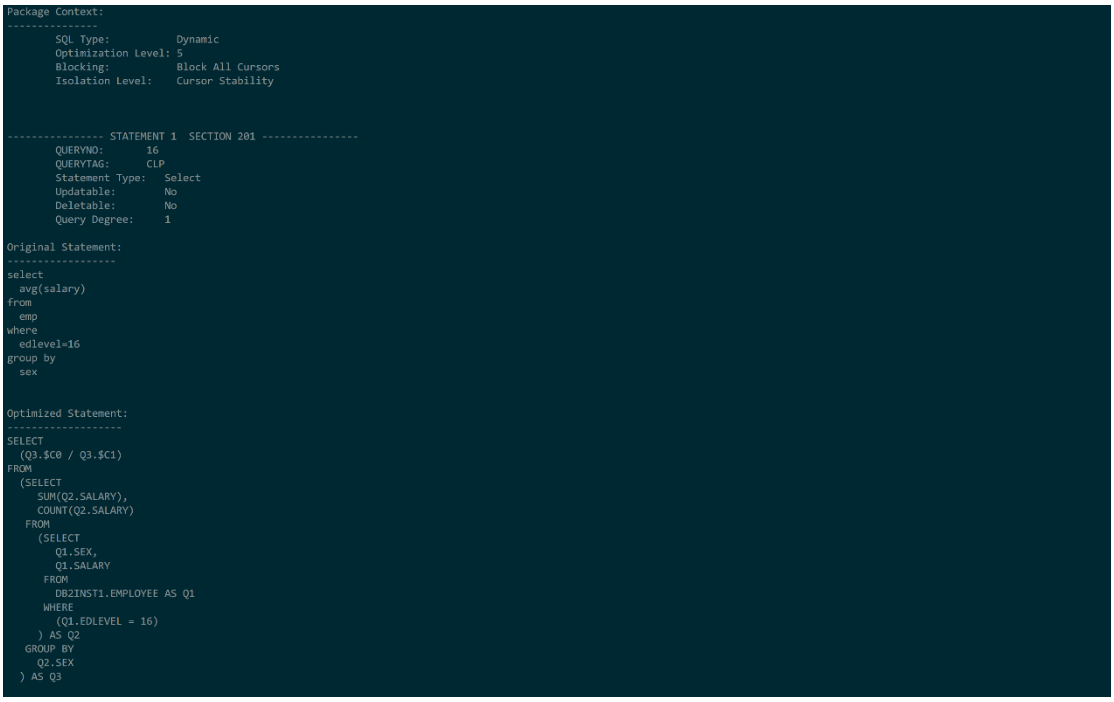


```
"BEGIN-OF-STATEMENT".  Expected tokens may include:  "<call>".  SQLS
db2 => select avg(salary) from emp where edlevel=16 group by sex

1
---------------------------------
   76742.50000000000000000000000
   54088.50000000000000000000000

  2 record(s) selected.

db2 => 
```

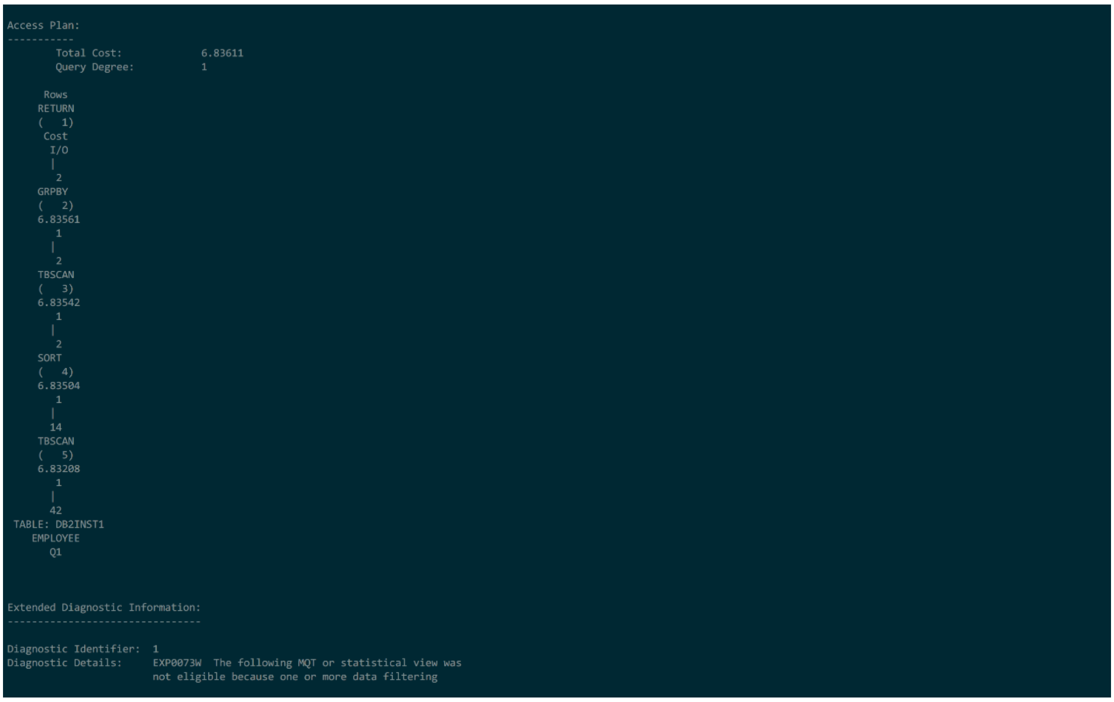
Generate query explain plan (use: db2exfmt tool):



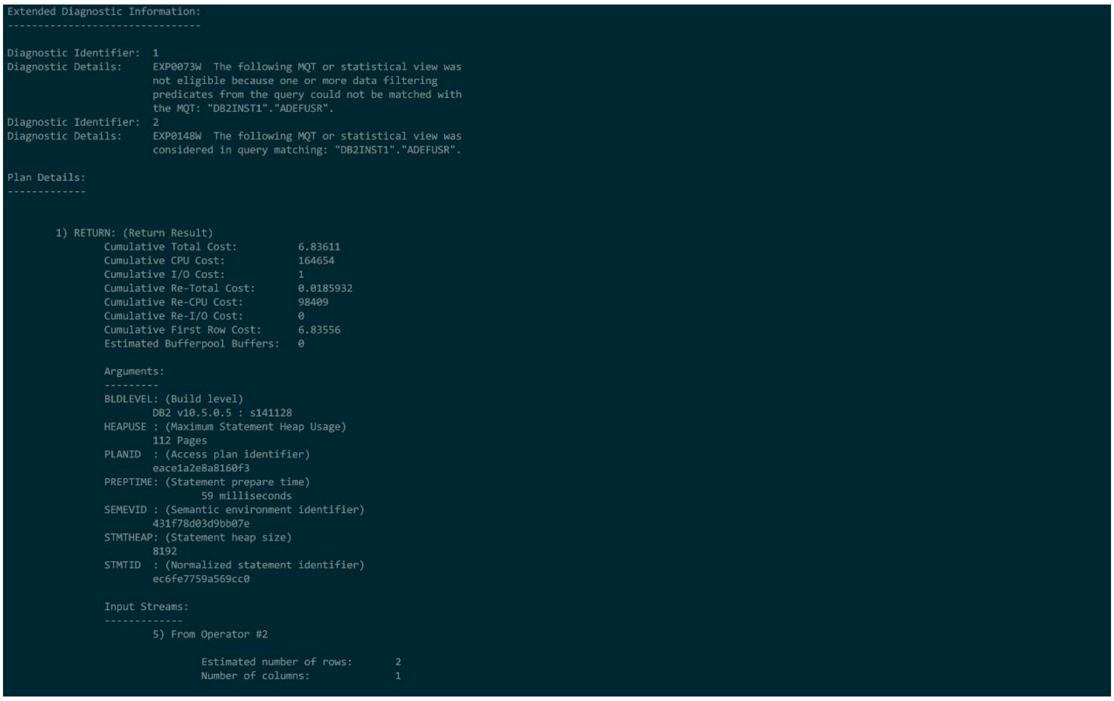
```
db2 => select avg(salary) from emp where edlevel=16 group by sex
SQL0217W  The statement was not executed as only Explain information requests
are being processed.  SQLSTATE=01604
db2 => quit
DB20000I  The QUIT command completed successfully.
[db2inst1@2167e931c0d5 ~]$ db2exfmt
DB2 Universal Database Version 10.5, 5622-044 (c) Copyright IBM Corp. 1991, 2012
Licensed Material - Program Property of IBM
IBM DATABASE 2 Explain Table Format Tool

Enter Database Name ==> sample
Connecting to the Database.
Connect to Database Successful.
Using SYSTOOLS schema for Explain tables.
Enter up to 26 character Explain timestamp (Default -1) ==>
Enter up to 128 character source name (SOURCE_NAME, Default %%) ==>
Enter source schema (SOURCE_SCHEMA, Default %%) ==>
Enter section number (0 for all, Default 0) ==>
Enter outfile name. Default is to terminal ==>
DB2 Universal Database Version 10.5, 5622-044 (c) Copyright IBM Corp. 1991, 2012
Licensed Material - Program Property of IBM
IBM DATABASE 2 Explain Table Format Tool



******************** EXPLAIN INSTANCE ********************

DB2_VERSION:       10.05.5
FORMATTED ON DB:   SAMPLE
SOURCE_NAME:       SQLC2K26
SOURCE_SCHEMA:     NULLID
SOURCE_VERSION:
EXPLAIN_TIME:      2017-02-19-06.23.09.013916
EXPLAIN_REQUESTER: DB2INST1

Database Context:
----------------
        Parallelism:          None
        CPU Speed:            1.889377e-07
        Comm Speed:           0
        Buffer Pool size:     1000
        Sort Heap size:       256
        Database Heap size:   1200
        Lock List size:       4096
        Maximum Lock List:    10
        Average Applications: 1
        Locks Available:      13107

Package Context:
```



