Data Analysis and Integration Project

2022/2023

Alameda – Group **07**

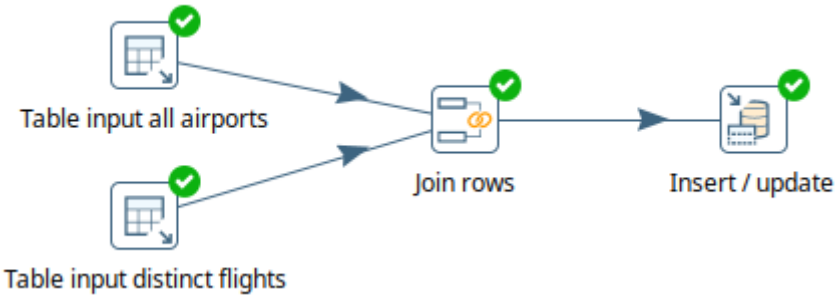
Joana Raposo, **92485** Rodrigo Gomes, **92548**

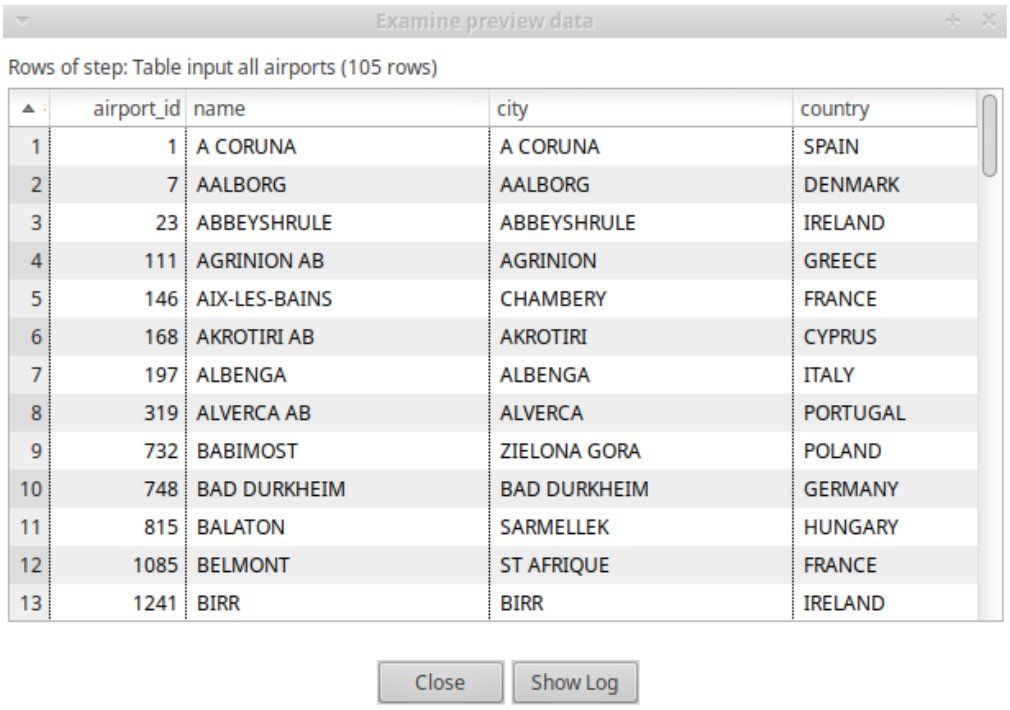
A) SQL instructions to create the data warehouse tables:

```
DROP DATABASE IF EXISTS airports_dw;

CREATE DATABASE airports_dw;

USE airports_dw;

CREATE TABLE dim_from_airport (
    from_airport_id INT,
    from_airport_name VARCHAR(50),
    from_airport_city VARCHAR(100),
    from_airport_country VARCHAR(100),
    PRIMARY KEY (from_airport_id)
);

CREATE TABLE dim_to_airport (
    to_airport_id INT,
    to_airport_name VARCHAR(50),
    to_airport_city VARCHAR(100),
    to_airport_country VARCHAR(100),
    PRIMARY KEY (to_airport_id)
);


CREATE TABLE dim_from_time (
    from_time_id DATETIME,
    day INT,
    month INT,
    year INT,
    PRIMARY KEY (from_time_id)
);

CREATE TABLE dim_to_time (
    to_time_id DATETIME,
    day INT,
    month INT,
    year INT,
    PRIMARY KEY (to_time_id)
);

CREATE TABLE dim_airplane (
    airplane_id INT,
    type_id INT,
    PRIMARY KEY (airplane_id)
);

CREATE TABLE dim_airline (
    airline_id INT,
    airline_name VARCHAR(30),
    PRIMARY KEY (airline_id)
);


CREATE TABLE fact_flight (
    flight_id INT,
    passenger_no INT,
    total_revenue DOUBLE,
    from_airport_id INT,
    to_airport_id INT,
    from_time_id DATETIME,
    to_time_id DATETIME,
    airline_id INT,
    airplane_id INT,
    PRIMARY KEY (flight_id),
    FOREIGN KEY (from_airport_id) REFERENCES dim_from_airport (from_airport_id),
    FOREIGN KEY (to_airport_id) REFERENCES dim_to_airport (to_airport_id),
    FOREIGN KEY (from_time_id) REFERENCES dim_from_time (from_time_id),
    FOREIGN KEY (to_time_id) REFERENCES dim_to_time (to_time_id),
    FOREIGN KEY (airline_id) REFERENCES dim_airline (airline_id),
    FOREIGN KEY (airplane_id) REFERENCES dim_airplane (airplane_id)
);
```

B) Transformation screenshots:

1- <u>Airport dimension for airport of origin</u>




Examine preview data
Rows of step: Table input all airports (105 rows)
airport_id name city country
1 1 A CORUNA A CORUNA SPAIN
2 7 AALBORG AALBORG DENMARK
3 23 ABBEYSHRULE ABBEYSHRULE IRELAND
4 111 AGRINION AB AGRINION GREECE
5 146 AIX-LES-BAINS CHAMBERY FRANCE
6 168 AKROTIRI AB AKROTIRI CYPRUS
7 197 ALBENGA ALBENGA ITALY
8 319 ALVERCA AB ALVERCA PORTUGAL
9 732 BABIMOST ZIELONA GORA POLAND
10 748 BAD DURKHEIM BAD DURKHEIM GERMANY
11 815 BALATON SARMELLEK HUNGARY
12 1085 BELMONT ST AFRIQUE FRANCE
13 1241 BIRR BIRR IRELAND
Close
Show Log

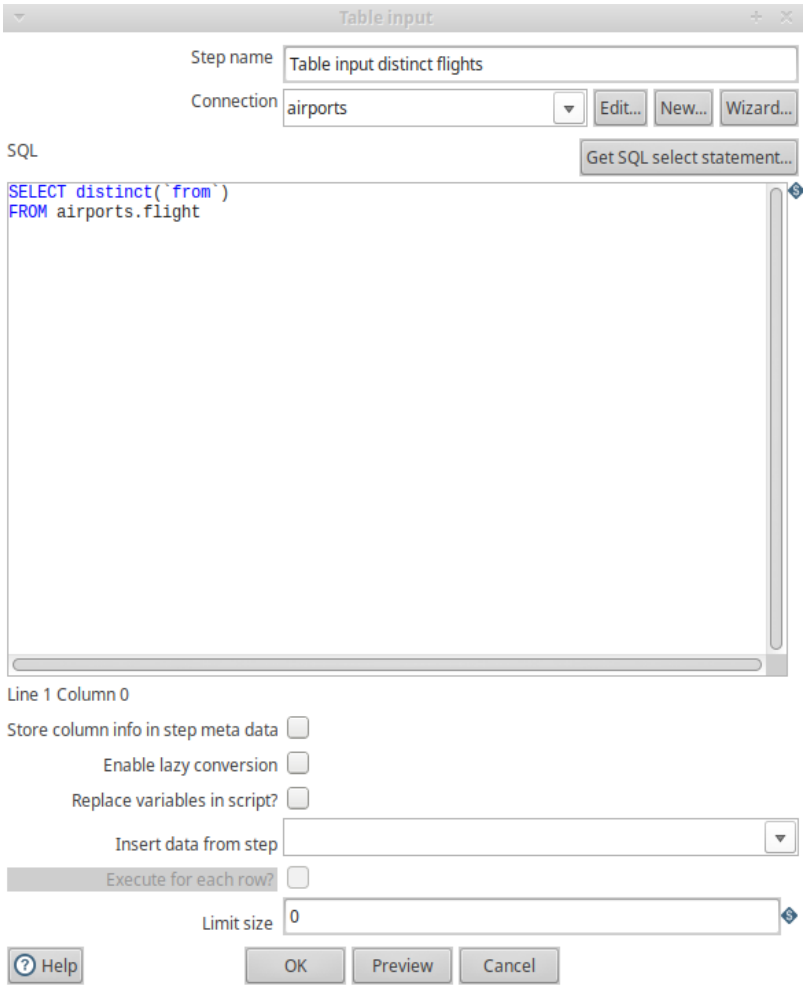
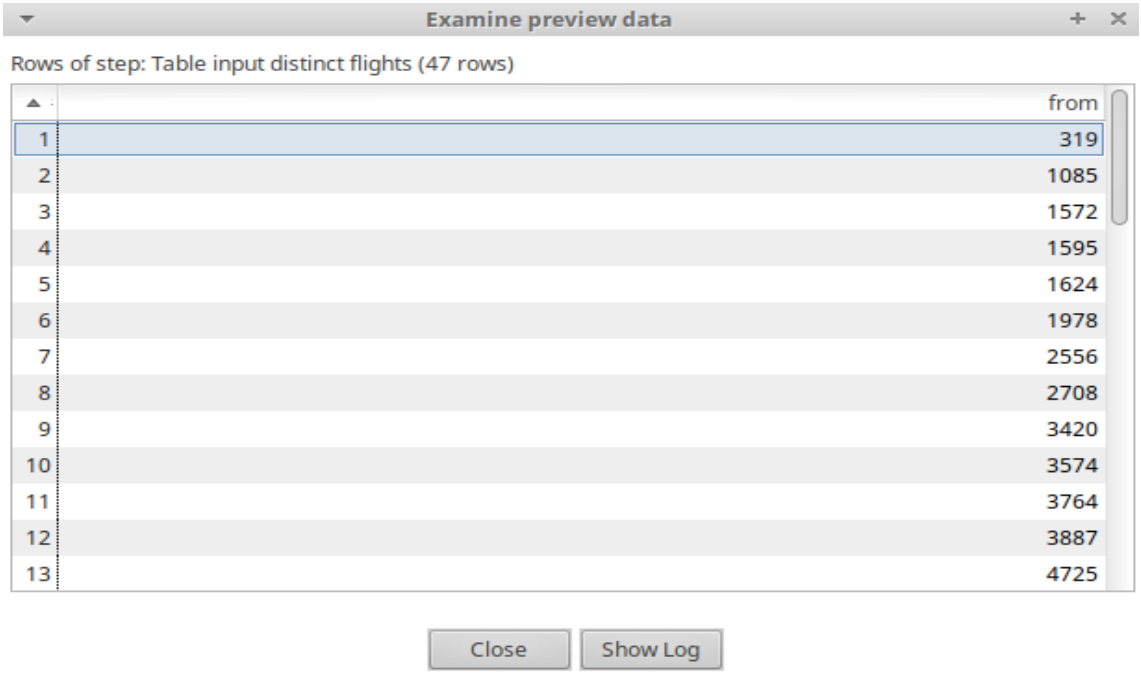
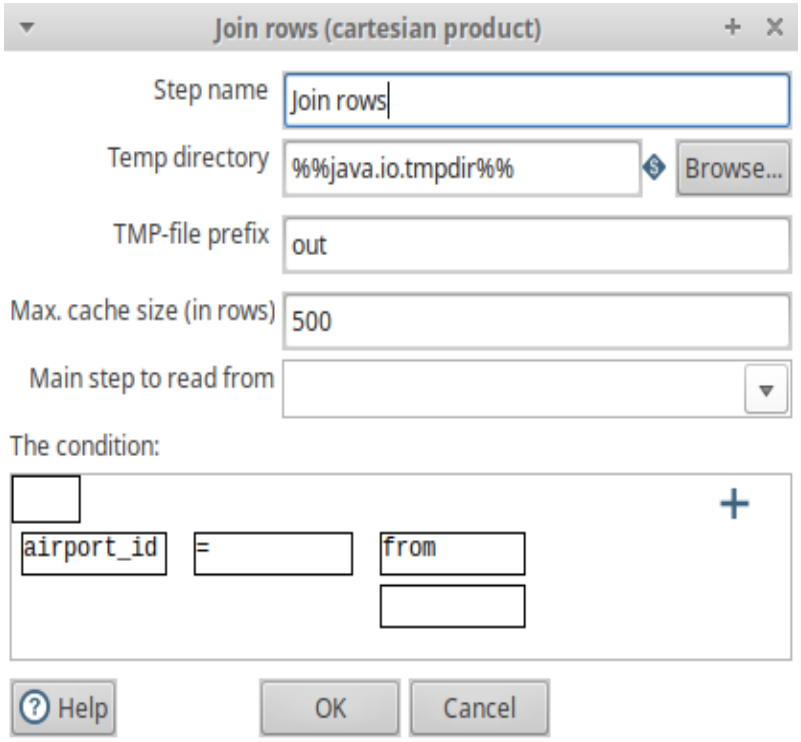


Table input
Step name
Table input distinct flights
Connection
airports
Edit...
New...
Wizard...
SQL
Get SQL select statement...
SELECT distinct(`from`)
FROM airports.flight
Line 1 Column 0
Store column info in step meta data
Enable lazy conversion
Replace variables in script?
Insert data from step
Execute for each row?
Limit size
0
Help
OK
Preview
Cancel


Examine preview data
Rows of step: Table input distinct flights (47 rows)
from
1 319
2 1085
3 1572
4 1595
5 1624
6 1978
7 2556
8 2708
9 3420
10 3574
11 3764
12 3887
13 4725
Close
Show Log


Join rows (cartesian product)
Step name
Join rows
Temp directory
%%java.io.tmpdir%%
Browse...
TMP-file prefix
out
Max. cache size (in rows)
500
Main step to read from
The condition:
airport_id
=
from
Help
OK
Cancel

Examine preview data

Rows of step: Join rows (47 rows)

	airport_id	name	city	country	from
1	319	ALVERCA AB	ALVERCA	PORTUGAL	319
2	1085	BELMONT	ST AFRIQUE	FRANCE	1085
3	1572	BRON	LYON	FRANCE	1572
4	1595	BROUMOV	BROUMOV	CZECH	1595
5	1624	BUCKEBURG ARMY	BUCKEBURG	GERMANY	1624
6	1978	CARPI BUDRIONE	CARPI BUDRIONE	ITALY	1978
7	2556	COLTINES	ST FLOUR	FRANCE	2556
8	2708	COTTBUS ARMY	COTTBUS	GERMANY	2708
9	3420	EICHSTATT	EICHSTATT	GERMANY	3420
10	3574	ELZ	ELZ	GERMANY	3574
11	3764	FALKOPING	FALKOPING	SWEDEN	3764
12	3887	FIUMICINO	ROME	ITALY	3887
13	4725	GUIDONIA MIL	GUIDONIA	ITALY	4725

Close

Insert / update

Step name: Insert / update

Connection: airports_dw [Edit...] [New...] [Wizard...]

Target schema: airports_dw [Browse...]

Target table: dim_from_airport [Browse...]