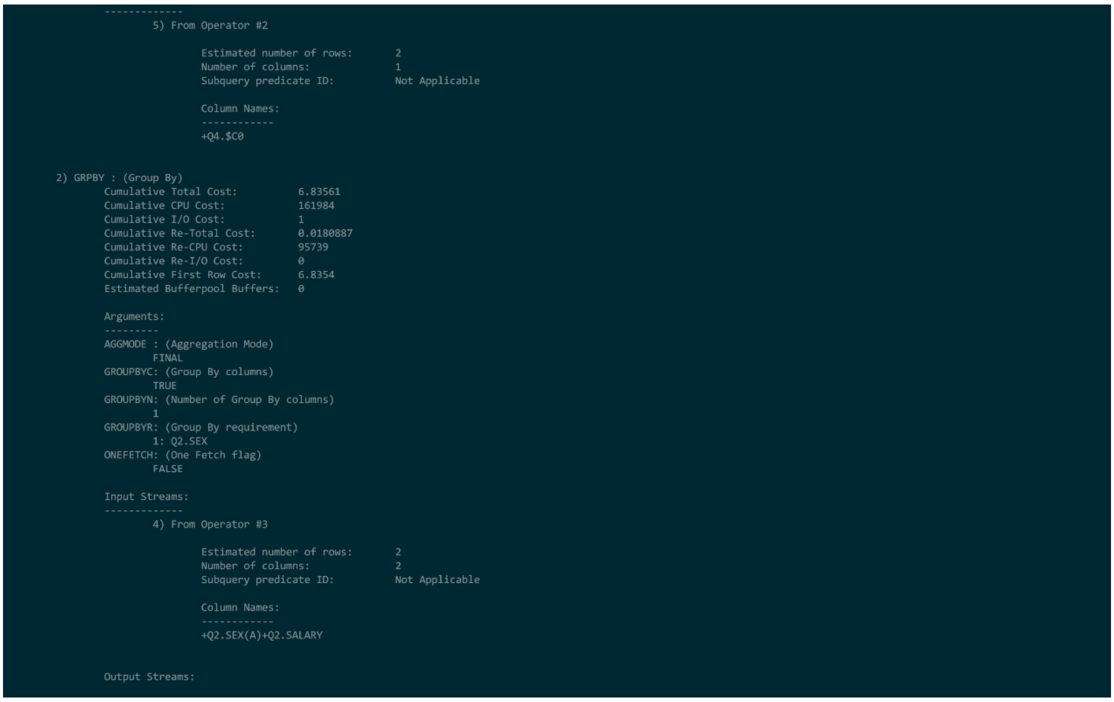
```
Package Context:
---------------
        SQL Type:           Dynamic
        Optimization Level: 5
        Blocking:           Block All Cursors
        Isolation Level:    Cursor Stability



---------------- STATEMENT 1  SECTION 201 ----------------
        QUERYNO:       16
        QUERYTAG:      CLP
        Statement Type:   Select
        Updatable:        No
        Deletable:        No
        Query Degree:     1

Original Statement:
------------------
select
  avg(salary)
from
  emp
where
  edlevel=16
group by
  sex


Optimized Statement:
-------------------
SELECT
  (Q3.$C0 / Q3.$C1)
FROM
  (SELECT
     SUM(Q2.SALARY),
     COUNT(Q2.SALARY)
   FROM
     (SELECT
        Q1.SEX,
        Q1.SALARY
      FROM
        DB2INST1.EMPLOYEE AS Q1
      WHERE
        (Q1.EDLEVEL = 16)
     ) AS Q2
   GROUP BY
     Q2.SEX
  ) AS Q3
```



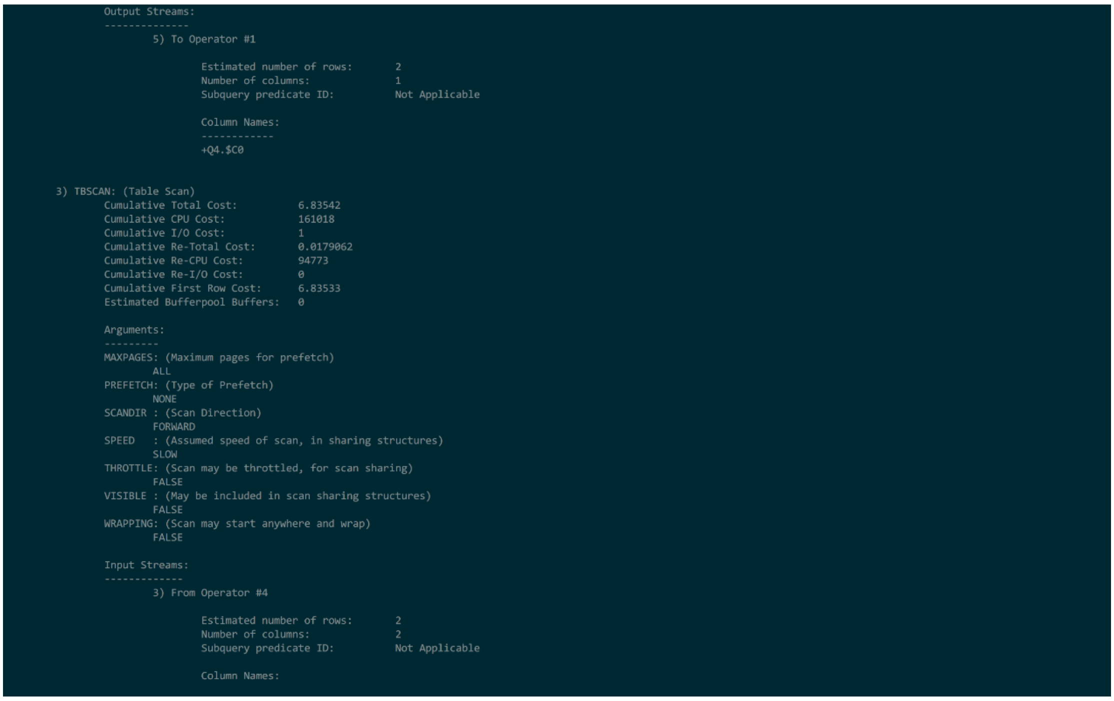
```
Access Plan:
-----------
        Total Cost:             6.83611
        Query Degree:           1

      Rows
     RETURN
     (   1)
      Cost
       I/O
       |
        2
     GRPBY
     (   2)
     6.83561
        1
       |
        2
     TBSCAN
     (   3)
     6.83542
        1
       |
        2
     SORT
     (   4)
     6.83504
        1
       |
       14
     TBSCAN
     (   5)
     6.83208
        1
       |
       42
 TABLE: DB2INST1
    EMPLOYEE
       Q1



Extended Diagnostic Information:
--------------------------------

Diagnostic Identifier:  1
Diagnostic Details:     EXP0073W  The following MQT or statistical view was
                        not eligible because one or more data filtering
```



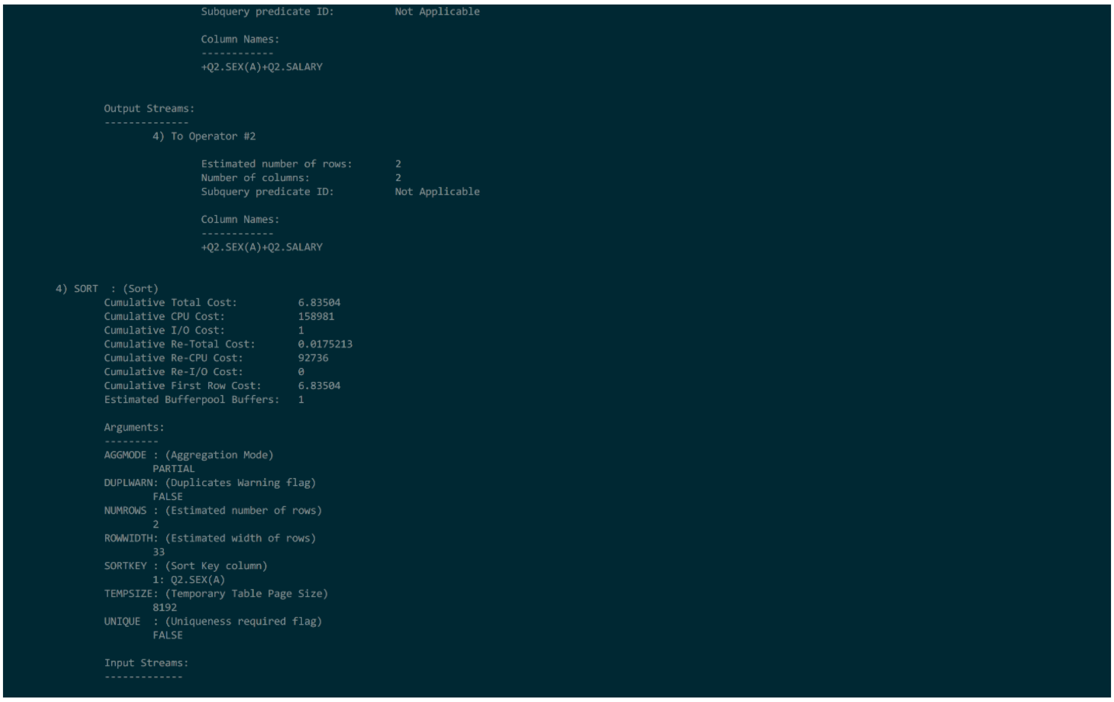
```
Extended Diagnostic Information:
--------------------------------

Diagnostic Identifier:  1
Diagnostic Details:     EXP0073W  The following MQT or statistical view was
                        not eligible because one or more data filtering
                        predicates from the query could not be matched with
                        the MQT: "DB2INST1"."ADEFUSR".
Diagnostic Identifier:  2
Diagnostic Details:     EXP0148W  The following MQT or statistical view was
                        considered in query matching: "DB2INST1"."ADEFUSR".

Plan Details:
-------------


        1) RETURN: (Return Result)
                Cumulative Total Cost:          6.83611
                Cumulative CPU Cost:            164654
                Cumulative I/O Cost:            1
                Cumulative Re-Total Cost:       0.0185932
                Cumulative Re-CPU Cost:         98409
                Cumulative Re-I/O Cost:         0
                Cumulative First Row Cost:      6.83556
                Estimated Bufferpool Buffers:   0

                Arguments:
                ---------
                BLDLEVEL: (Build level)
                        DB2 v10.5.0.5 : s141128
                HEAPUSE : (Maximum Statement Heap Usage)
                        112 Pages
                PLANID  : (Access plan identifier)
                        eace1a2e8a8160f3
                PREPTIME: (Statement prepare time)
                                59 milliseconds
                SEMEVID : (Semantic environment identifier)
                        431f78d03d9bb07e
                STMTHEAP: (Statement heap size)
                        8192
                STMTID  : (Normalized statement identifier)
                        ec6fe7759a569cc0

                Input Streams:
                -------------
                        5) From Operator #2

                                Estimated number of rows:       2
                                Number of columns:              1
```



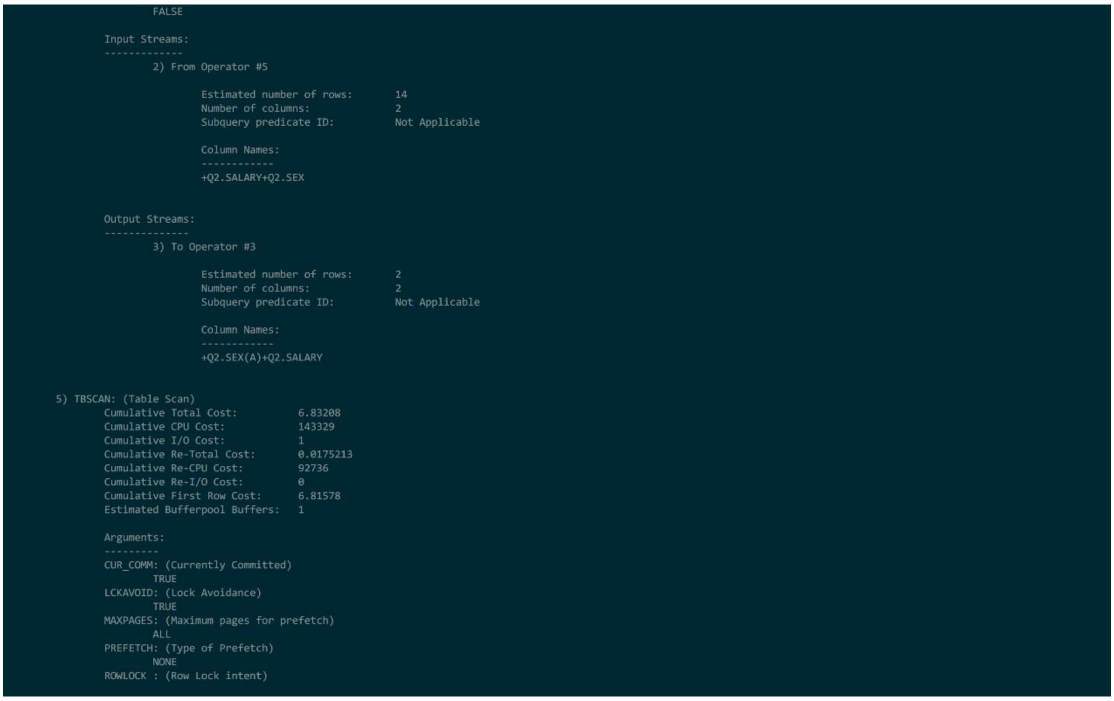
```
                -------------
                        5) From Operator #2

                                Estimated number of rows:       2
                                Number of columns:              1
                                Subquery predicate ID:          Not Applicable

                                Column Names:
                                ------------
                                +Q4.$C0


        2) GRPBY : (Group By)
                Cumulative Total Cost:          6.83561
                Cumulative CPU Cost:            161984
                Cumulative I/O Cost:            1
                Cumulative Re-Total Cost:       0.0180887
                Cumulative Re-CPU Cost:         95739
                Cumulative Re-I/O Cost:         0
                Cumulative First Row Cost:      6.8354
                Estimated Bufferpool Buffers:   0

                Arguments:
                ---------
                AGGMODE : (Aggregation Mode)
                        FINAL
                GROUPBYC: (Group By columns)
                        TRUE
                GROUPBYN: (Number of Group By columns)
                        1
                GROUPBYR: (Group By requirement)
                        1: Q2.SEX
                ONEFETCH: (One Fetch flag)
                        FALSE

                Input Streams:
                -------------
                        4) From Operator #3

                                Estimated number of rows:       2
                                Number of columns:              2
                                Subquery predicate ID:          Not Applicable

                                Column Names:
                                ------------
                                +Q2.SEX(A)+Q2.SALARY


                Output Streams:
```