Commit size: 100

Don't perform any updates:

The key(s) to look up the value(s):

	Table field	Comparator	Stream field1	Stream field2
1	from_airport_id	=	airport_id	

[Get fields]

Update fields:

	Table field	Stream field	Update
1	from_airport_id	airport_id	Y
2	from_airport_name	name	Y
3	from_airport_city	city	Y
4	from_airport_country	country	Y

[Get update fields] [Edit mapping]

[Help] [OK] [Cancel] [SQL]

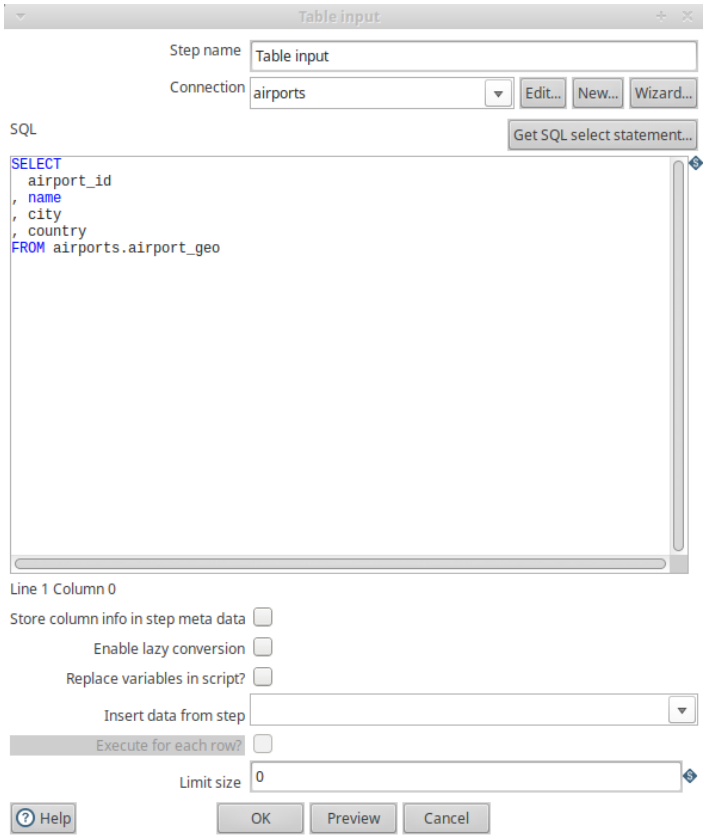
Examine preview data

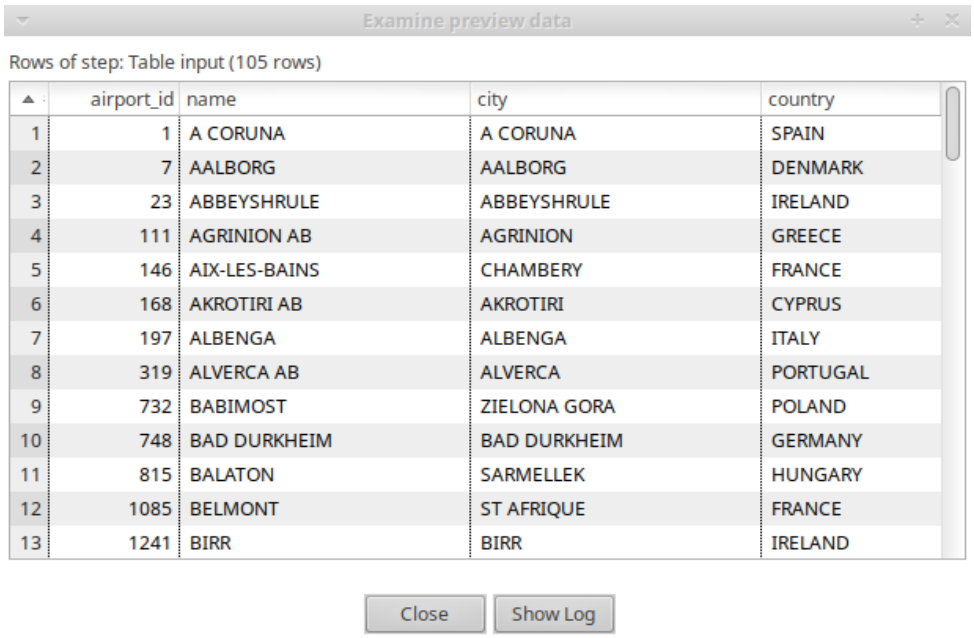
Rows of step: Insert / update (47 rows)

	airport_id	name	city	country	from
1	319	ALVERCA AB	ALVERCA	PORTUGAL	319
2	1085	BELMONT	ST AFRIQUE	FRANCE	1085
3	1572	BRON	LYON	FRANCE	1572
4	1595	BROUMOV	BROUMOV	CZECH	1595
5	1624	BUCKEBURG ARMY	BUCKEBURG	GERMANY	1624
6	1978	CARPI BUDRIONE	CARPI BUDRIONE	ITALY	1978
7	2556	COLTINES	ST FLOUR	FRANCE	2556
8	2708	COTTBUS ARMY	COTTBUS	GERMANY	2708
9	3420	EICHSTATT	EICHSTATT	GERMANY	3420
10	3574	ELZ	ELZ	GERMANY	3574
11	3764	FALKOPING	FALKOPING	SWEDEN	3764
12	3887	FIUMICINO	ROME	ITALY	3887
13	4725	GUIDONIA MIL	GUIDONIA	ITALY	4725

Close

2- Airport dimension for airport of destination




Examine preview data
Rows of step: Table input (105 rows)
airport_id
name
city
country
1 1 A CORUNA A CORUNA SPAIN
2 7 AALBORG AALBORG DENMARK
3 23 ABBEYSHRULE ABBEYSHRULE IRELAND
4 111 AGRINION AB AGRINION GREECE
5 146 AIX-LES-BAINS CHAMBERY FRANCE
6 168 AKROTIRI AB AKROTIRI CYPRUS
7 197 ALBENGA ALBENGA ITALY
8 319 ALVERCA AB ALVERCA PORTUGAL
9 732 BABIMOST ZIELONA GORA POLAND
10 748 BAD DURKHEIM BAD DURKHEIM GERMANY
11 815 BALATON SARMELLEK HUNGARY
12 1085 BELMONT ST AFRIQUE FRANCE
13 1241 BIRR BIRR IRELAND
Close
Show Log

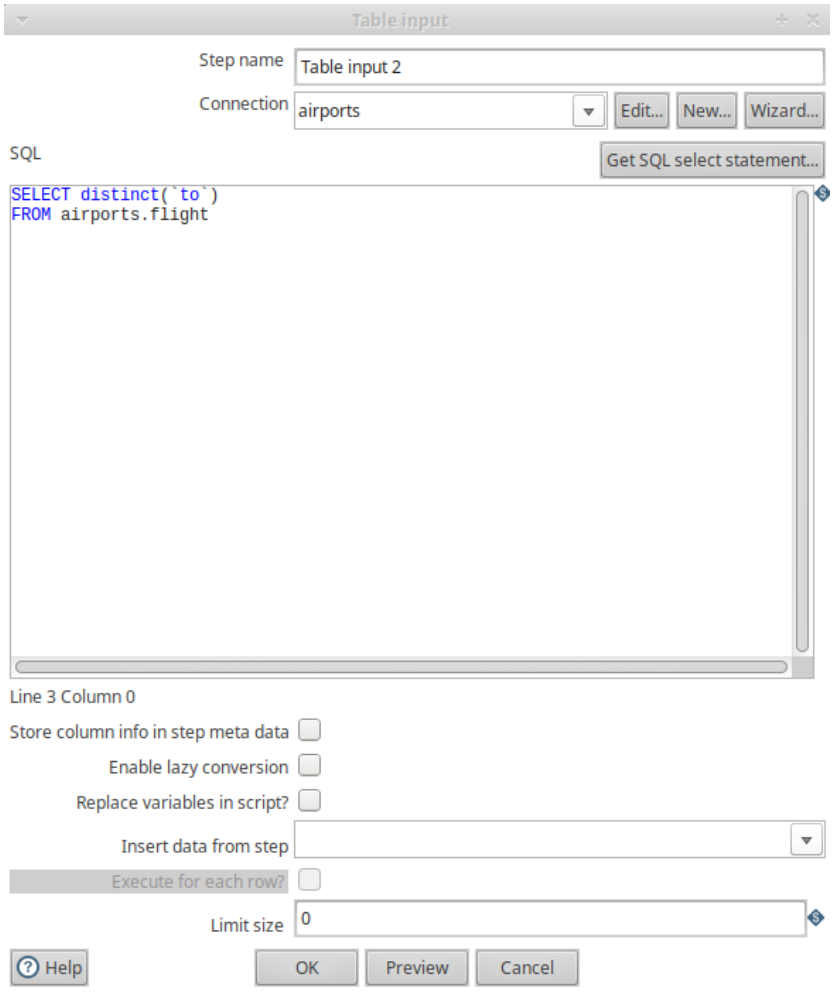
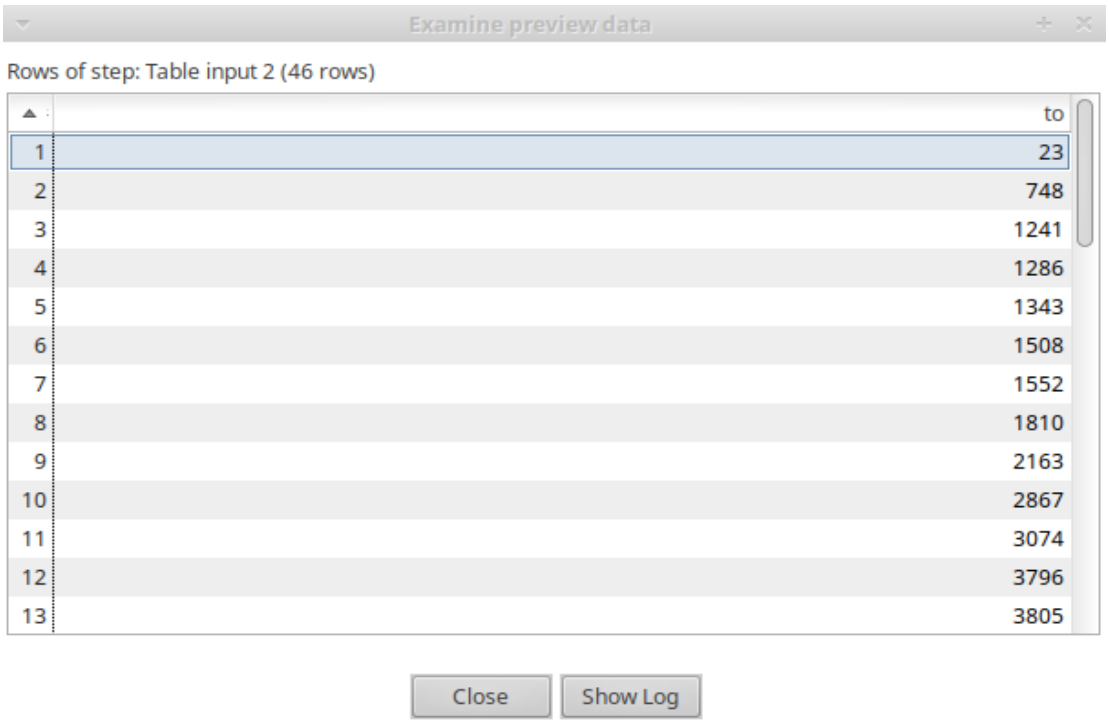
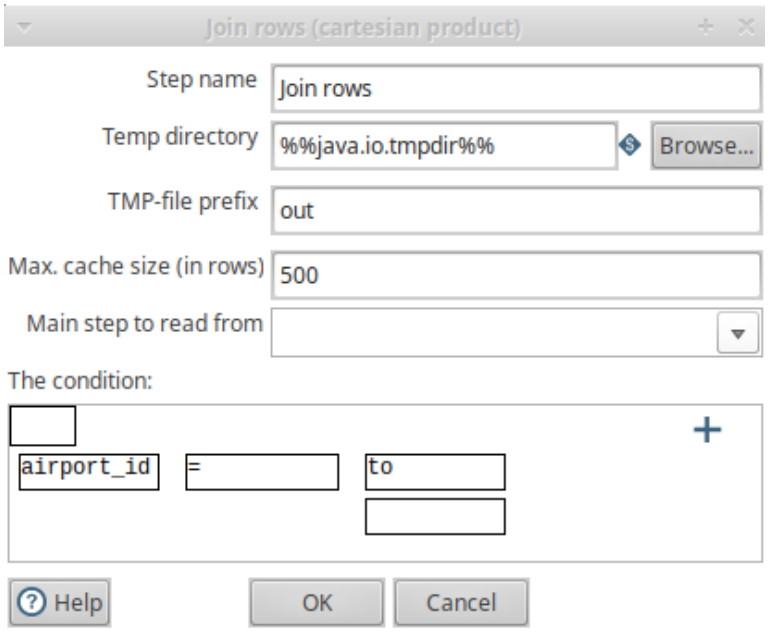


Table input
Step name
Table input 2
Connection
airports
Edit...
New...
Wizard...
SQL
Get SQL select statement...
SELECT distinct(`to`)
FROM airports.flight
Line 3 Column 0
Store column info in step meta data
Enable lazy conversion
Replace variables in script?
Insert data from step
Execute for each row?
Limit size
0
Help
OK
Preview
Cancel


Examine preview data
Rows of step: Table input 2 (46 rows)
to
1 23
2 748
3 1241
4 1286
5 1343
6 1508
7 1552
8 1810
9 2163
10 2867
11 3074
12 3796
13 3805
Close
Show Log


Join rows (cartesian product)
Step name
Join rows
Temp directory
%%java.io.tmpdir%%
Browse...
TMP-file prefix
out
Max. cache size (in rows)
500
Main step to read from
The condition:
airport_id
=
to
Help
OK
Cancel

Examine preview data

Rows of step: Join rows (46 rows)

	airport_id	name	city	country	to
1	23	ABBEYSHRULE	ABBEYSHRULE	IRELAND	23
2	748	BAD DURKHEIM	BAD DURKHEIM	GERMANY	748
3	1241	BIRR	BIRR	IRELAND	1241
4	1286	BLAUBEUREN	BLAUBEUREN	GERMANY	1286
5	1343	BOHUNOVICE	BOHUNOVICE	CZECH	1343
6	1508	BRASSCHAAT ARMY	BRASSCHAAT	BELGIUM	1508
7	1552	BRILON/ HOCHSAUERLAND	BRILON/ HOCHSAUERLAND	GERMANY	1552
8	1810	CAMERI MIL	CAMERI	ITALY	1810
9	2163	CHAMPAGNE AB	REIMS	FRANCE	2163
10	2867	DACHAU-GROBENRIED	DACHAU-GROBENRIED	GERMANY	2867
11	3074	DESSAU	DESSAU	GERMANY	3074
12	3796	FARO	FARO	PORTUGAL	3796
13	3805	FAYENCE	FAYENCE	FRANCE	3805

Close

Insert / update

Step name: Insert / update

Connection: airports_dw [Edit...] [New...] [Wizard...]

Target schema: airports_dw [Browse...]

Target table: dim_to_airport [Browse...]

Commit size: 100

Don't perform any updates: ☐

The key(s) to look up the value(s):

	Table field	Comparator	Stream field1	Stream field2
1	to_airport_id	=	airport_id	

[Get fields]

Update fields:

	Table field	Stream field	Update
1	to_airport_id	airport_id	Y
2	to_airport_name	name	Y
3	to_airport_city	city	Y
4	to_airport_country	country	Y

[Get update fields] [Edit mapping]

[Help] [OK] [Cancel] [SQL]

Examine preview data

Rows of step: Insert / update (46 rows)

	airport_id	name	city	country	to
1	23	ABBEYSHRULE	ABBEYSHRULE	IRELAND	23
2	748	BAD DURKHEIM	BAD DURKHEIM	GERMANY	748
3	1241	BIRR	BIRR	IRELAND	1241
4	1286	BLAUBEUREN	BLAUBEUREN	GERMANY	1286
5	1343	BOHUNOVICE	BOHUNOVICE	CZECH	1343
6	1508	BRASSCHAAT ARMY	BRASSCHAAT	BELGIUM	1508
7	1552	BRILON/ HOCHSAUERLAND	BRILON/ HOCHSAUERLAND	GERMANY	1552
8	1810	CAMERI MIL	CAMERI	ITALY	1810
9	2163	CHAMPAGNE AB	REIMS	FRANCE	2163
10	2867	DACHAU-GROBENRIED	DACHAU-GROBENRIED	GERMANY	2867
11	3074	DESSAU	DESSAU	GERMANY	3074
12	3796	FARO	FARO	PORTUGAL	3796
13	3805	FAYENCE	FAYENCE	FRANCE	3805

Close

3- Time dimension for date/time of departure

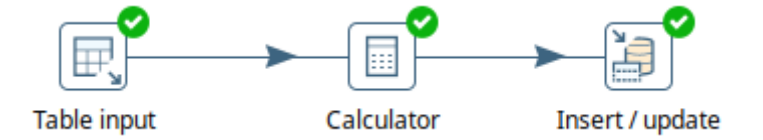


Table input

Step name: Table input

Connection: airports — Edit... New... Wizard...