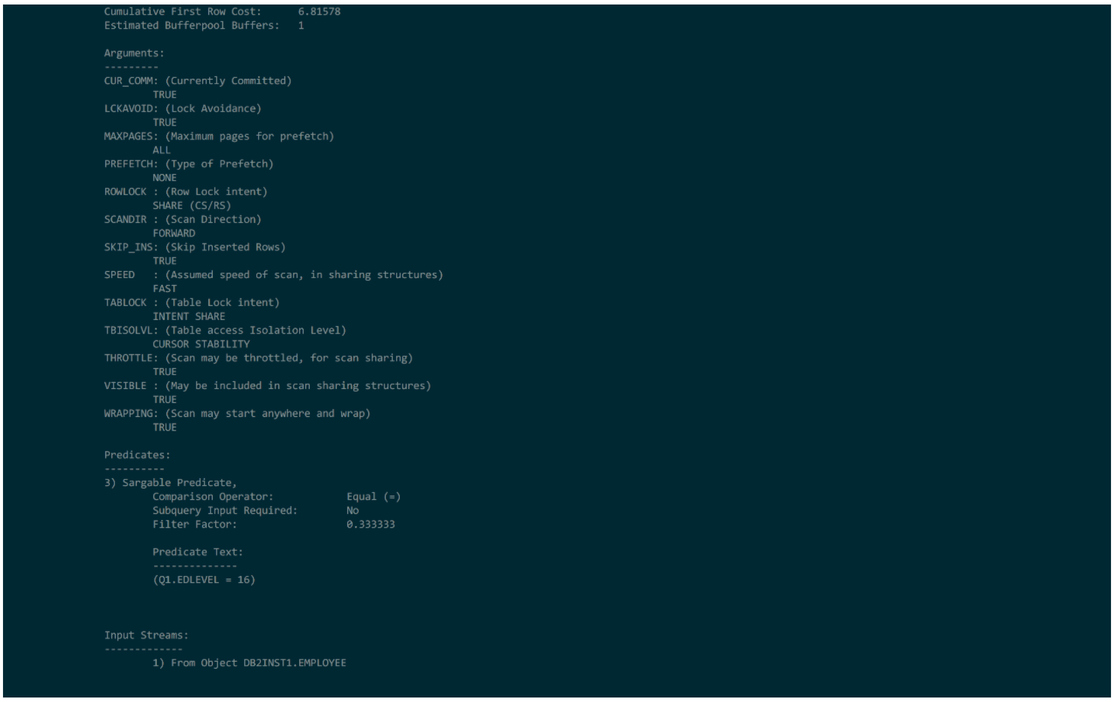
```
                Output Streams:
                --------------
                        5) To Operator #1

                                Estimated number of rows:       2
                                Number of columns:              1
                                Subquery predicate ID:          Not Applicable

                                Column Names:
                                ------------
                                +Q4.$C0


        3) TBSCAN: (Table Scan)
                Cumulative Total Cost:          6.83542
                Cumulative CPU Cost:            161018
                Cumulative I/O Cost:            1
                Cumulative Re-Total Cost:       0.0179062
                Cumulative Re-CPU Cost:         94773
                Cumulative Re-I/O Cost:         0
                Cumulative First Row Cost:      6.83533
                Estimated Bufferpool Buffers:   0

                Arguments:
                ---------
                MAXPAGES: (Maximum pages for prefetch)
                        ALL
                PREFETCH: (Type of Prefetch)
                        NONE
                SCANDIR : (Scan Direction)
                        FORWARD
                SPEED   : (Assumed speed of scan, in sharing structures)
                        SLOW
                THROTTLE: (Scan may be throttled, for scan sharing)
                        FALSE
                VISIBLE : (May be included in scan sharing structures)
                        FALSE
                WRAPPING: (Scan may start anywhere and wrap)
                        FALSE

                Input Streams:
                -------------
                        3) From Operator #4

                                Estimated number of rows:       2
                                Number of columns:              2
                                Subquery predicate ID:          Not Applicable

                                Column Names:
```



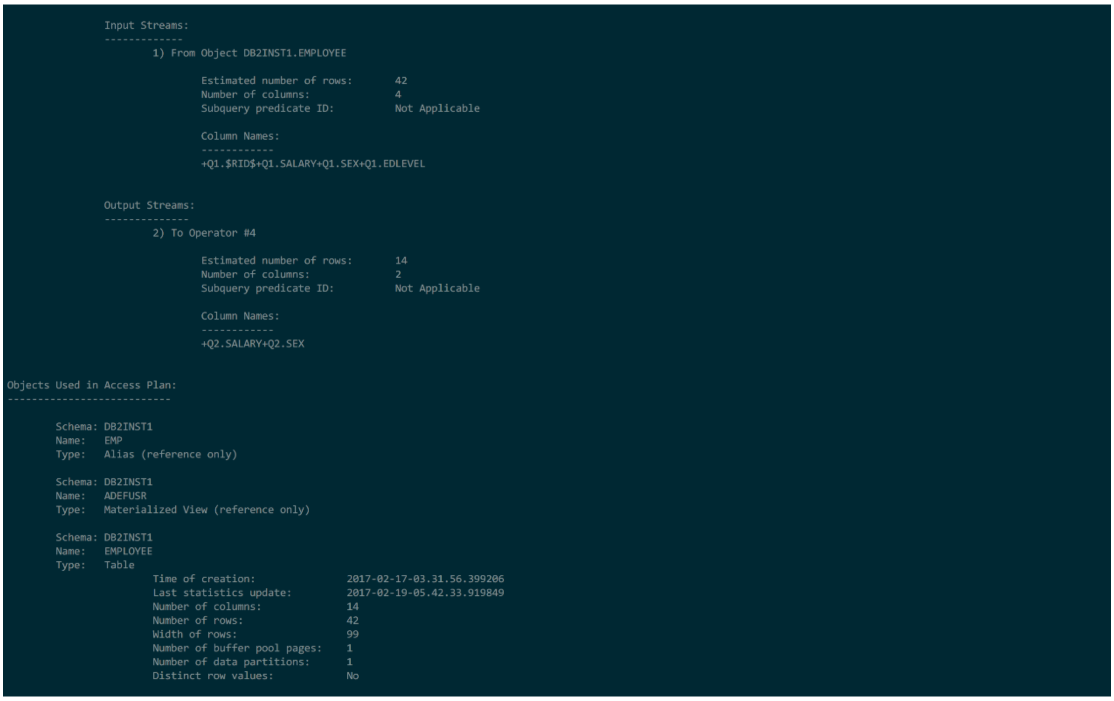
```
                        Subquery predicate ID:          Not Applicable

                        Column Names:
                        ------------
                        +Q2.SEX(A)+Q2.SALARY


        Output Streams:
        --------------
                4) To Operator #2

                        Estimated number of rows:       2
                        Number of columns:              2
                        Subquery predicate ID:          Not Applicable

                        Column Names:
                        ------------
                        +Q2.SEX(A)+Q2.SALARY


4) SORT  : (Sort)
        Cumulative Total Cost:          6.83504
        Cumulative CPU Cost:            158981
        Cumulative I/O Cost:            1
        Cumulative Re-Total Cost:       0.0175213
        Cumulative Re-CPU Cost:         92736
        Cumulative Re-I/O Cost:         0
        Cumulative First Row Cost:      6.83504
        Estimated Bufferpool Buffers:   1

        Arguments:
        ---------
        AGGMODE : (Aggregation Mode)
                PARTIAL
        DUPLWARN: (Duplicates Warning flag)
                FALSE
        NUMROWS : (Estimated number of rows)
                2
        ROWWIDTH: (Estimated width of rows)
                33
        SORTKEY : (Sort Key column)
                1: Q2.SEX(A)
        TEMPSIZE: (Temporary Table Page Size)
                8192
        UNIQUE  : (Uniqueness required flag)
                FALSE

        Input Streams:
        -------------
```