SQL — Get SQL select statement...

```
SELECT departure FROM airports.flight
```

Line 1 Column 0

Store column info in step meta data

Enable lazy conversion

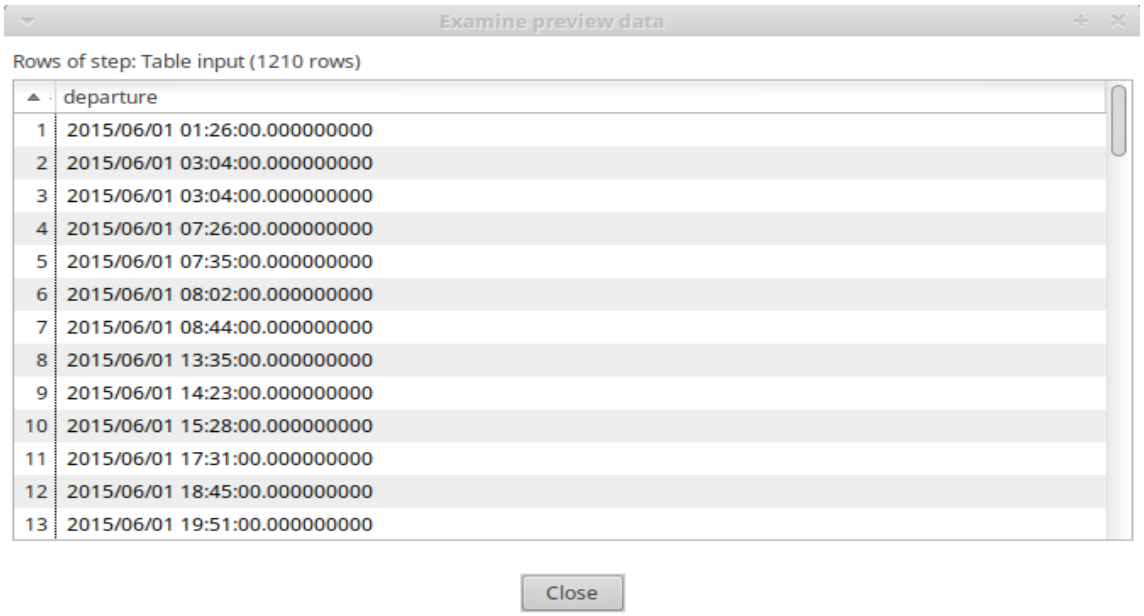
Replace variables in script?

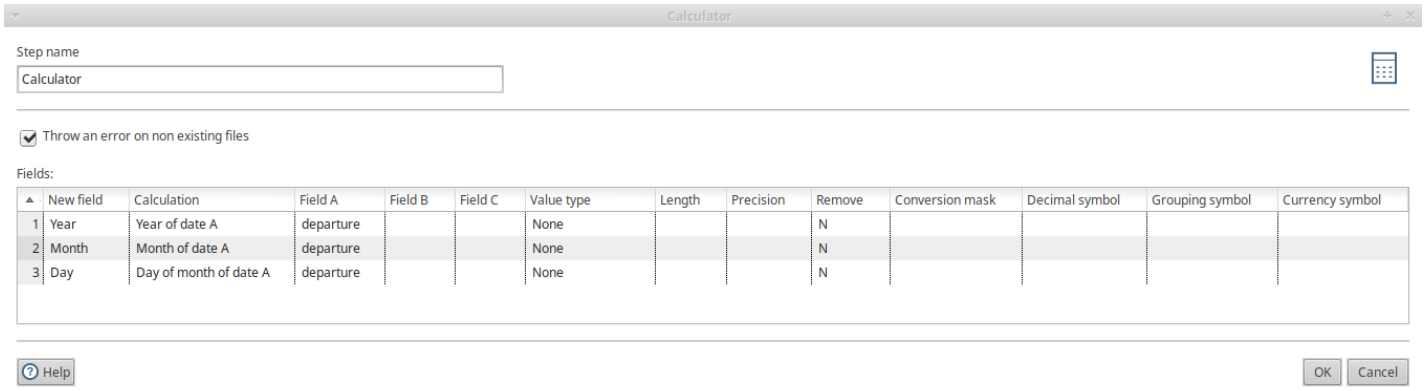
Insert data from step

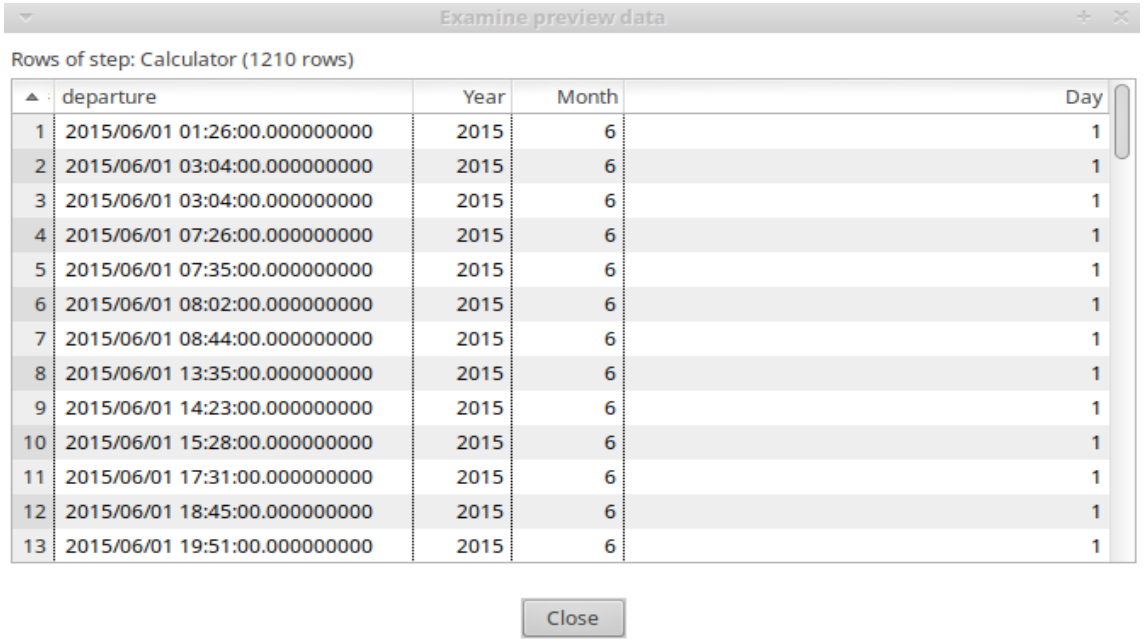
Execute for each row?

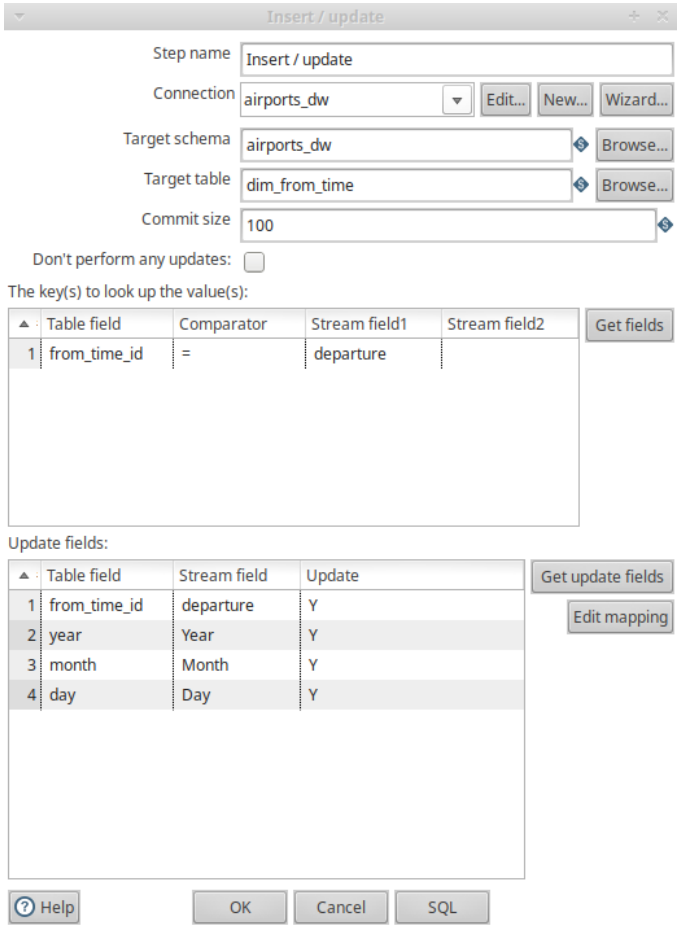
Limit size: 0

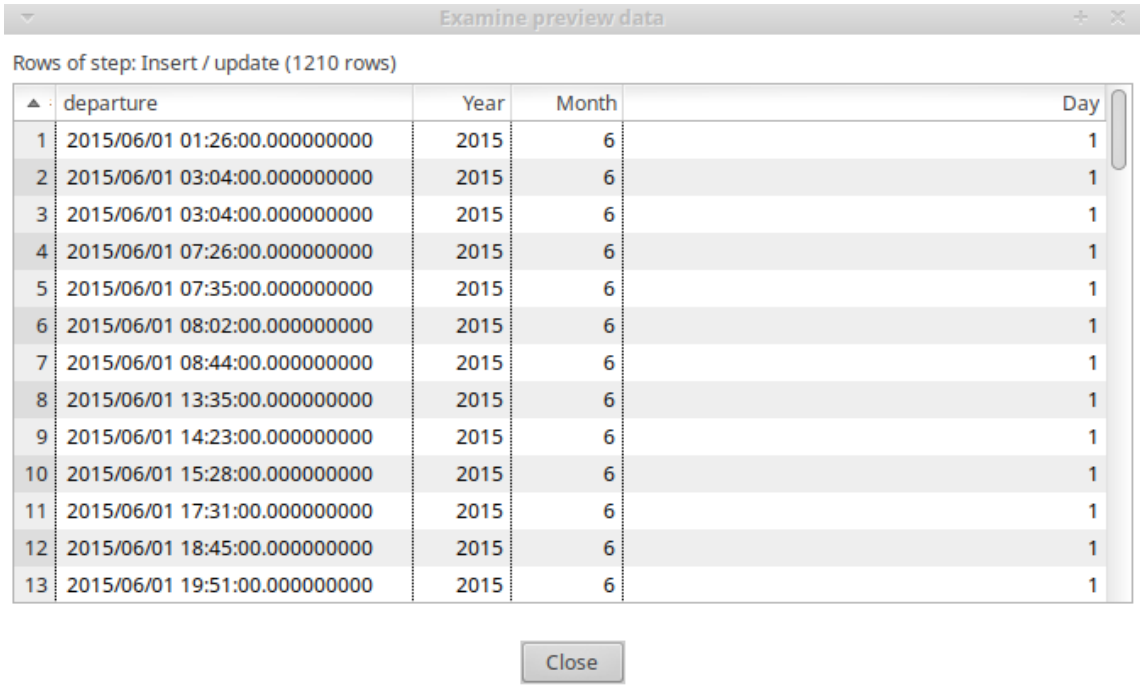
Help OK Preview Cancel


Examine preview data
Rows of step: Table input (1210 rows)
departure
1 2015/06/01 01:26:00.000000000
2 2015/06/01 03:04:00.000000000
3 2015/06/01 03:04:00.000000000
4 2015/06/01 07:26:00.000000000
5 2015/06/01 07:35:00.000000000
6 2015/06/01 08:02:00.000000000
7 2015/06/01 08:44:00.000000000
8 2015/06/01 13:35:00.000000000
9 2015/06/01 14:23:00.000000000
10 2015/06/01 15:28:00.000000000
11 2015/06/01 17:31:00.000000000
12 2015/06/01 18:45:00.000000000
13 2015/06/01 19:51:00.000000000
Close

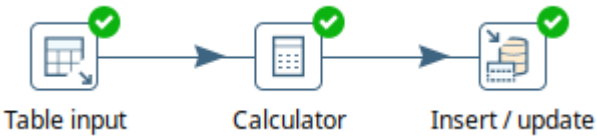

Calculator
Step name
Calculator
Throw an error on non existing files
Fields:
New field | Calculation | Field A | Field B | Field C | Value type | Length | Precision | Remove | Conversion mask | Decimal symbol | Grouping symbol | Currency symbol
1 Year | Year of date A | departure | | | None | | | N
2 Month | Month of date A | departure | | | None | | | N
3 Day | Day of month of date A | departure | | | None | | | N
Help
OK
Cancel

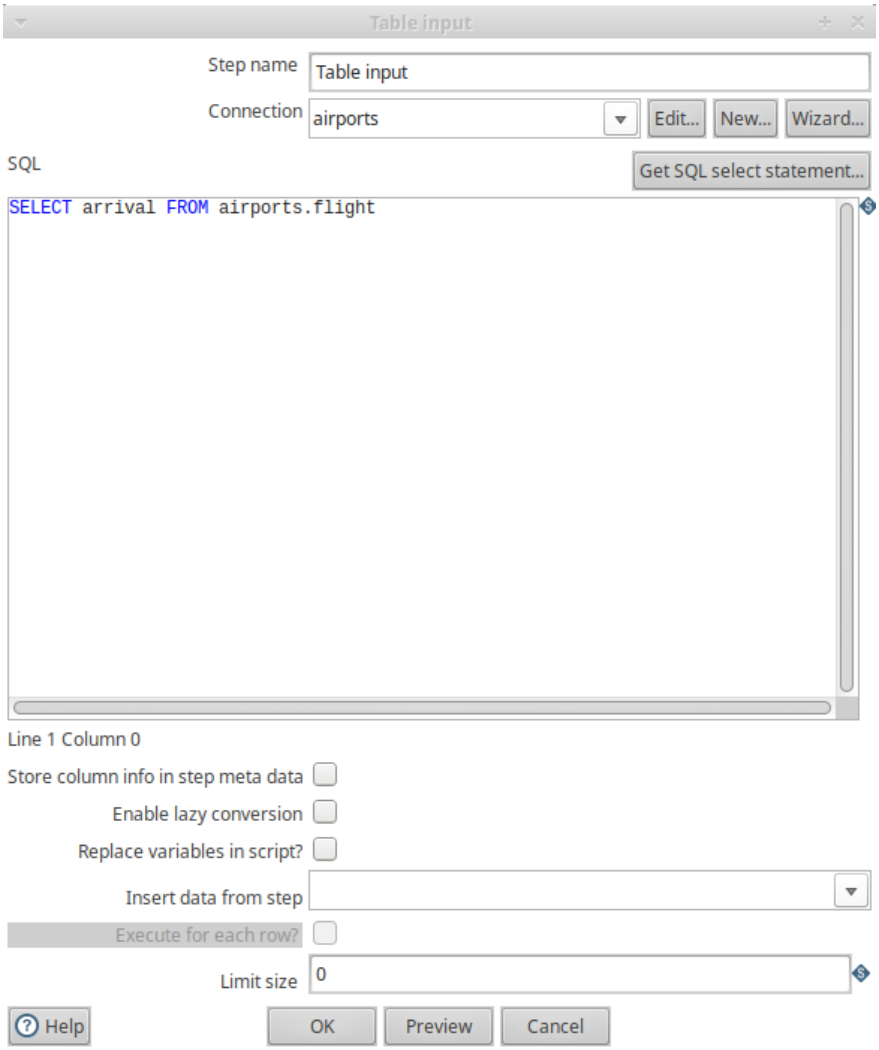

Examine preview data
Rows of step: Calculator (1210 rows)
departure | Year | Month | Day
1 2015/06/01 01:26:00.000000000 | 2015 | 6 | 1
2 2015/06/01 03:04:00.000000000 | 2015 | 6 | 1
3 2015/06/01 03:04:00.000000000 | 2015 | 6 | 1
4 2015/06/01 07:26:00.000000000 | 2015 | 6 | 1
5 2015/06/01 07:35:00.000000000 | 2015 | 6 | 1
6 2015/06/01 08:02:00.000000000 | 2015 | 6 | 1
7 2015/06/01 08:44:00.000000000 | 2015 | 6 | 1
8 2015/06/01 13:35:00.000000000 | 2015 | 6 | 1
9 2015/06/01 14:23:00.000000000 | 2015 | 6 | 1
10 2015/06/01 15:28:00.000000000 | 2015 | 6 | 1
11 2015/06/01 17:31:00.000000000 | 2015 | 6 | 1
12 2015/06/01 18:45:00.000000000 | 2015 | 6 | 1
13 2015/06/01 19:51:00.000000000 | 2015 | 6 | 1
Close