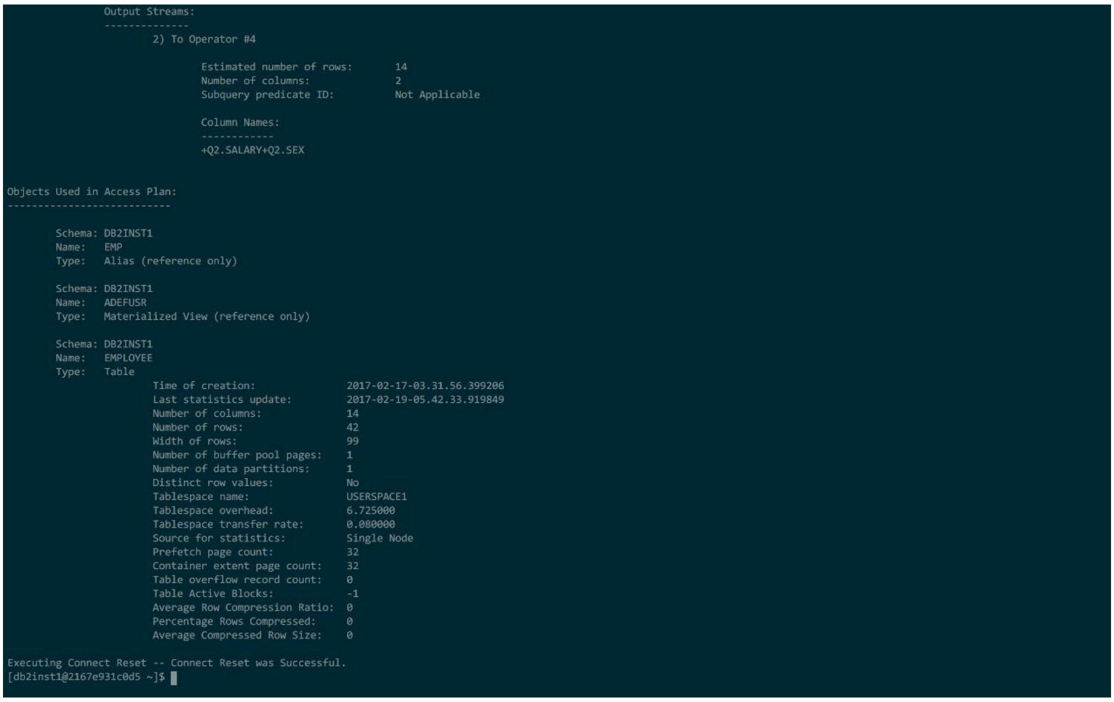
```
                FALSE

        Input Streams:
        -------------
                2) From Operator #5

                        Estimated number of rows:       14
                        Number of columns:              2
                        Subquery predicate ID:          Not Applicable

                        Column Names:
                        ------------
                        +Q2.SALARY+Q2.SEX


        Output Streams:
        --------------
                3) To Operator #3

                        Estimated number of rows:       2
                        Number of columns:              2
                        Subquery predicate ID:          Not Applicable

                        Column Names:
                        ------------
                        +Q2.SEX(A)+Q2.SALARY


5) TBSCAN: (Table Scan)
        Cumulative Total Cost:          6.83208
        Cumulative CPU Cost:            143329
        Cumulative I/O Cost:            1
        Cumulative Re-Total Cost:       0.0175213
        Cumulative Re-CPU Cost:         92736
        Cumulative Re-I/O Cost:         0
        Cumulative First Row Cost:      6.81578
        Estimated Bufferpool Buffers:   1

        Arguments:
        ---------
        CUR_COMM: (Currently Committed)
                TRUE
        LCKAVOID: (Lock Avoidance)
                TRUE
        MAXPAGES: (Maximum pages for prefetch)
                ALL
        PREFETCH: (Type of Prefetch)
                NONE
        ROWLOCK : (Row Lock intent)
```



```
                Cumulative First Row Cost:      6.81578
                Estimated Bufferpool Buffers:   1

                Arguments:
                ---------
                CUR_COMM: (Currently Committed)
                        TRUE
                LCKAVOID: (Lock Avoidance)
                        TRUE
                MAXPAGES: (Maximum pages for prefetch)
                        ALL
                PREFETCH: (Type of Prefetch)
                        NONE
                ROWLOCK : (Row Lock intent)
                        SHARE (CS/RS)
                SCANDIR : (Scan Direction)
                        FORWARD
                SKIP_INS: (Skip Inserted Rows)
                        TRUE
                SPEED   : (Assumed speed of scan, in sharing structures)
                        FAST
                TABLOCK : (Table Lock intent)
                        INTENT SHARE
                TBISOLVL: (Table access Isolation Level)
                        CURSOR STABILITY
                THROTTLE: (Scan may be throttled, for scan sharing)
                        TRUE
                VISIBLE : (May be included in scan sharing structures)
                        TRUE
                WRAPPING: (Scan may start anywhere and wrap)
                        TRUE

                Predicates:
                ----------
                3) Sargable Predicate,
                        Comparison Operator:            Equal (=)
                        Subquery Input Required:        No
                        Filter Factor:                  0.333333

                        Predicate Text:
                        --------------
                        (Q1.EDLEVEL = 16)



                Input Streams:
                -------------
                        1) From Object DB2INST1.EMPLOYEE
```

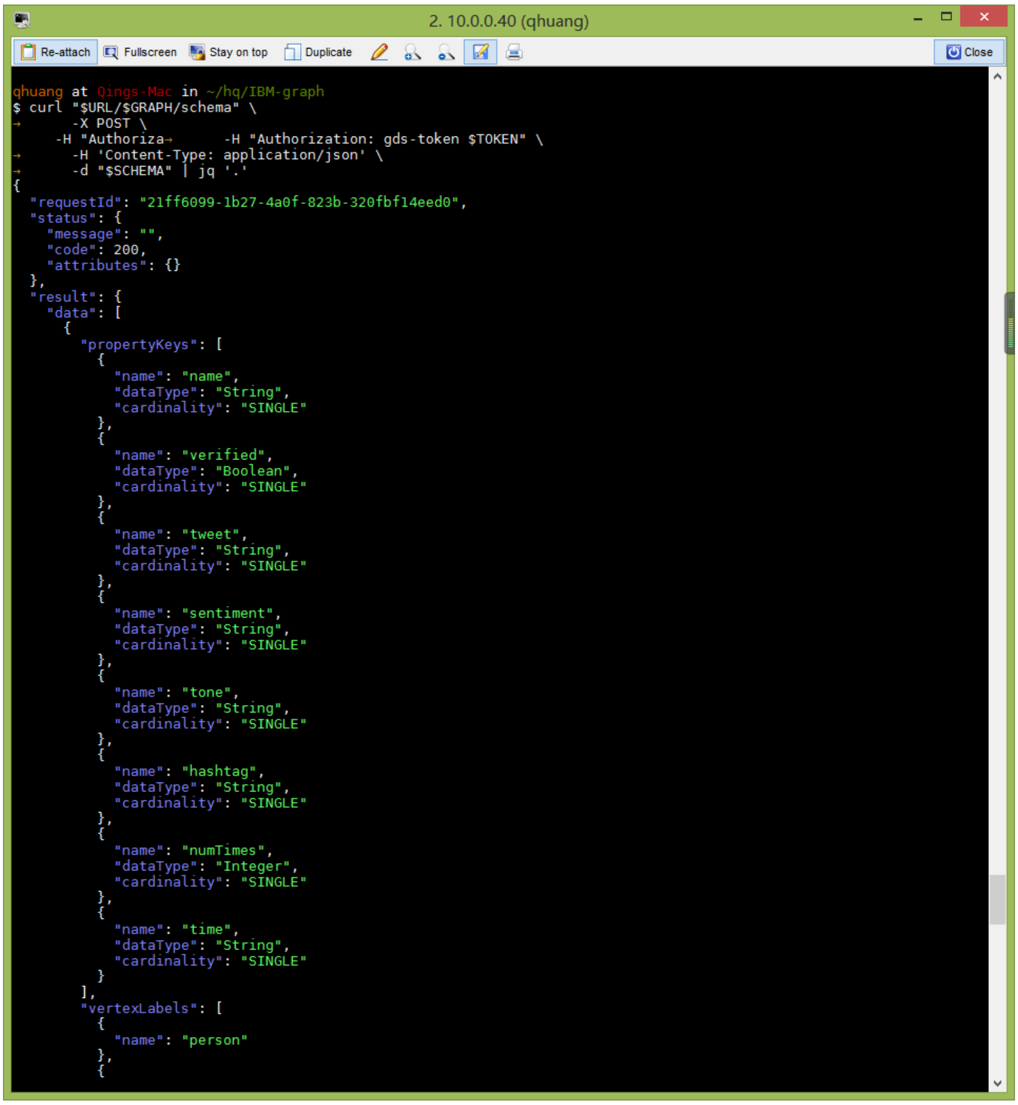


```
                Input Streams:
                -------------
                        1) From Object DB2INST1.EMPLOYEE

                                Estimated number of rows:       42
                                Number of columns:              4
                                Subquery predicate ID:          Not Applicable

                                Column Names:
                                ------------
                                +Q1.$RID$+Q1.SALARY+Q1.SEX+Q1.EDLEVEL


                Output Streams:
                --------------
                        2) To Operator #4

                                Estimated number of rows:       14
                                Number of columns:              2
                                Subquery predicate ID:          Not Applicable

                                Column Names:
                                ------------
                                +Q2.SALARY+Q2.SEX


Objects Used in Access Plan:
---------------------------

        Schema: DB2INST1
        Name:   EMP
        Type:   Alias (reference only)

        Schema: DB2INST1
        Name:   ADEFUSR
        Type:   Materialized View (reference only)

        Schema: DB2INST1
        Name:   EMPLOYEE
        Type:   Table
                        Time of creation:               2017-02-17-03.31.56.399206
                        Last statistics update:         2017-02-19-05.42.33.919849
                        Number of columns:              14
                        Number of rows:                 42
                        Width of rows:                  99
                        Number of buffer pool pages:    1
                        Number of data partitions:      1
                        Distinct row values:            No
```