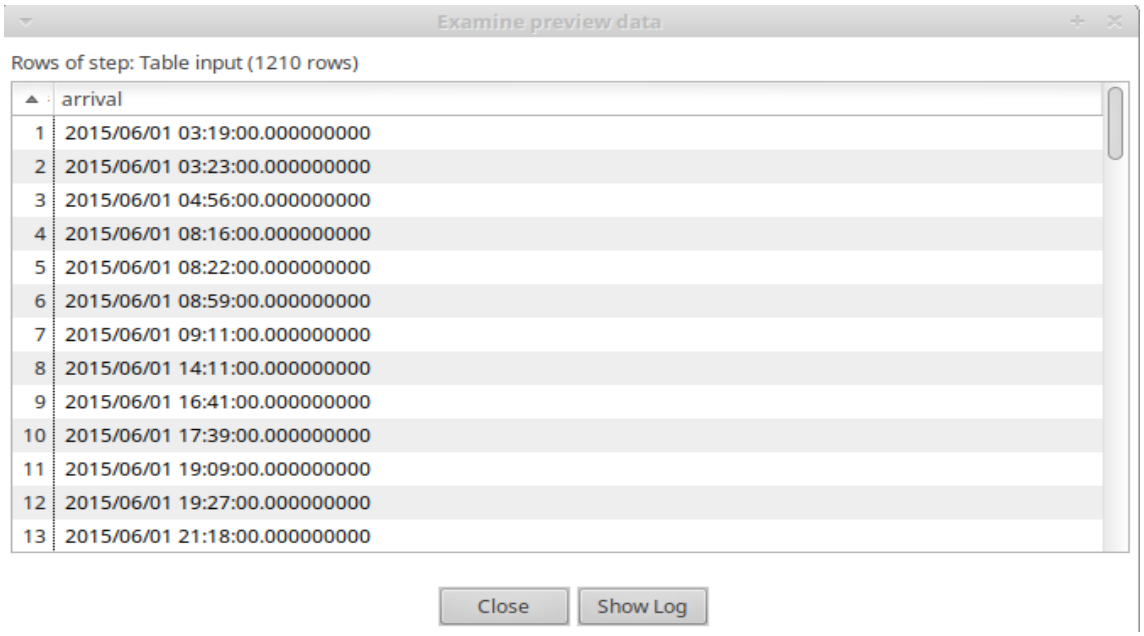

Insert / update
Step name
Insert / update
Connection
airports_dw
Edit...
New...
Wizard...
Target schema
airports_dw
Browse...
Target table
dim_from_time
Browse...
Commit size
100
Don't perform any updates:
The key(s) to look up the value(s):
Table field
Comparator
Stream field1
Stream field2
Get fields
1
from_time_id
=
departure
Update fields:
Table field
Stream field
Update
Get update fields
Edit mapping
1
from_time_id
departure
Y
2
year
Year
Y
3
month
Month
Y
4
day
Day
Y
Help
OK
Cancel
SQL

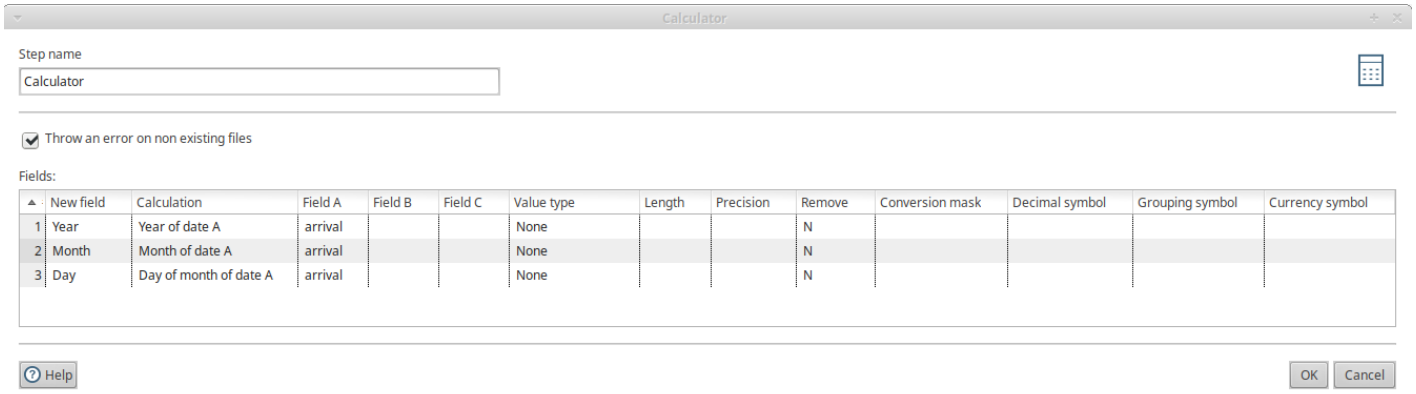

Examine preview data
Rows of step: Insert / update (1210 rows)
departure
Year
Month
Day
1 2015/06/01 01:26:00.000000000 2015 6 1
2 2015/06/01 03:04:00.000000000 2015 6 1
3 2015/06/01 03:04:00.000000000 2015 6 1
4 2015/06/01 07:26:00.000000000 2015 6 1
5 2015/06/01 07:35:00.000000000 2015 6 1
6 2015/06/01 08:02:00.000000000 2015 6 1
7 2015/06/01 08:44:00.000000000 2015 6 1
8 2015/06/01 13:35:00.000000000 2015 6 1
9 2015/06/01 14:23:00.000000000 2015 6 1
10 2015/06/01 15:28:00.000000000 2015 6 1
11 2015/06/01 17:31:00.000000000 2015 6 1
12 2015/06/01 18:45:00.000000000 2015 6 1
13 2015/06/01 19:51:00.000000000 2015 6 1
Close

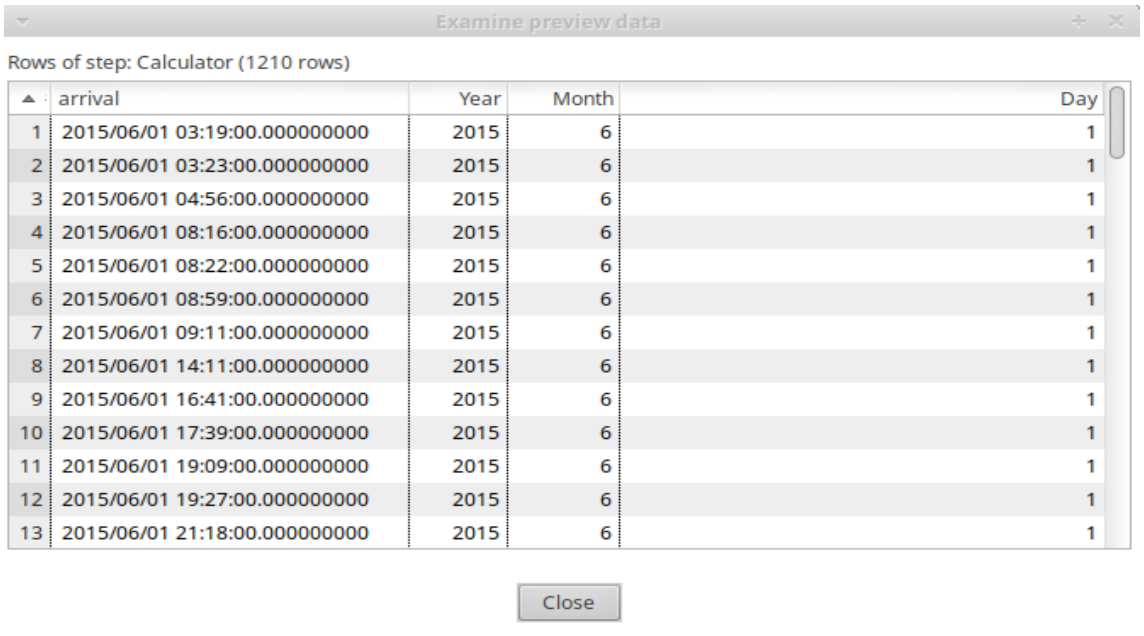
4- Time dimension for date/time of arrival

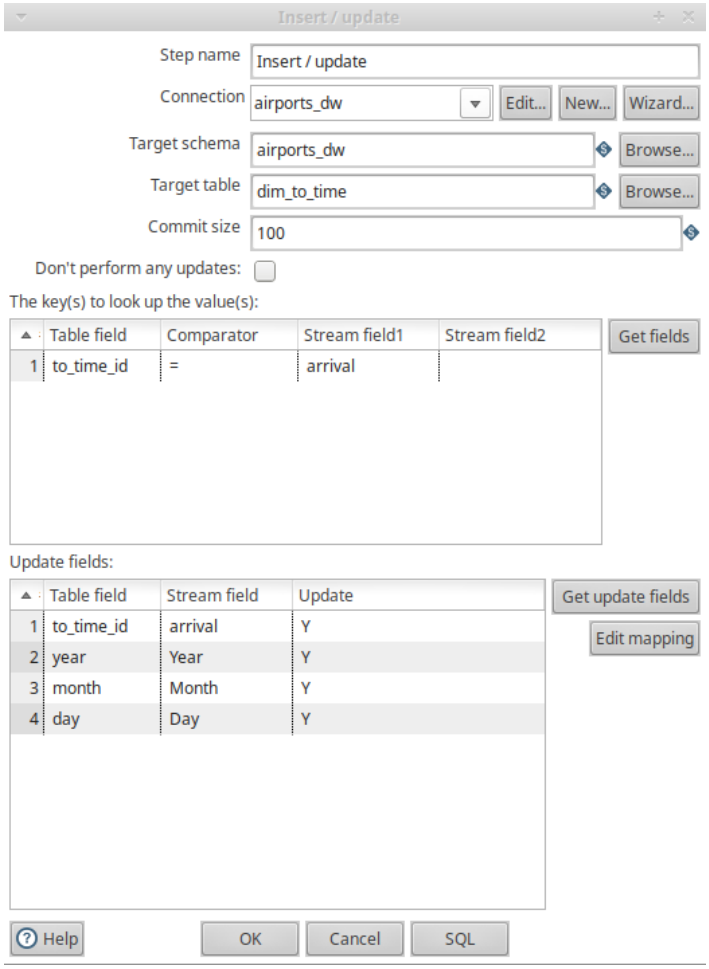


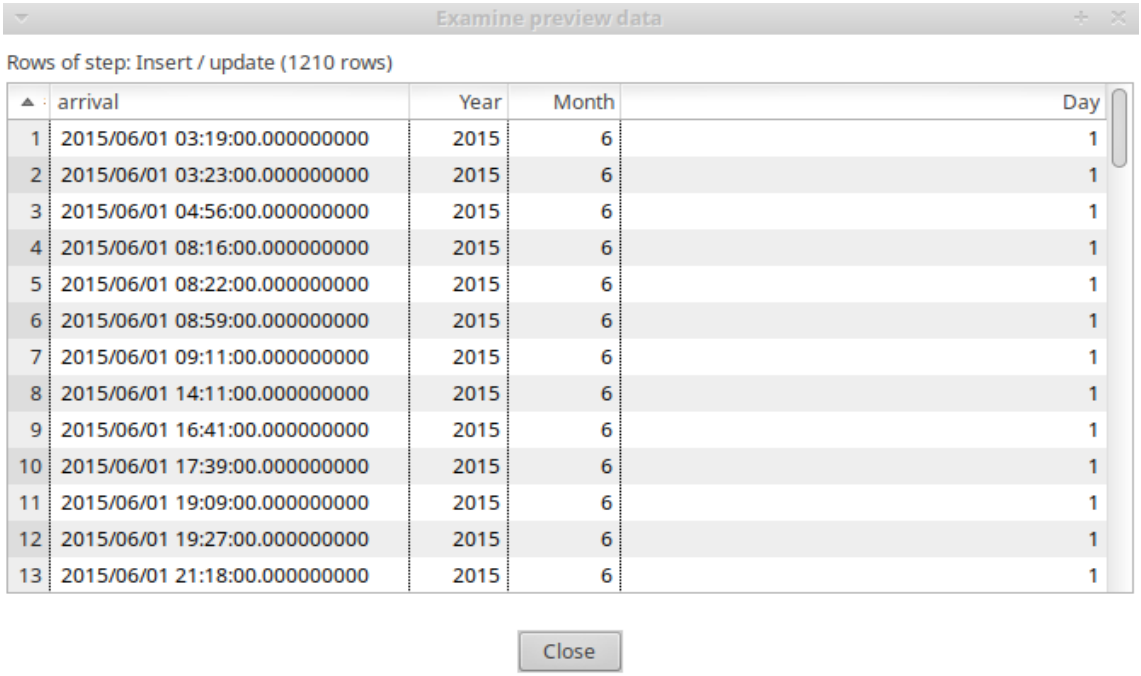



Examine preview data
Rows of step: Table input (1210 rows)
arrival
1 2015/06/01 03:19:00.000000000
2 2015/06/01 03:23:00.000000000
3 2015/06/01 04:56:00.000000000
4 2015/06/01 08:16:00.000000000
5 2015/06/01 08:22:00.000000000
6 2015/06/01 08:59:00.000000000
7 2015/06/01 09:11:00.000000000
8 2015/06/01 14:11:00.000000000
9 2015/06/01 16:41:00.000000000
10 2015/06/01 17:39:00.000000000
11 2015/06/01 19:09:00.000000000
12 2015/06/01 19:27:00.000000000
13 2015/06/01 21:18:00.000000000
Close
Show Log


Calculator
Step name
Calculator
Throw an error on non existing files
Fields:
New field | Calculation | Field A | Field B | Field C | Value type | Length | Precision | Remove | Conversion mask | Decimal symbol | Grouping symbol | Currency symbol
1 Year | Year of date A | arrival | None | N
2 Month | Month of date A | arrival | None | N
3 Day | Day of month of date A | arrival | None | N
Help
OK
Cancel


Examine preview data
Rows of step: Calculator (1210 rows)
arrival | Year | Month | Day
1 2015/06/01 03:19:00.000000000 2015 6 1
2 2015/06/01 03:23:00.000000000 2015 6 1
3 2015/06/01 04:56:00.000000000 2015 6 1
4 2015/06/01 08:16:00.000000000 2015 6 1
5 2015/06/01 08:22:00.000000000 2015 6 1
6 2015/06/01 08:59:00.000000000 2015 6 1
7 2015/06/01 09:11:00.000000000 2015 6 1
8 2015/06/01 14:11:00.000000000 2015 6 1
9 2015/06/01 16:41:00.000000000 2015 6 1
10 2015/06/01 17:39:00.000000000 2015 6 1
11 2015/06/01 19:09:00.000000000 2015 6 1
12 2015/06/01 19:27:00.000000000 2015 6 1
13 2015/06/01 21:18:00.000000000 2015 6 1
Close


Insert / update
Step name
Insert / update
Connection
airports_dw
Edit...
New...
Wizard...
Target schema
airports_dw
Browse...
Target table
dim_to_time
Browse...
Commit size
100
Don't perform any updates:
The key(s) to look up the value(s):
Table field
Comparator
Stream field1
Stream field2
1
to_time_id
=
arrival
Get fields
Update fields:
Table field
Stream field
Update
1
to_time_id
arrival
Y
2
year
Year
Y
3
month
Month
Y
4
day
Day
Y
Get update fields
Edit mapping
Help
OK
Cancel
SQL


Examine preview data
Rows of step: Insert / update (1210 rows)
arrival
Year
Month
Day
1 2015/06/01 03:19:00.000000000 2015 6 1
2 2015/06/01 03:23:00.000000000 2015 6 1
3 2015/06/01 04:56:00.000000000 2015 6 1
4 2015/06/01 08:16:00.000000000 2015 6 1
5 2015/06/01 08:22:00.000000000 2015 6 1
6 2015/06/01 08:59:00.000000000 2015 6 1
7 2015/06/01 09:11:00.000000000 2015 6 1
8 2015/06/01 14:11:00.000000000 2015 6 1
9 2015/06/01 16:41:00.000000000 2015 6 1
10 2015/06/01 17:39:00.000000000 2015 6 1
11 2015/06/01 19:09:00.000000000 2015 6 1
12 2015/06/01 19:27:00.000000000 2015 6 1
13 2015/06/01 21:18:00.000000000 2015 6 1
Close

5- Airline dimension

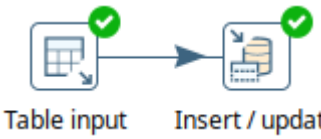


Table input

Step name: Table input

Connection: airports [Edit...] [New...] [Wizard...]

SQL [Get SQL select statement...]

```
SELECT
  airline_id
, airlinename
FROM airports.airline
```

Line 1 Column 0

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Limit size: 0

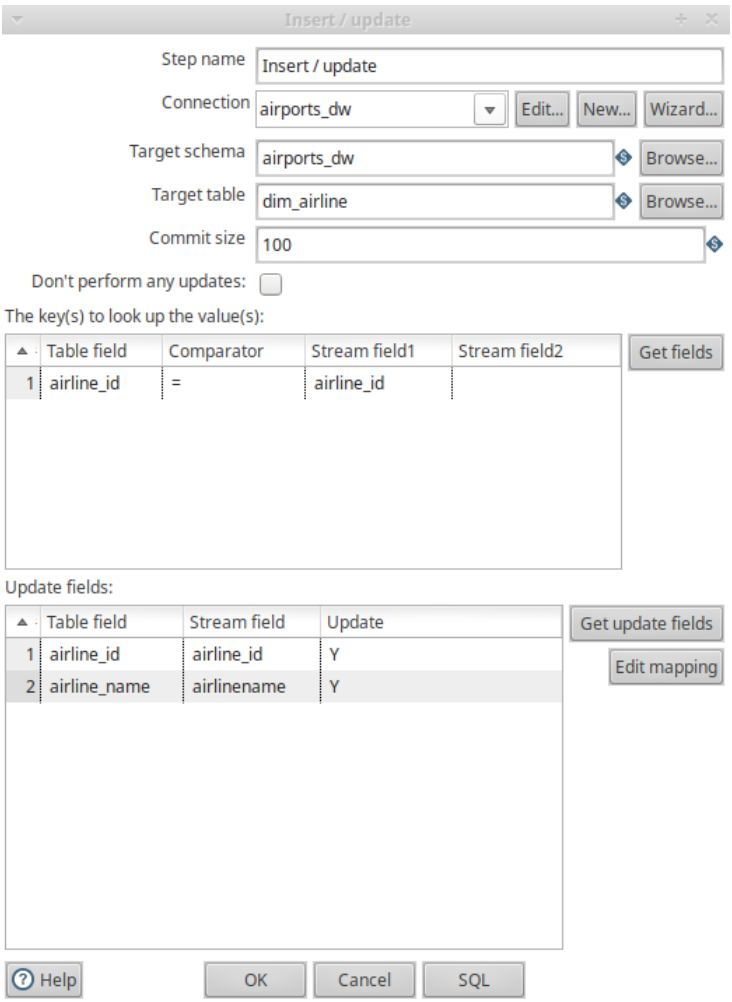
[Help] [OK] [Preview] [Cancel]

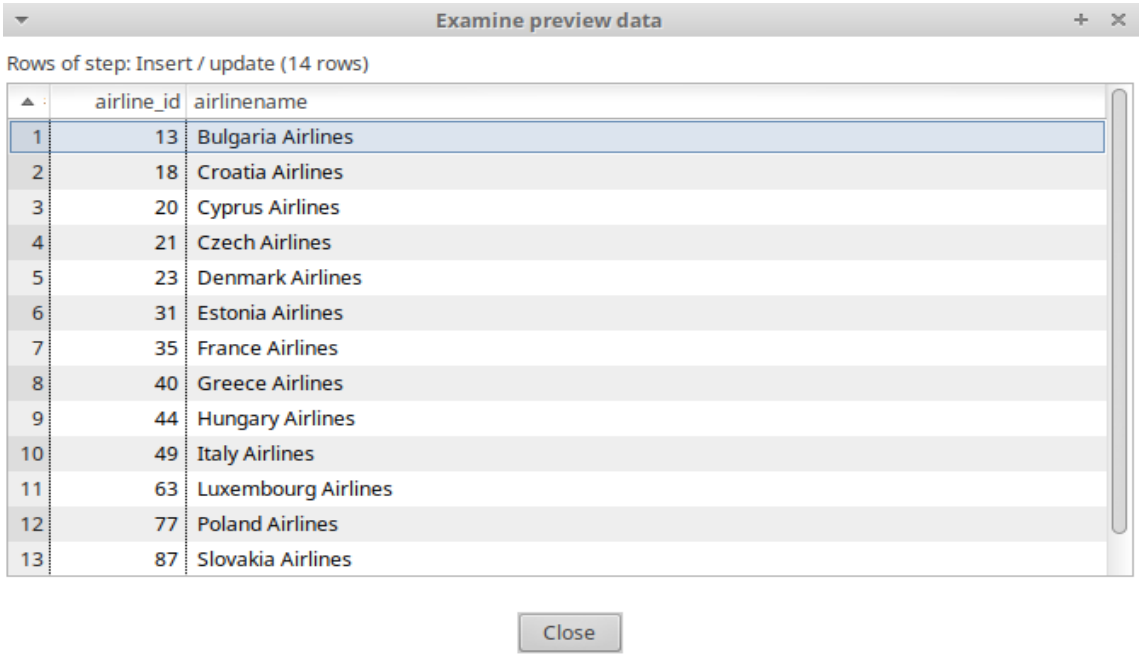
Examine preview data

Rows of step: Table input (14 rows)

	airline_id	airlinename
1	13	Bulgaria Airlines
2	18	Croatia Airlines
3	20	Cyprus Airlines
4	21	Czech Airlines
5	23	Denmark Airlines
6	31	Estonia Airlines
7	35	France Airlines
8	40	Greece Airlines
9	44	Hungary Airlines
10	49	Italy Airlines
11	63	Luxembourg Airlines
12	77	Poland Airlines
13	87	Slovakia Airlines

[Close] [Show Log]


Insert / update
Step name
Insert / update
Connection
airports_dw
Edit...
New...
Wizard...
Target schema
airports_dw
Browse...
Target table
dim_airline
Browse...
Commit size
100
Don't perform any updates:
The key(s) to look up the value(s):
Table field
Comparator
Stream field1
Stream field2
Get fields
1
airline_id
=
airline_id
Update fields:
Table field
Stream field
Update
Get update fields
Edit mapping
1
airline_id
airline_id
Y
2
airline_name
airlinename
Y
Help
OK
Cancel
SQL


Examine preview data
Rows of step: Insert / update (14 rows)
airline_id
airlinename
1 13 Bulgaria Airlines
2 18 Croatia Airlines
3 20 Cyprus Airlines
4 21 Czech Airlines
5 23 Denmark Airlines
6 31 Estonia Airlines
7 35 France Airlines
8 40 Greece Airlines
9 44 Hungary Airlines
10 49 Italy Airlines
11 63 Luxembourg Airlines
12 77 Poland Airlines
13 87 Slovakia Airlines
Close

6- Airplane dimension

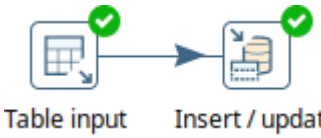


Table input

Step name: Table input

Connection: airports [Edit...] [New...] [Wizard...]

SQL [Get SQL select statement...]

```
SELECT
  airplane_id
, type_id
FROM airports.airplane
```

Line 1 Column 0

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Limit size: 0

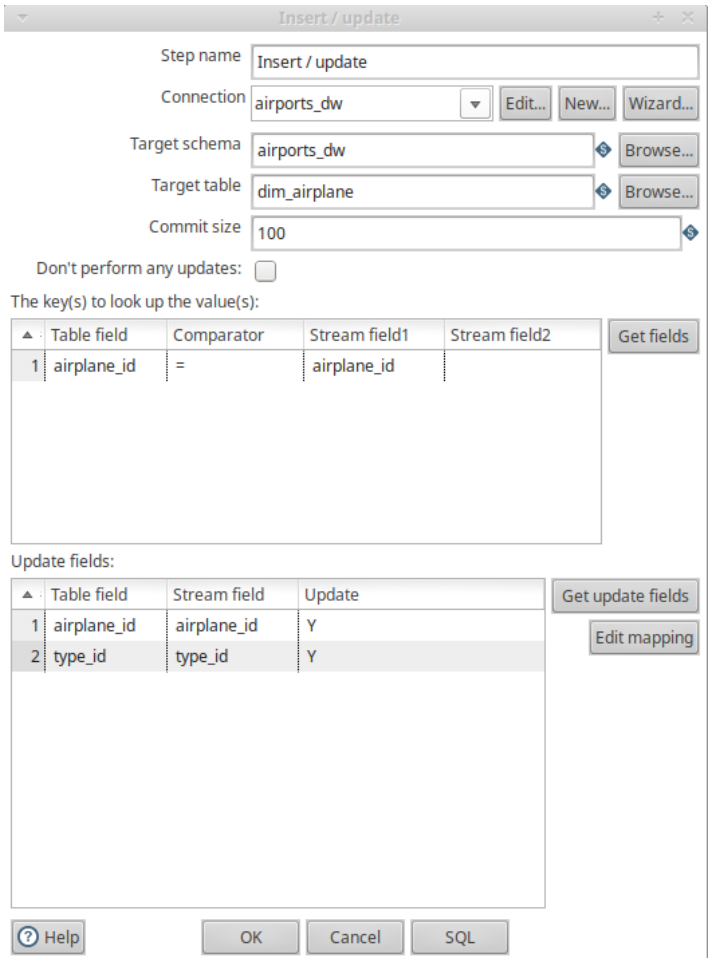
[Help] [OK] [Preview] [Cancel]

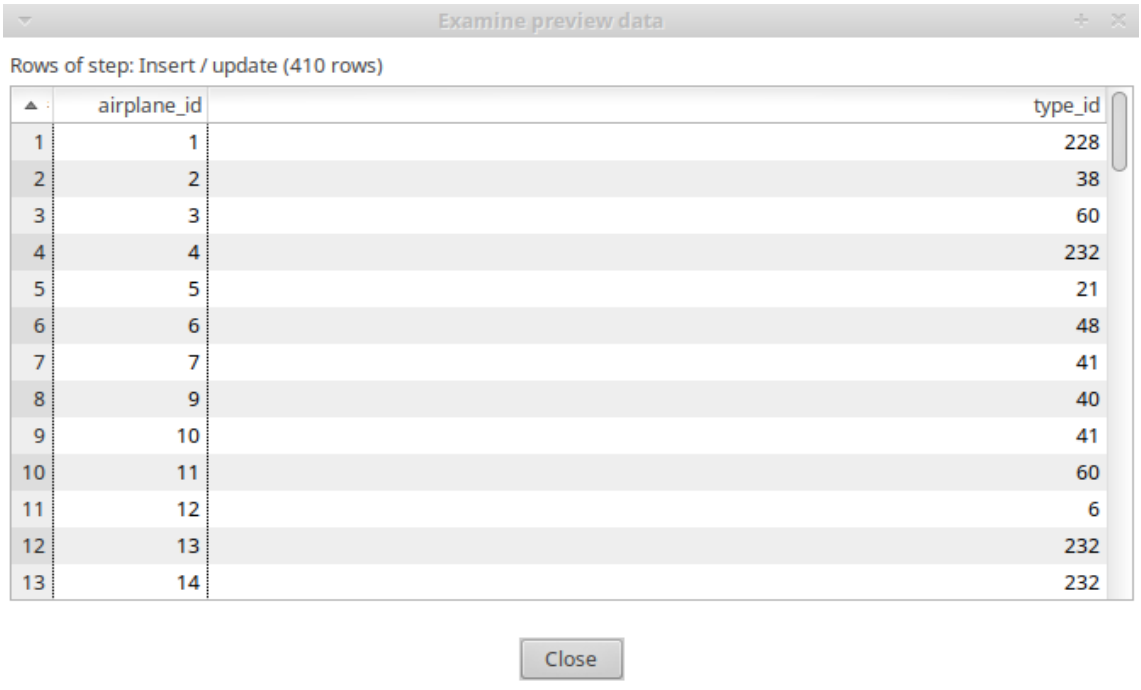
Examine preview data

Rows of step: Table input (410 rows)

	airplane_id	type_id
1	1	228
2	2	38
3	3	60
4	4	232
5	5	21
6	6	48
7	7	41
8	9	40
9	10	41
10	11	60
11	12	6
12	13	232
13	14	232

[Close] [Show Log]


Insert / update
Step name
Insert / update
Connection
airports_dw
Edit...
New...
Wizard...
Target schema
airports_dw
Browse...
Target table
dim_airplane
Browse...
Commit size
100
Don't perform any updates:
The key(s) to look up the value(s):
Table field
Comparator
Stream field1
Stream field2
Get fields
1
airplane_id
=
airplane_id
Update fields:
Table field
Stream field
Update
Get update fields
Edit mapping
1
airplane_id
airplane_id
Y
2
type_id
type_id
Y
Help
OK
Cancel
SQL


Examine preview data
Rows of step: Insert / update (410 rows)
airplane_id
type_id
1 1 228
2 2 38
3 3 60
4 4 232
5 5 21
6 6 48
7 7 41
8 9 40
9 10 41
10 11 60
11 12 6
12 13 232
13 14 232
Close

7- Flights fact table

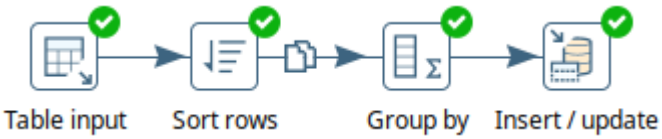


Table input

Step name: Table input

Connection: airports [Edit...] [New...] [Wizard...]

SQL [Get SQL select statement...]

```
SELECT
  flight_id
, `from`
, `to`
, departure
, arrival
, airline_id
, airplane_id
, seat
, price
FROM airports.flight NATURAL JOIN airports.booking
```

Line 1 Column 0

Store column info in step meta data ☐

Enable lazy conversion ☐

Replace variables in script? ☐

Insert data from step

Execute for each row? ☐

Limit size: 0

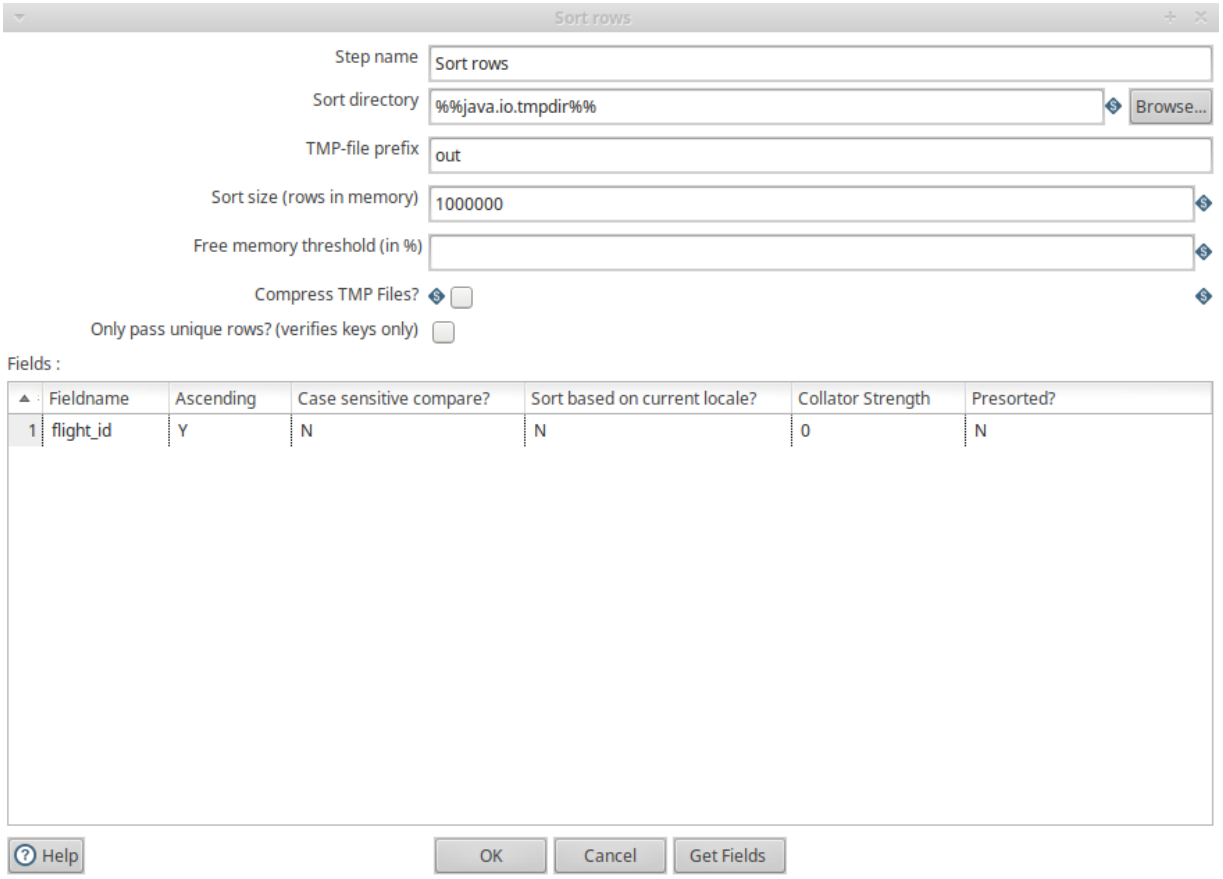
[Help] [OK] [Preview] [Cancel]