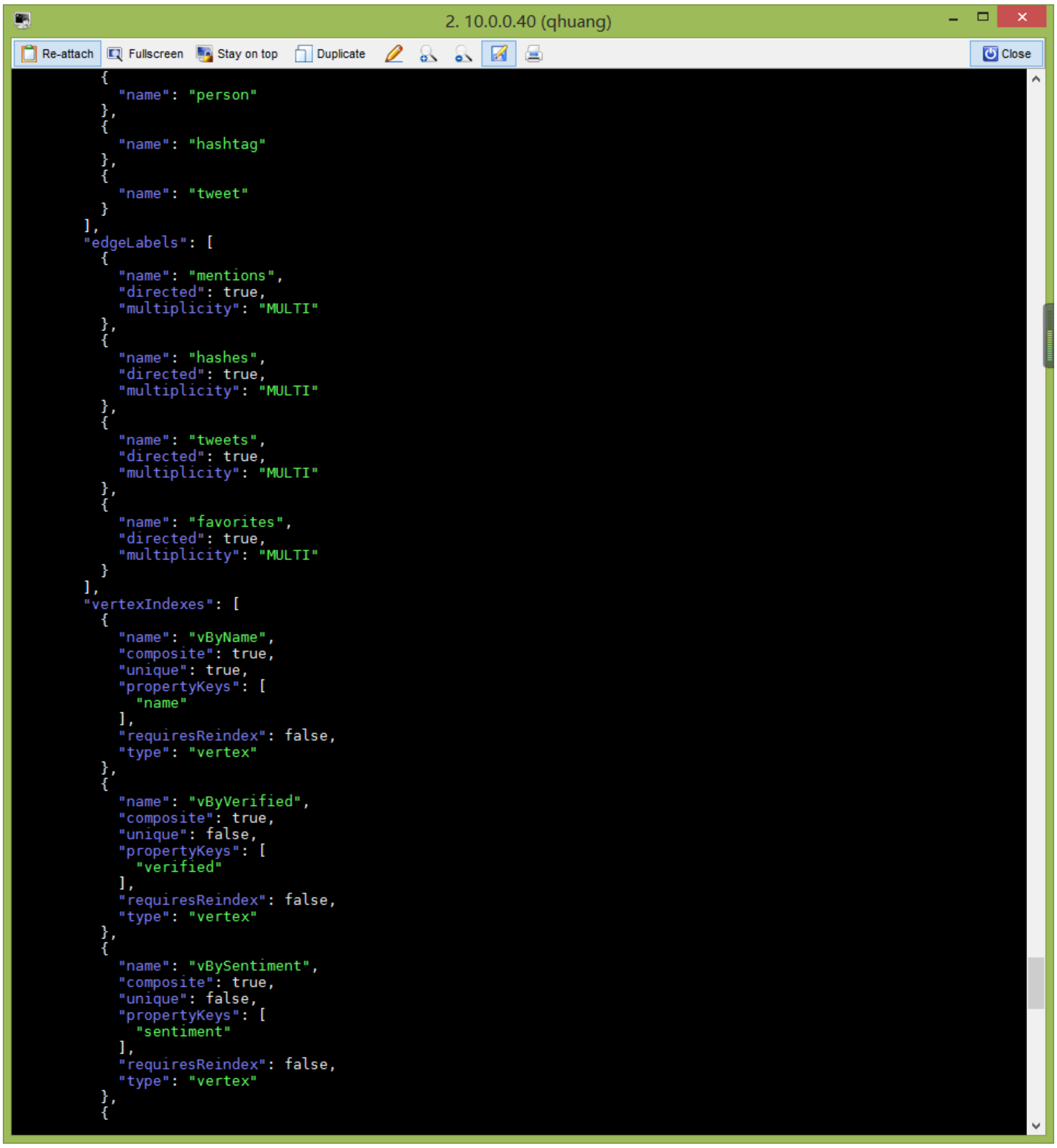


```
            Output Streams:
            --------------
                    2) To Operator #4

                            Estimated number of rows:       14
                            Number of columns:              2
                            Subquery predicate ID:          Not Applicable

                            Column Names:
                            ------------
                            +Q2.SALARY+Q2.SEX


Objects Used in Access Plan:
---------------------------

        Schema: DB2INST1
        Name:   EMP
        Type:   Alias (reference only)

        Schema: DB2INST1
        Name:   ADEFUSR
        Type:   Materialized View (reference only)

        Schema: DB2INST1
        Name:   EMPLOYEE
        Type:   Table
                        Time of creation:               2017-02-17-03.31.56.399206
                        Last statistics update:         2017-02-19-05.42.33.919849
                        Number of columns:              14
                        Number of rows:                 42
                        Width of rows:                  99
                        Number of buffer pool pages:    1
                        Number of data partitions:      1
                        Distinct row values:            No
                        Tablespace name:                USERSPACE1
                        Tablespace overhead:            6.725000
                        Tablespace transfer rate:       0.080000
                        Source for statistics:          Single Node
                        Prefetch page count:            32
                        Container extent page count:    32
                        Table overflow record count:    0
                        Table Active Blocks:            -1
                        Average Row Compression Ratio:  0
                        Percentage Rows Compressed:     0
                        Average Compressed Row Size:    0

Executing Connect Reset -- Connect Reset was Successful.
[db2inst1@2167e931c0d5 ~]$
```