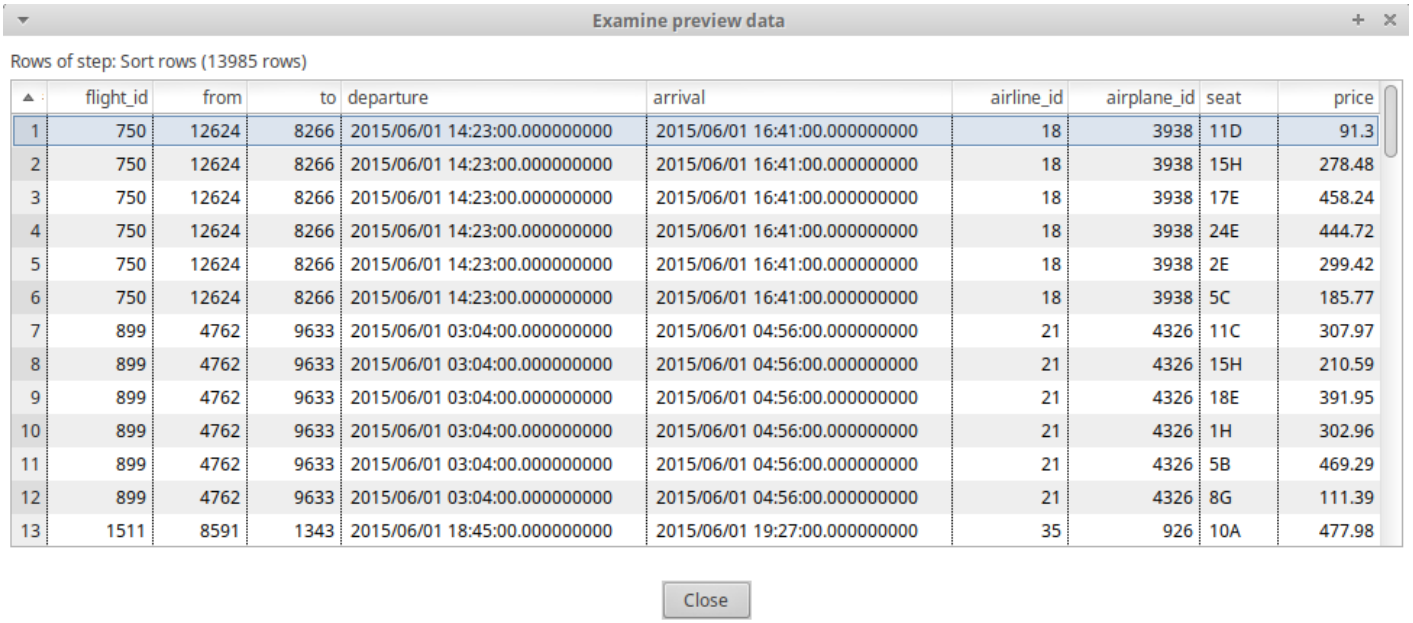
Examine preview data

Rows of step: Table input (13985 rows)

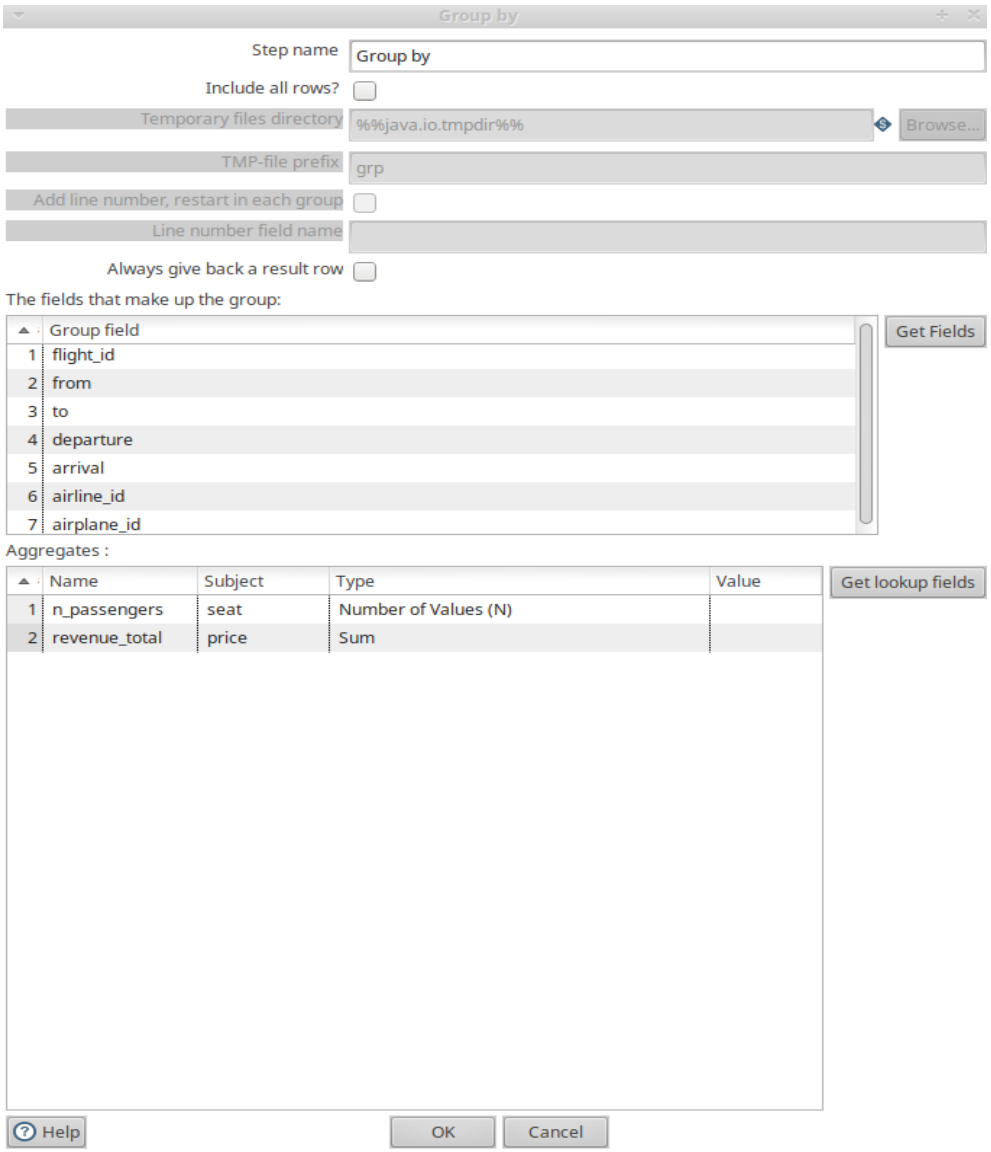
	flight_id	from	to	departure	arrival	airline_id	airplane_id	seat	price
1	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	11D	91.3
2	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	15H	278.48
3	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	17E	458.24
4	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	24E	444.72
5	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	2E	299.42
6	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	5C	185.77
7	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	11C	307.97
8	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	15H	210.59
9	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	18E	391.95
10	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	1H	302.96
11	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	5B	469.29
12	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	8G	111.39
13	1511	8591	1343	2015/06/01 18:45:00.000000000	2015/06/01 19:27:00.000000000	35	926	10A	477.98

[Close]





	flight_id	from	to	departure	arrival	airline_id	airplane_id	seat	price
1	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	11D	91.3
2	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	15H	278.48
3	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	17E	458.24
4	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	24E	444.72
5	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	2E	299.42
6	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	5C	185.77
7	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	11C	307.97
8	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	15H	210.59
9	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	18E	391.95
10	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	1H	302.96
11	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	5B	469.29
12	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	8G	111.39
13	1511	8591	1343	2015/06/01 18:45:00.000000000	2015/06/01 19:27:00.000000000	35	926	10A	477.98

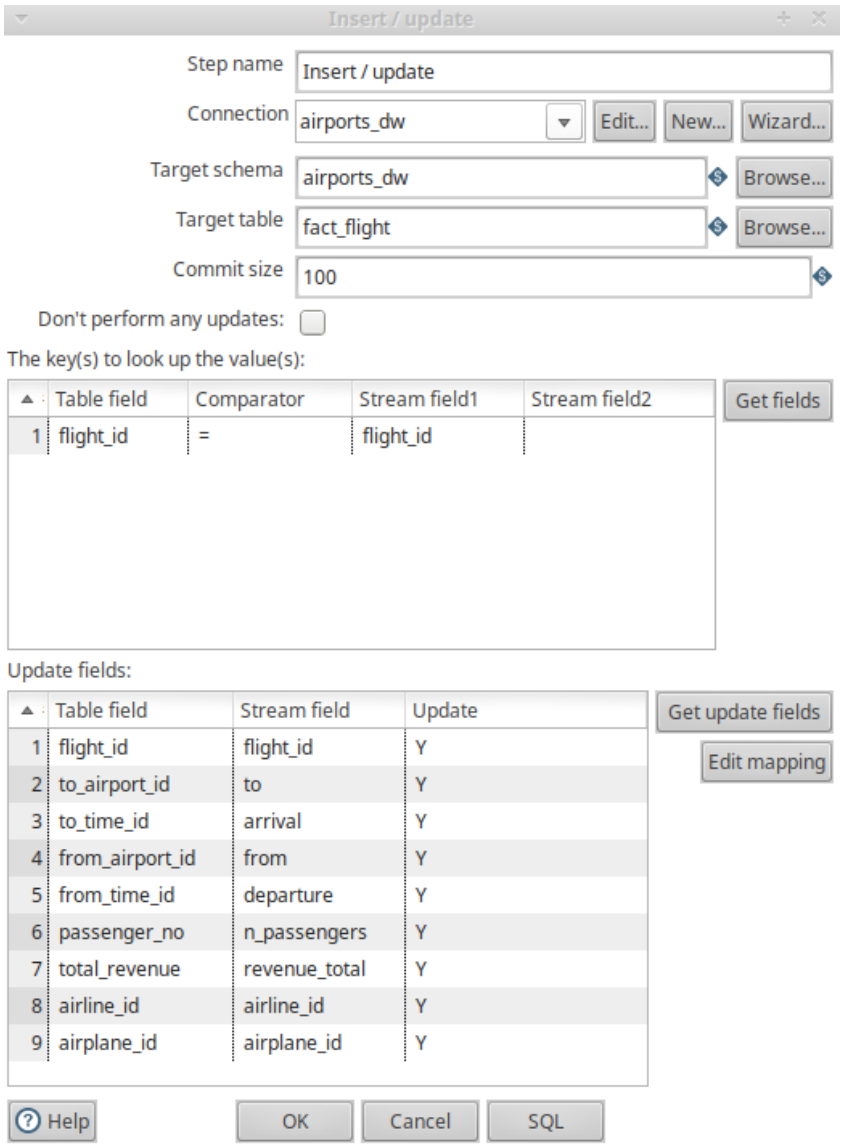


Examine preview data

Rows of step: Group by (1210 rows)

	flight_id	from	to	departure	arrival	airline_id	airplane_id	n_passengers	revenue_total
1	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	6	1757.93
2	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	6	1794.15
3	1511	8591	1343	2015/06/01 18:45:00.000000000	2015/06/01 19:27:00.000000000	35	926	18	4166.23
4	1515	3420	748	2015/06/01 08:02:00.000000000	2015/06/01 08:22:00.000000000	35	902	4	651.13
5	1543	3887	10805	2015/06/01 20:23:00.000000000	2015/06/01 22:20:00.000000000	35	902	3	483.37
6	1871	1085	6973	2015/06/01 21:16:00.000000000	2015/06/01 22:26:00.000000000	44	2557	5	1544.67
7	1880	11812	6429	2015/06/01 01:26:00.000000000	2015/06/01 03:19:00.000000000	44	2564	17	3945.94
8	2117	1572	8829	2015/06/01 08:44:00.000000000	2015/06/01 09:11:00.000000000	49	1207	4	1347.69
9	2673	10564	6002	2015/06/01 22:50:00.000000000	2015/06/02 01:24:00.000000000	63	4869	21	4986.82
10	2703	4725	9838	2015/06/01 19:51:00.000000000	2015/06/01 21:18:00.000000000	63	4827	17	4490.16
11	2715	10904	2867	2015/06/01 17:31:00.000000000	2015/06/01 19:09:00.000000000	63	4840	6	1678.74
12	2717	8210	5024	2015/06/01 13:35:00.000000000	2015/06/01 14:11:00.000000000	63	4869	20	4245.1
13	3359	12159	3796	2015/06/01 15:28:00.000000000	2015/06/01 17:39:00.000000000	77	2255	18	4450.93
14	3367	10903	3074	2015/06/01 03:04:00.000000000	2015/06/01 03:23:00.000000000	77	2254	5	1469.83

Close



Examine preview data

Rows of step: Insert / update (1210 rows)

	flight_id	from	to	departure	arrival	airline_id	airplane_id	n_passengers	revenue_total
1	750	12624	8266	2015/06/01 14:23:00.000000000	2015/06/01 16:41:00.000000000	18	3938	6	1757.93
2	899	4762	9633	2015/06/01 03:04:00.000000000	2015/06/01 04:56:00.000000000	21	4326	6	1794.15
3	1511	8591	1343	2015/06/01 18:45:00.000000000	2015/06/01 19:27:00.000000000	35	926	18	4166.23
4	1515	3420	748	2015/06/01 08:02:00.000000000	2015/06/01 08:22:00.000000000	35	902	4	651.13
5	1543	3887	10805	2015/06/01 20:23:00.000000000	2015/06/01 22:20:00.000000000	35	902	3	483.37
6	1871	1085	6973	2015/06/01 21:16:00.000000000	2015/06/01 22:26:00.000000000	44	2557	5	1544.67
7	1880	11812	6429	2015/06/01 01:26:00.000000000	2015/06/01 03:19:00.000000000	44	2564	17	3945.94
8	2117	1572	8829	2015/06/01 08:44:00.000000000	2015/06/01 09:11:00.000000000	49	1207	4	1347.69
9	2673	10564	6002	2015/06/01 22:50:00.000000000	2015/06/02 01:24:00.000000000	63	4869	21	4986.82
10	2703	4725	9838	2015/06/01 19:51:00.000000000	2015/06/01 21:18:00.000000000	63	4827	17	4490.16
11	2715	10904	2867	2015/06/01 17:31:00.000000000	2015/06/01 19:09:00.000000000	63	4840	6	1678.74
12	2717	8210	5024	2015/06/01 13:35:00.000000000	2015/06/01 14:11:00.000000000	63	4869	20	4245.1
13	3359	12159	3796	2015/06/01 15:28:00.000000000	2015/06/01 17:39:00.000000000	77	2255	18	4450.93
14	3367	10903	3074	2015/06/01 03:04:00.000000000	2015/06/01 03:23:00.000000000	77	2254	5	1469.83