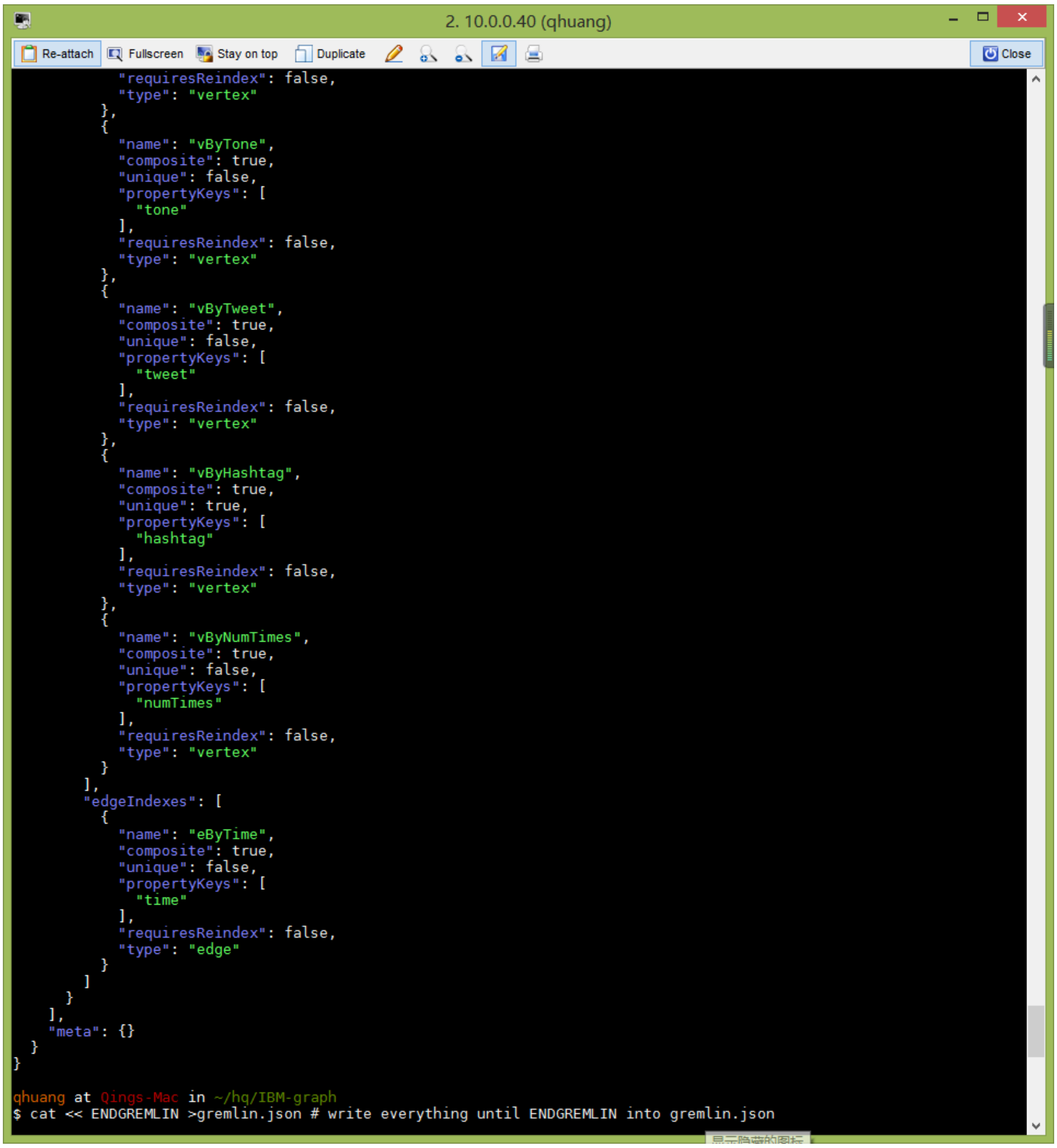
Graph Data store

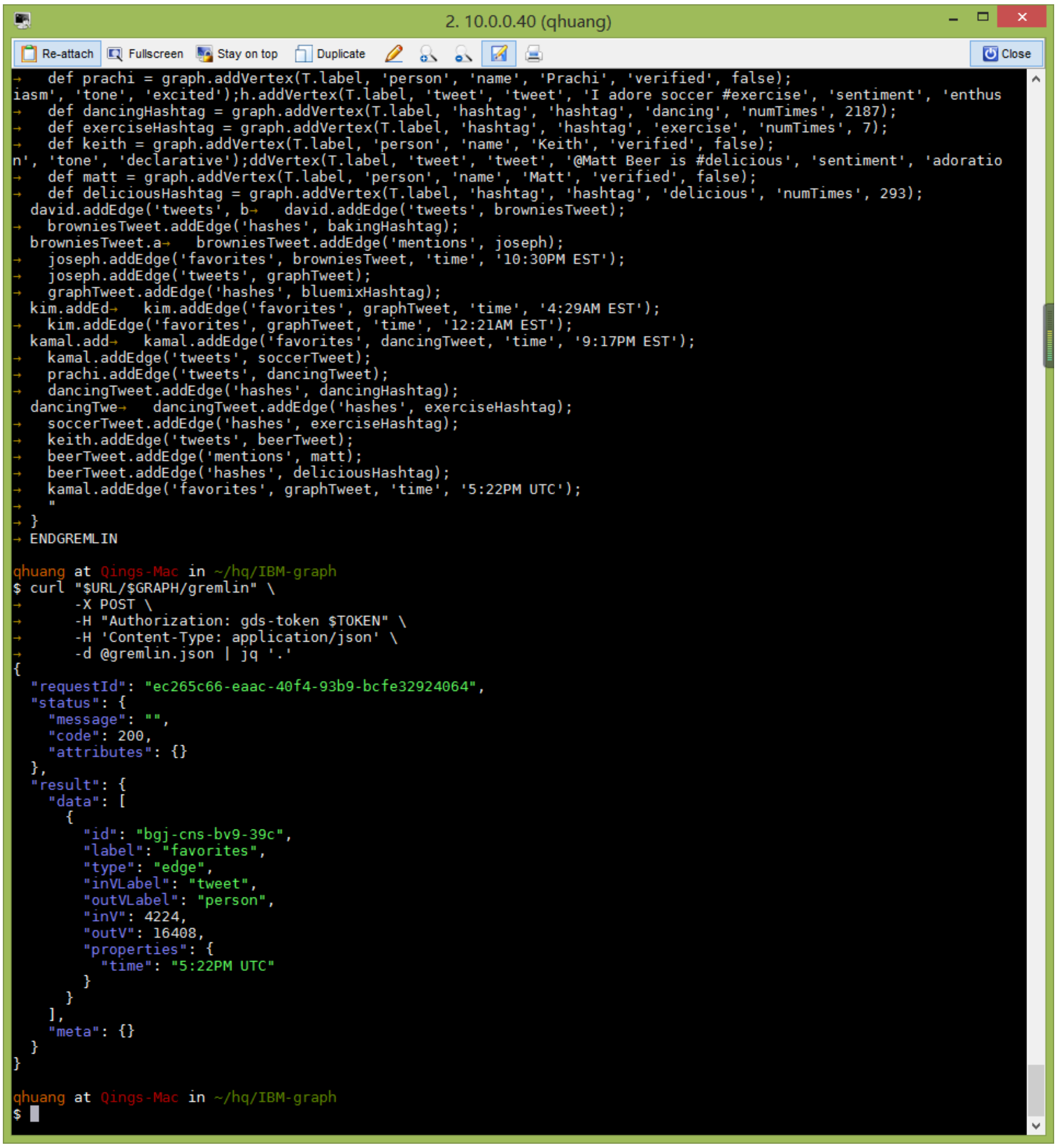


```
qhuang at Qings-Mac in ~/hq/IBM-graph
$ curl "$URL/$GRAPH/schema" \
→      -X POST \
     -H "Authoriza→      -H "Authorization: gds-token $TOKEN" \
→      -H 'Content-Type: application/json' \
→      -d "$SCHEMA" | jq '.'
{
  "requestId": "21ff6099-1b27-4a0f-823b-320fbf14eed0",
  "status": {
    "message": "",
    "code": 200,
    "attributes": {}
  },
  "result": {
    "data": [
      {
        "propertyKeys": [
          {
            "name": "name",
            "dataType": "String",
            "cardinality": "SINGLE"
          },
          {
            "name": "verified",
            "dataType": "Boolean",
            "cardinality": "SINGLE"
          },
          {
            "name": "tweet",
            "dataType": "String",
            "cardinality": "SINGLE"
          },
          {
            "name": "sentiment",
            "dataType": "String",
            "cardinality": "SINGLE"
          },
          {
            "name": "tone",
            "dataType": "String",
            "cardinality": "SINGLE"
          },
          {
            "name": "hashtag",
            "dataType": "String",
            "cardinality": "SINGLE"
          },
          {
            "name": "numTimes",
            "dataType": "Integer",
            "cardinality": "SINGLE"
          },
          {
            "name": "time",
            "dataType": "String",
            "cardinality": "SINGLE"
          }
        ],
        "vertexLabels": [
          {
            "name": "person"
          },
          {
```

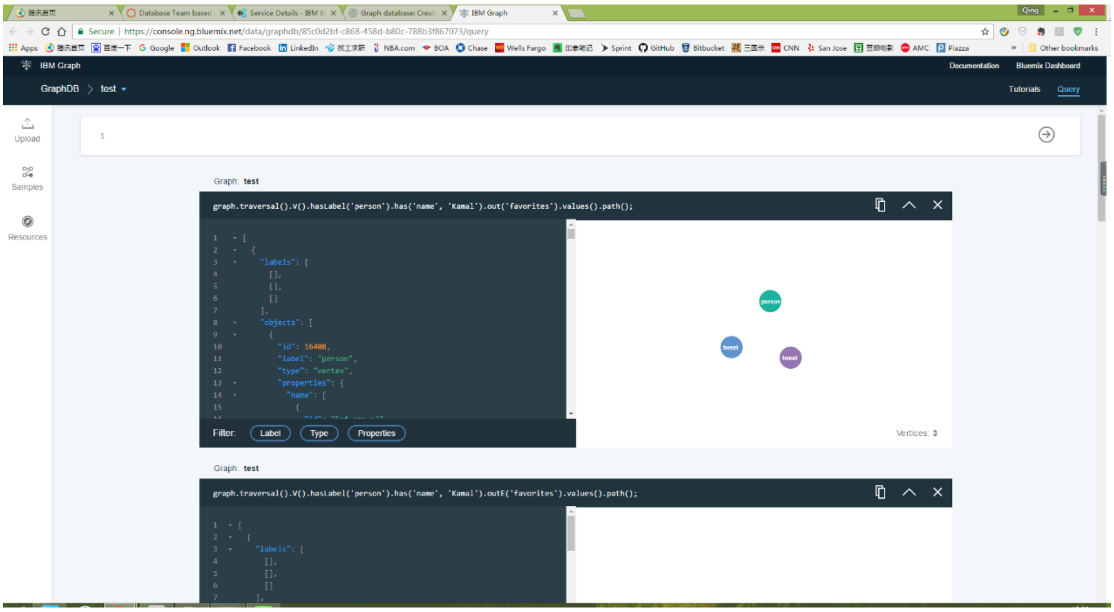


```
          {
            "name": "person"
          },
          {
            "name": "hashtag"
          },
          {
            "name": "tweet"
          }
        ],
        "edgeLabels": [
          {
            "name": "mentions",
            "directed": true,
            "multiplicity": "MULTI"
          },
          {
            "name": "hashes",
            "directed": true,
            "multiplicity": "MULTI"
          },
          {
            "name": "tweets",
            "directed": true,
            "multiplicity": "MULTI"
          },
          {
            "name": "favorites",
            "directed": true,
            "multiplicity": "MULTI"
          }
        ],
        "vertexIndexes": [
          {
            "name": "vByName",
            "composite": true,
            "unique": true,
            "propertyKeys": [
              "name"
            ],
            "requiresReindex": false,
            "type": "vertex"
          },
          {
            "name": "vByVerified",
            "composite": true,
            "unique": false,
            "propertyKeys": [
              "verified"
            ],
            "requiresReindex": false,
            "type": "vertex"
          },
          {
            "name": "vBySentiment",
            "composite": true,
            "unique": false,
            "propertyKeys": [
              "sentiment"
            ],
            "requiresReindex": false,
            "type": "vertex"
          },
          {
```





```
→   def prachi = graph.addVertex(T.label, 'person', 'name', 'Prachi', 'verified', false);
iasm', 'tone', 'excited');h.addVertex(T.label, 'tweet', 'tweet', 'I adore soccer #exercise', 'sentiment', 'enthus
→   def dancingHashtag = graph.addVertex(T.label, 'hashtag', 'hashtag', 'dancing', 'numTimes', 2187);
→   def exerciseHashtag = graph.addVertex(T.label, 'hashtag', 'hashtag', 'exercise', 'numTimes', 7);
→   def keith = graph.addVertex(T.label, 'person', 'name', 'Keith', 'verified', false);
n', 'tone', 'declarative');ddVertex(T.label, 'tweet', 'tweet', '@Matt Beer is #delicious', 'sentiment', 'adoratio
→   def matt = graph.addVertex(T.label, 'person', 'name', 'Matt', 'verified', false);
→   def deliciousHashtag = graph.addVertex(T.label, 'hashtag', 'hashtag', 'delicious', 'numTimes', 293);
  david.addEdge('tweets', b→   david.addEdge('tweets', browniesTweet);
→   browniesTweet.addEdge('hashes', bakingHashtag);
  browniesTweet.a→   browniesTweet.addEdge('mentions', joseph);
→   joseph.addEdge('favorites', browniesTweet, 'time', '10:30PM EST');
→   joseph.addEdge('tweets', graphTweet);
→   graphTweet.addEdge('hashes', bluemixHashtag);
  kim.addEd→   kim.addEdge('favorites', graphTweet, 'time', '4:29AM EST');
→   kim.addEdge('favorites', graphTweet, 'time', '12:21AM EST');
  kamal.add→   kamal.addEdge('favorites', dancingTweet, 'time', '9:17PM EST');
→   kamal.addEdge('tweets', soccerTweet);
→   prachi.addEdge('tweets', dancingTweet);
→   dancingTweet.addEdge('hashes', dancingHashtag);
  dancingTwe→   dancingTweet.addEdge('hashes', exerciseHashtag);
→   soccerTweet.addEdge('hashes', exerciseHashtag);
→   keith.addEdge('tweets', beerTweet);
→   beerTweet.addEdge('mentions', matt);
→   beerTweet.addEdge('hashes', deliciousHashtag);
→   kamal.addEdge('favorites', graphTweet, 'time', '5:22PM UTC');
→   "
→ }
→ ENDGREMLIN

qhuang at Qings-Mac in ~/hq/IBM-graph
$ curl "$URL/$GRAPH/gremlin" \
→      -X POST \
→      -H "Authorization: gds-token $TOKEN" \
→      -H 'Content-Type: application/json' \
→      -d @gremlin.json | jq '.'
{
  "requestId": "ec265c66-eaac-40f4-93b9-bcfe32924064",
  "status": {
    "message": "",
    "code": 200,
    "attributes": {}
  },
  "result": {
    "data": [
      {
        "id": "bgj-cns-bv9-39c",
        "label": "favorites",
        "type": "edge",
        "inVLabel": "tweet",
        "outVLabel": "person",
        "inV": 4224,
        "outV": 16408,
        "properties": {
          "time": "5:22PM UTC"
        }
      }
    ],
    "meta": {}
  }
}

qhuang at Qings-Mac in ~/hq/IBM-graph
$ 
```