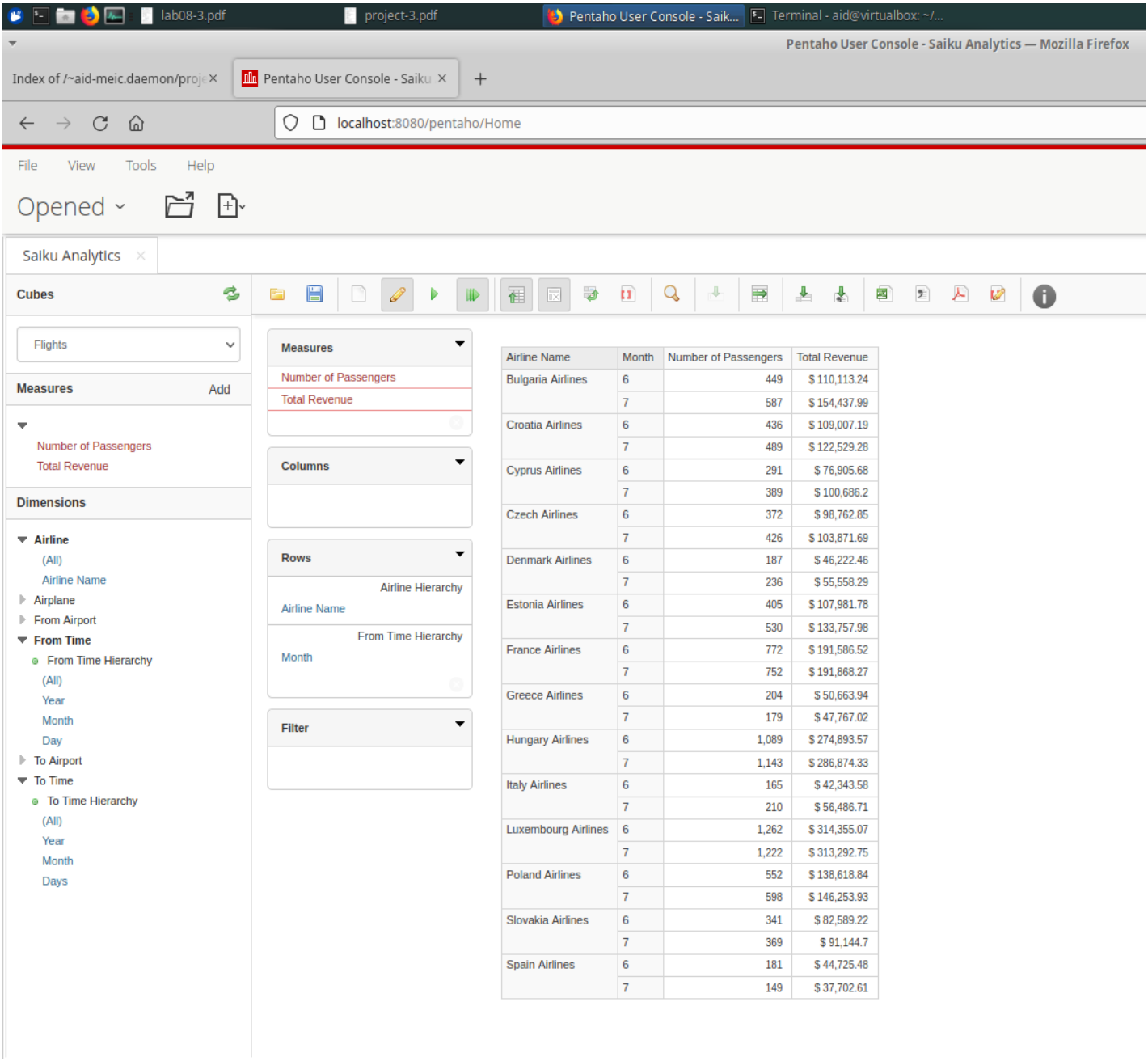
Close

C) XML code for the cube definition:

```
<Schema name="airports_dw">
  <Cube name="Flights" visible="true" cache="true" enabled="true">
    <Table name="fact_flight">
    </Table>
    <Dimension type="StandardDimension" visible="true" foreignKey="airline_id" highCardinality="false" name="Airline">
      <Hierarchy name="Airline Hierarchy" visible="true" hasAll="true" allMemberName="All Airlines" primaryKey="airline_id">
        <Table name="dim_airline">
        </Table>
        <Level name="Airline Name" visible="true" column="airline_name" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true" foreignKey="airplane_id" highCardinality="false" name="Airplane">
      <Hierarchy name="Airplane Hierarchy" visible="true" hasAll="true" allMemberName="All Airplanes" primaryKey="airplane_id">
        <Table name="dim_airplane">
        </Table>
        <Level name="Type" visible="true" column="type_id" type="Integer" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true" foreignKey="from_airport_id" highCardinality="false" name="From Airport">
      <Hierarchy name="From Airport Hierarchy" visible="true" hasAll="true" allMemberName="All From Airports" primaryKey="from_airport_id">
        <Table name="dim_from_airport">
        </Table>
        <Level name="From Airport Country" visible="true" column="from_airport_country" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="From Airport City" visible="true" column="from_airport_city" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="From Airport Name" visible="true" column="from_airport_name" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="StandardDimension" visible="true" foreignKey="to_airport_id" highCardinality="false" name="To Airport">
      <Hierarchy name="To Airport Hierarchy" visible="true" hasAll="true" primaryKey="to_airport_id">
        <Table name="dim_to_airport">
        </Table>
        <Level name="To Airport Country" visible="true" column="to_airport_country" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="To Airport City" visible="true" column="to_airport_city" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="To Airport Name" visible="true" column="to_airport_name" type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="TimeDimension" visible="true" foreignKey="from_time_id" highCardinality="false" name="From Time">
      <Hierarchy name="From Time Hierarchy" visible="true" hasAll="true" allMemberName="All From Time" primaryKey="from_time_id">
        <Table name="dim_from_time">
        </Table>
        <Level name="Year" visible="true" column="year" type="Integer" uniqueMembers="false" levelType="TimeYears" hideMemberIf="Never">
        </Level>
        <Level name="Month" visible="true" column="month" type="Integer" uniqueMembers="false" levelType="TimeMonths" hideMemberIf="Never">
        </Level>
        <Level name="Day" visible="true" column="day" type="Integer" uniqueMembers="false" levelType="TimeDays" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Dimension type="TimeDimension" visible="true" foreignKey="to_time_id" highCardinality="false" name="To Time">
      <Hierarchy name="To Time Hierarchy" visible="true" hasAll="true" allMemberName="All To Times" primaryKey="to_time_id">
        <Table name="dim_to_time">
        </Table>
        <Level name="Year" visible="true" column="year" type="Integer" uniqueMembers="false" levelType="TimeYears" hideMemberIf="Never">
        </Level>
        <Level name="Month" visible="true" column="month" type="Integer" uniqueMembers="false" levelType="TimeMonths" hideMemberIf="Never">
        </Level>
        <Level name="Days" visible="true" column="day" type="Integer" uniqueMembers="false" levelType="TimeDays" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Measure name="Number of Passengers" column="passenger_no" datatype="Integer" aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total Revenue" column="total_revenue" datatype="Numeric" formatString="$ #,###.##" aggregator="sum" visible="true">
    </Measure>
  </Cube>
</Schema>
```

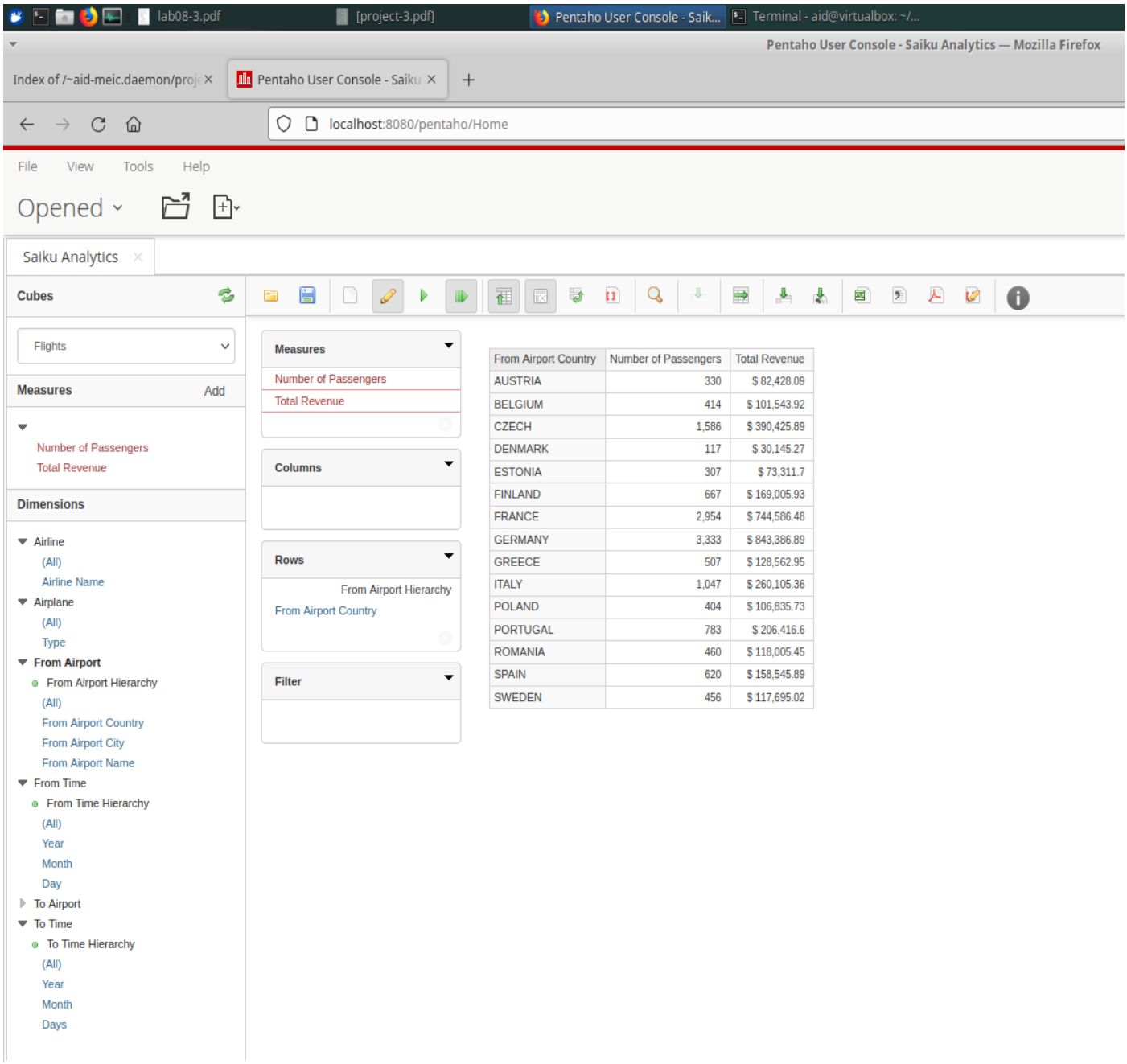
D) Analysis Queries Results:

1- <u>Passengers and revenue by airline and month</u>



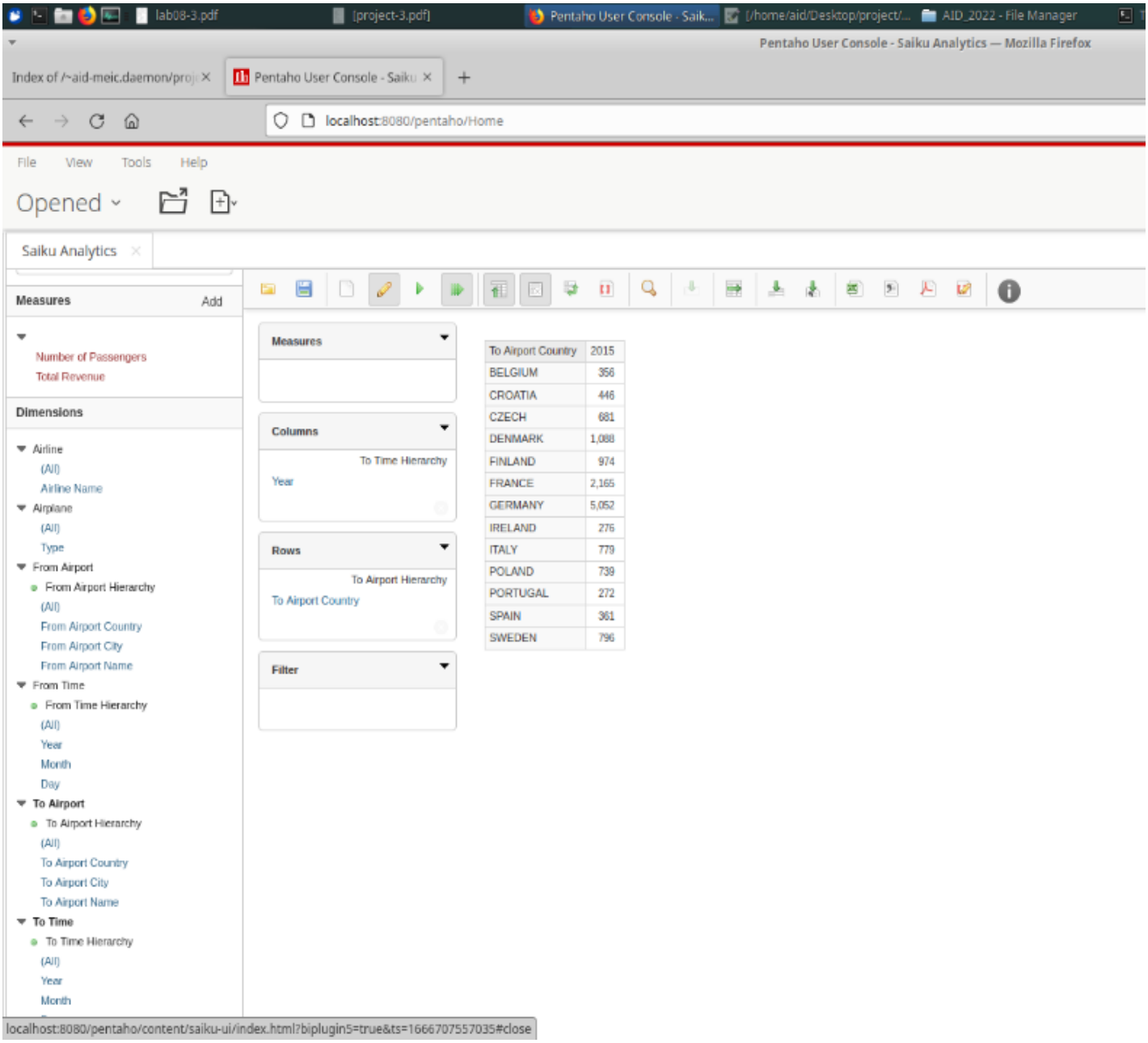
Airline Name	Month	Number of Passengers	Total Revenue
Bulgaria Airlines	6	449	$ 110,113.24
	7	587	$ 154,437.99
Croatia Airlines	6	436	$ 109,007.19
	7	489	$ 122,529.28
Cyprus Airlines	6	291	$ 76,905.68
	7	389	$ 100,686.2
Czech Airlines	6	372	$ 98,762.85
	7	426	$ 103,871.69
Denmark Airlines	6	187	$ 46,222.46
	7	236	$ 55,558.29
Estonia Airlines	6	405	$ 107,981.78
	7	530	$ 133,757.98
France Airlines	6	772	$ 191,586.52
	7	752	$ 191,868.27
Greece Airlines	6	204	$ 50,663.94
	7	179	$ 47,767.02
Hungary Airlines	6	1,089	$ 274,893.57
	7	1,143	$ 286,874.33
Italy Airlines	6	165	$ 42,343.58
	7	210	$ 56,486.71
Luxembourg Airlines	6	1,262	$ 314,355.07
	7	1,222	$ 313,292.75
Poland Airlines	6	552	$ 138,618.84
	7	598	$ 146,253.93
Slovakia Airlines	6	341	$ 82,589.22
	7	369	$ 91,144.7
Spain Airlines	6	181	$ 44,725.48
	7	149	$ 37,702.61

2- Passengers and revenue by country of departure



From Airport Country	Number of Passengers	Total Revenue
AUSTRIA	330	$ 82,428.09
BELGIUM	414	$ 101,543.92
CZECH	1,586	$ 390,425.89
DENMARK	117	$ 30,145.27
ESTONIA	307	$ 73,311.7
FINLAND	667	$ 169,005.93
FRANCE	2,954	$ 744,586.48
GERMANY	3,333	$ 843,386.89
GREECE	507	$ 128,562.95
ITALY	1,047	$ 260,105.36
POLAND	404	$ 106,835.73
PORTUGAL	783	$ 206,416.6
ROMANIA	460	$ 118,005.45
SPAIN	620	$ 158,545.89
SWEDEN	456	$ 117,695.02

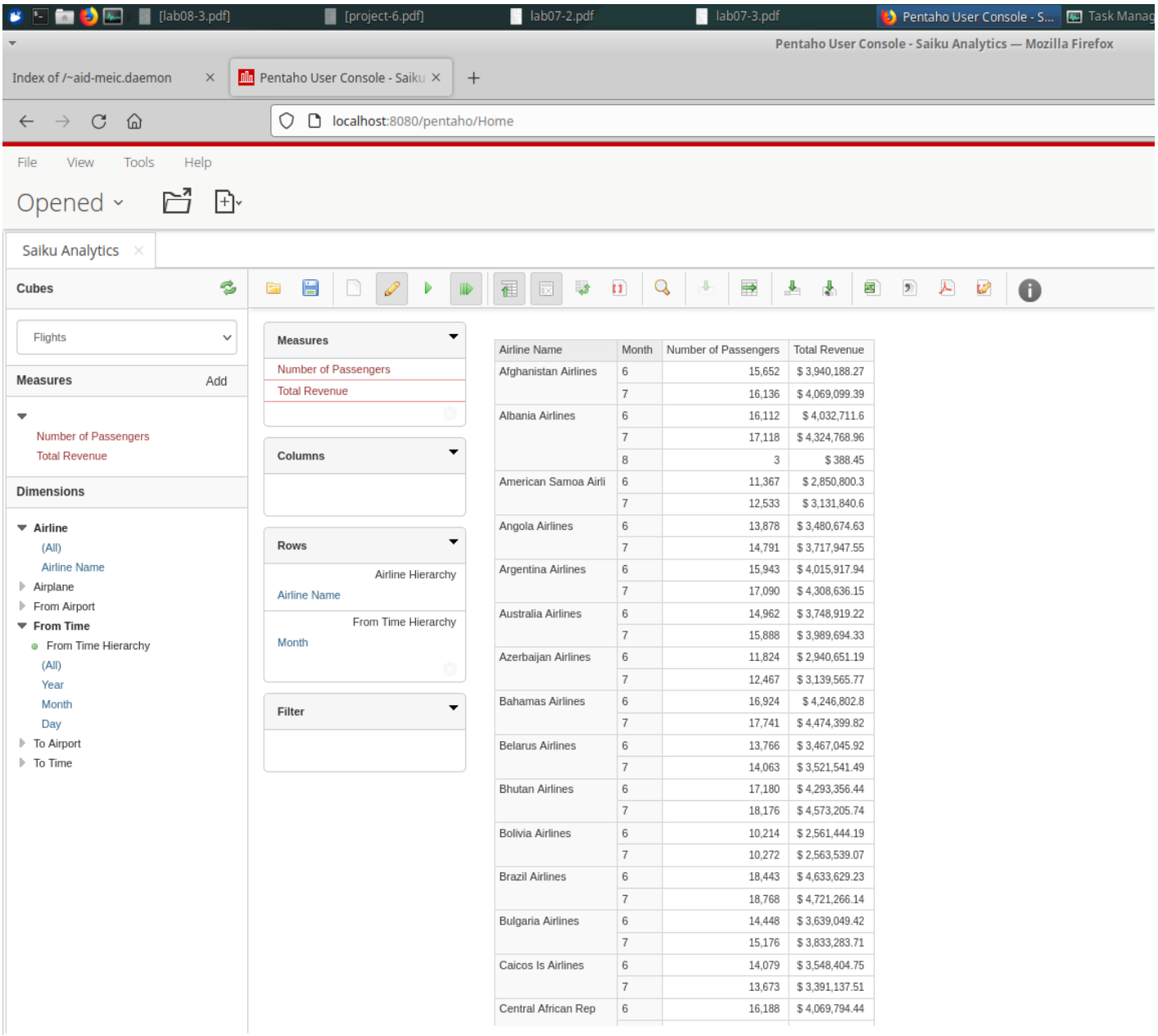
3- Number of flights by country of arrival per year