Graph: test

```
graph.traversal().V().hasLabel('person').has('name', 'Kamal').out('favorites').values().path();
```

```
[
  {
    "labels": [
      [],
      [],
      []
    ],
    "objects": [
      {
        "id": 16408,
        "label": "person",
        "type": "vertex",
        "properties": {
          "name": [
            {
```

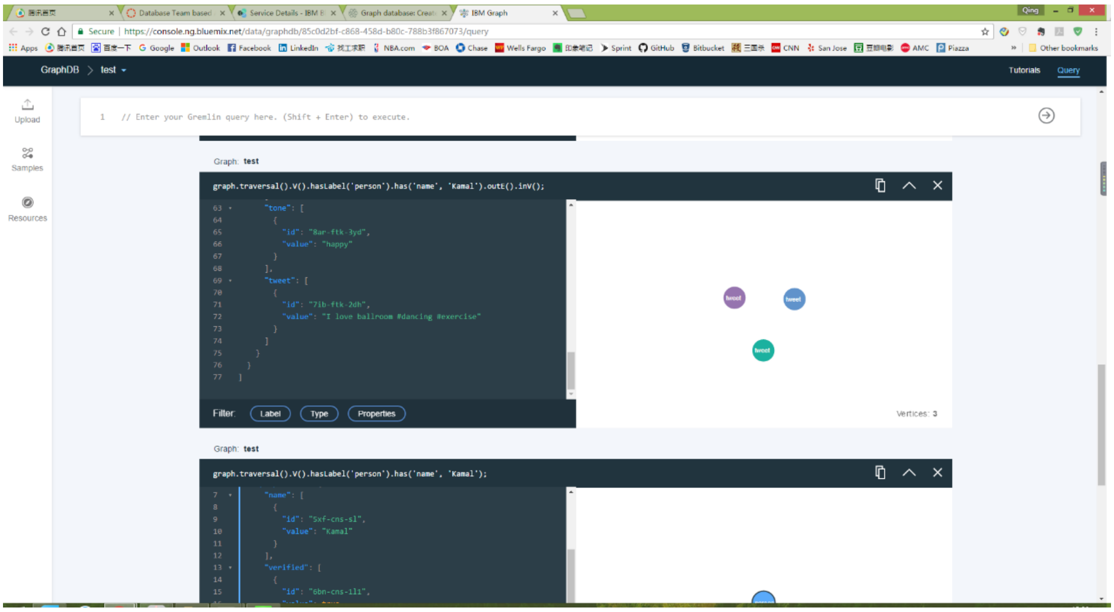
Filter: Label Type Properties

Vertices: 3

Graph: test

```
graph.traversal().V().hasLabel('person').has('name', 'Kamal').outE('favorites').values().path();
```

```
[
  {
    "labels": [
      [],
      [],
      []
    ],
```



```
1   // Enter your Gremlin query here. (Shift + Enter) to execute.
```

Graph: test

```
graph.traversal().V().hasLabel('person').has('name', 'Kamal').outE().inV();
```

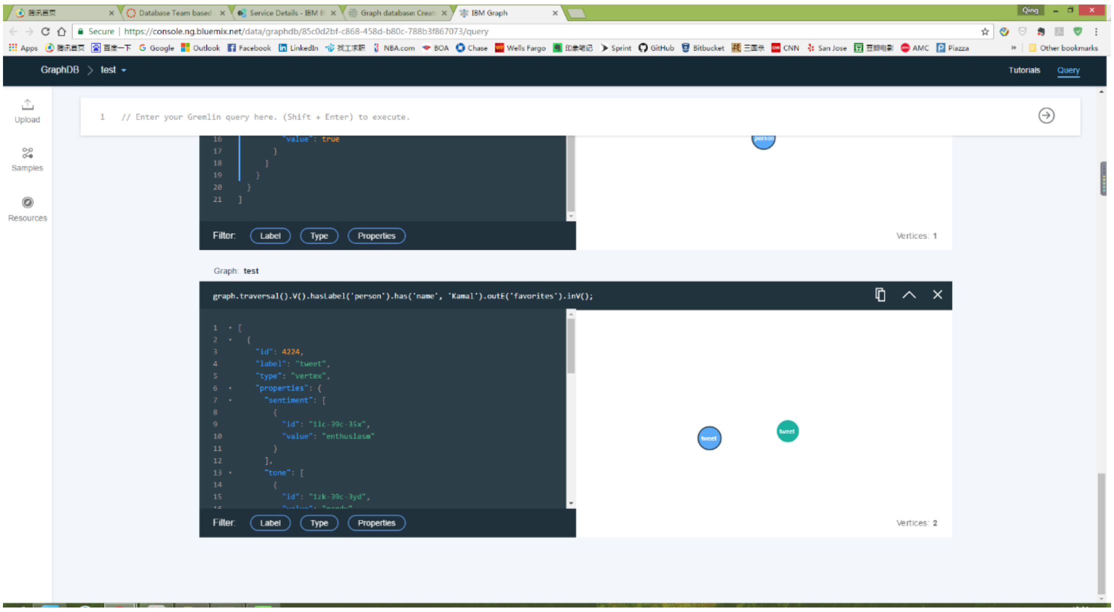
```
        "tone": [
          {
            "id": "8ar-ftk-3yd",
            "value": "happy"
          }
        ],
        "tweet": [
          {
            "id": "7ib-ftk-2dh",
            "value": "I love ballroom #dancing #exercise"
          }
        ]
      }
    }
  ]
```

Filter: Label Type Properties

Graph: test

```
graph.traversal().V().hasLabel('person').has('name', 'Kamal');
```

```
        "name": [
          {
            "id": "5xf-cns-sl",
            "value": "Kamal"
          }
        ],
        "verified": [
          {
            "id": "6bn-cns-1l1",
```



```
1   // Enter your Gremlin query here. (Shift + Enter) to execute.
```

```
            "value": true
          }
        ]
      }
    }
  ]
```

Filter: Label Type Properties

Graph: test

```
graph.traversal().V().hasLabel('person').has('name', 'Kamal').outE('favorites').inV();
```

```
  [
    {
      "id": 4224,
      "label": "tweet",
      "type": "vertex",
      "properties": {
        "sentiment": [
          {
            "id": "11c-39c-35x",
            "value": "enthusiasm"
          }
        ],
        "tone": [
          {
            "id": "1zk-39c-3yd",
```

Filter: Label Type Properties