E) Large and extra-large analysis queries results:

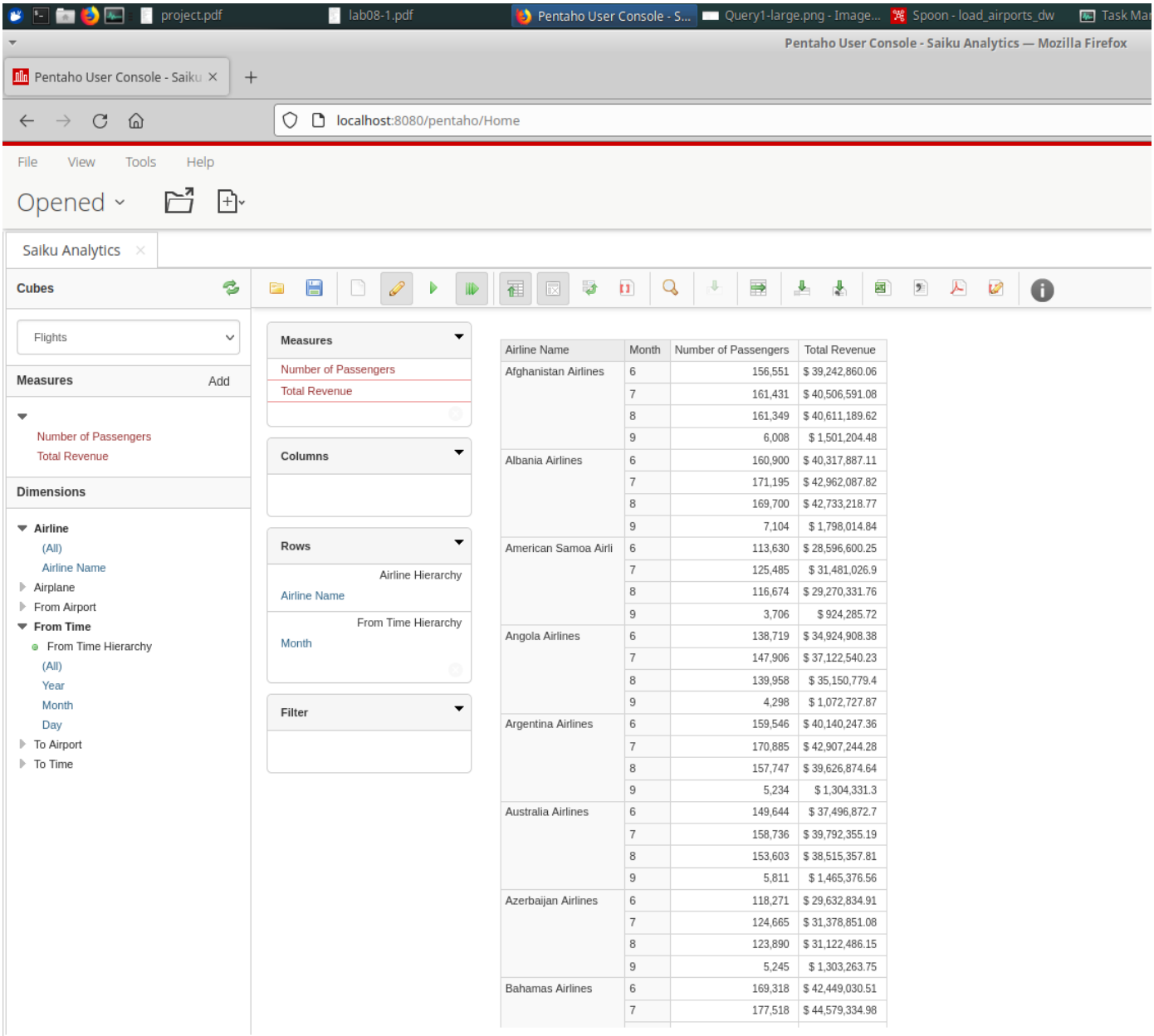
1- <u>Passengers and revenue by airline and month</u>

- Large dataset



Airline Name	Month	Number of Passengers	Total Revenue
Afghanistan Airlines	6	15,652	$ 3,940,188.27
	7	16,136	$ 4,069,099.39
Albania Airlines	6	16,112	$ 4,032,711.6
	7	17,118	$ 4,324,768.96
	8	3	$ 388.45
American Samoa Airli	6	11,367	$ 2,850,800.3
	7	12,533	$ 3,131,840.6
Angola Airlines	6	13,878	$ 3,480,674.63
	7	14,791	$ 3,717,947.55
Argentina Airlines	6	15,943	$ 4,015,917.94
	7	17,090	$ 4,308,636.15
Australia Airlines	6	14,962	$ 3,748,919.22
	7	15,888	$ 3,989,694.33
Azerbaijan Airlines	6	11,824	$ 2,940,651.19
	7	12,467	$ 3,139,565.77
Bahamas Airlines	6	16,924	$ 4,246,802.8
	7	17,741	$ 4,474,399.82
Belarus Airlines	6	13,766	$ 3,467,045.92
	7	14,063	$ 3,521,541.49
Bhutan Airlines	6	17,180	$ 4,293,356.44
	7	18,176	$ 4,573,205.74
Bolivia Airlines	6	10,214	$ 2,561,444.19
	7	10,272	$ 2,563,539.07
Brazil Airlines	6	18,443	$ 4,633,629.23
	7	18,768	$ 4,721,266.14
Bulgaria Airlines	6	14,448	$ 3,639,049.42
	7	15,176	$ 3,833,283.71
Caicos Is Airlines	6	14,079	$ 3,548,404.75
	7	13,673	$ 3,391,137.51
Central African Rep	6	16,188	$ 4,069,794.44

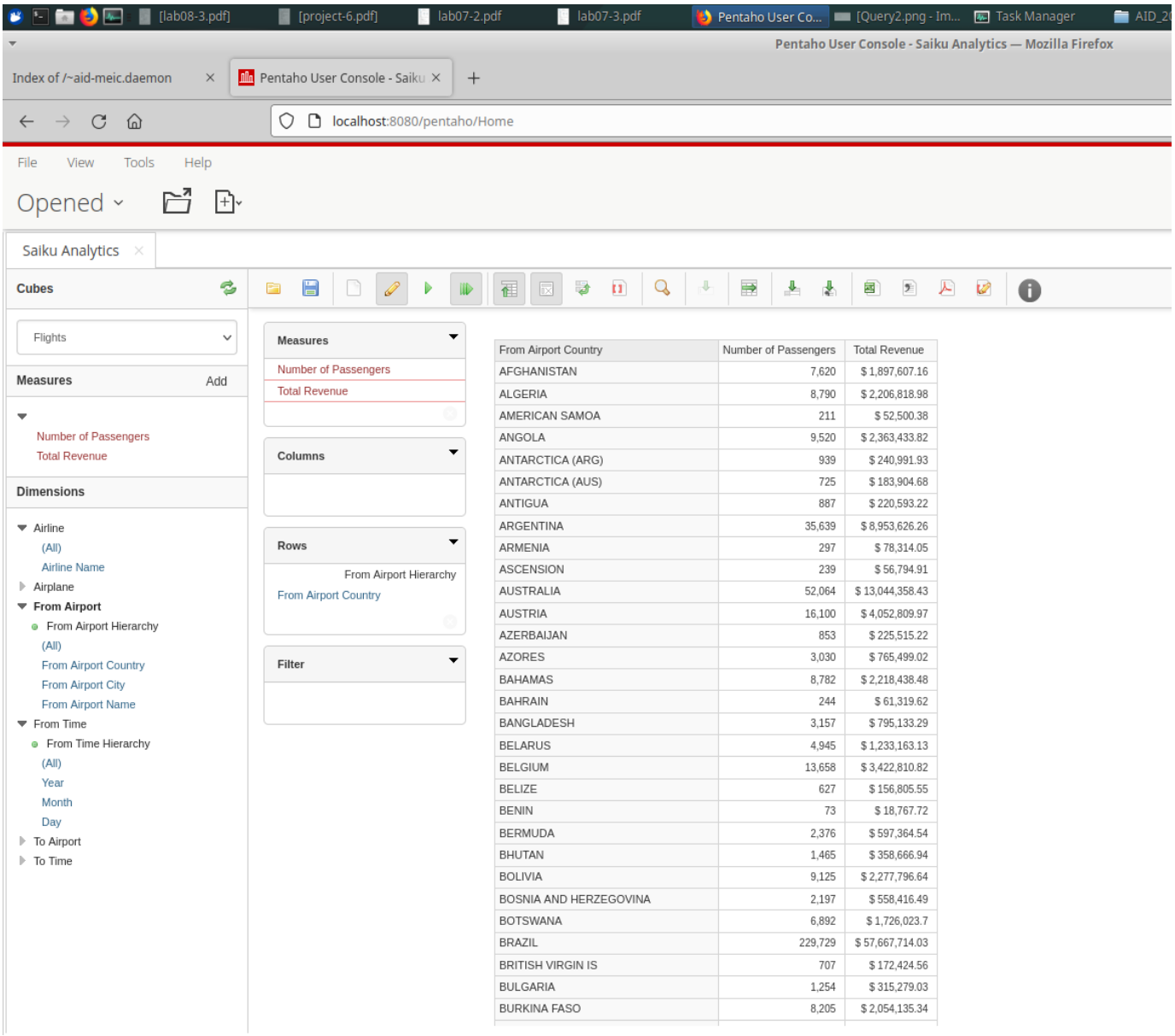
- Large-extra dataset



Airline Name	Month	Number of Passengers	Total Revenue
Afghanistan Airlines	6	156,551	$ 39,242,860.06
	7	161,431	$ 40,506,591.08
	8	161,349	$ 40,611,189.62
	9	6,008	$ 1,501,204.48
Albania Airlines	6	160,900	$ 40,317,887.11
	7	171,195	$ 42,962,087.82
	8	169,700	$ 42,733,218.77
	9	7,104	$ 1,798,014.84
American Samoa Airli	6	113,630	$ 28,596,600.25
	7	125,485	$ 31,481,026.9
	8	116,674	$ 29,270,331.76
	9	3,706	$ 924,285.72
Angola Airlines	6	138,719	$ 34,924,908.38
	7	147,906	$ 37,122,540.23
	8	139,958	$ 35,150,779.4
	9	4,298	$ 1,072,727.87
Argentina Airlines	6	159,546	$ 40,140,247.36
	7	170,885	$ 42,907,244.28
	8	157,747	$ 39,626,874.64
	9	5,234	$ 1,304,331.3
Australia Airlines	6	149,644	$ 37,496,872.7
	7	158,736	$ 39,792,355.19
	8	153,603	$ 38,515,357.81
	9	5,811	$ 1,465,376.56
Azerbaijan Airlines	6	118,271	$ 29,632,834.91
	7	124,665	$ 31,378,851.08
	8	123,890	$ 31,122,486.15
	9	5,245	$ 1,303,263.75
Bahamas Airlines	6	169,318	$ 42,449,030.51
	7	177,518	$ 44,579,334.98

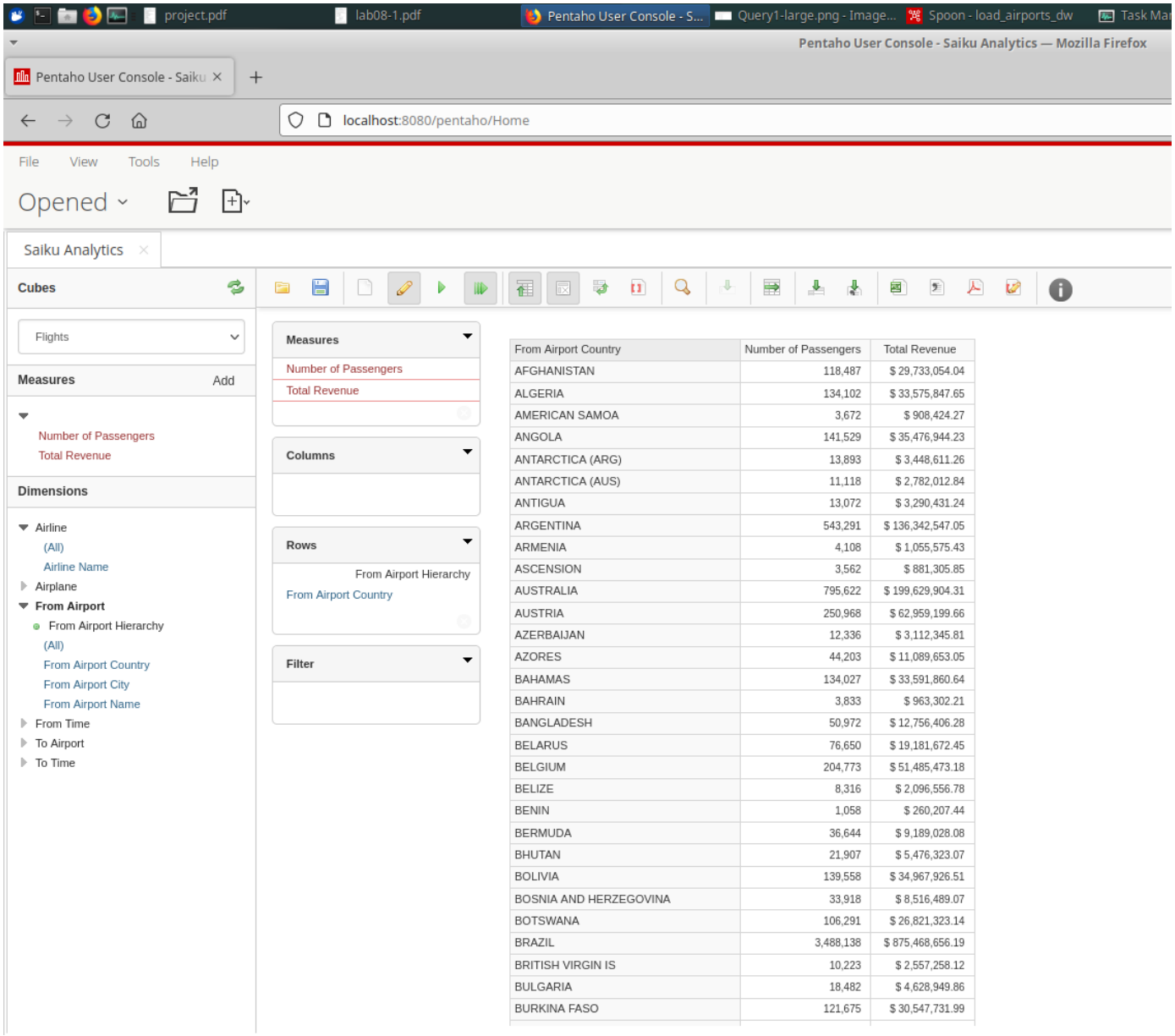
2- Passengers and revenue by country of departure

- Large dataset



From Airport Country	Number of Passengers	Total Revenue
AFGHANISTAN	7,620	$ 1,897,607.16
ALGERIA	8,790	$ 2,206,818.98
AMERICAN SAMOA	211	$ 52,500.38
ANGOLA	9,520	$ 2,363,433.82
ANTARCTICA (ARG)	939	$ 240,991.93
ANTARCTICA (AUS)	725	$ 183,904.68
ANTIGUA	887	$ 220,593.22
ARGENTINA	35,639	$ 8,953,626.26
ARMENIA	297	$ 78,314.05
ASCENSION	239	$ 56,794.91
AUSTRALIA	52,064	$ 13,044,358.43
AUSTRIA	16,100	$ 4,052,809.97
AZERBAIJAN	853	$ 225,515.22
AZORES	3,030	$ 765,499.02
BAHAMAS	8,782	$ 2,218,438.48
BAHRAIN	244	$ 61,319.62
BANGLADESH	3,157	$ 795,133.29
BELARUS	4,945	$ 1,233,163.13
BELGIUM	13,658	$ 3,422,810.82
BELIZE	627	$ 156,805.55
BENIN	73	$ 18,767.72
BERMUDA	2,376	$ 597,364.54
BHUTAN	1,465	$ 358,666.94
BOLIVIA	9,125	$ 2,277,796.64
BOSNIA AND HERZEGOVINA	2,197	$ 558,416.49
BOTSWANA	6,892	$ 1,726,023.7
BRAZIL	229,729	$ 57,667,714.03
BRITISH VIRGIN IS	707	$ 172,424.56
BULGARIA	1,254	$ 315,279.03
BURKINA FASO	8,205	$ 2,054,135.34

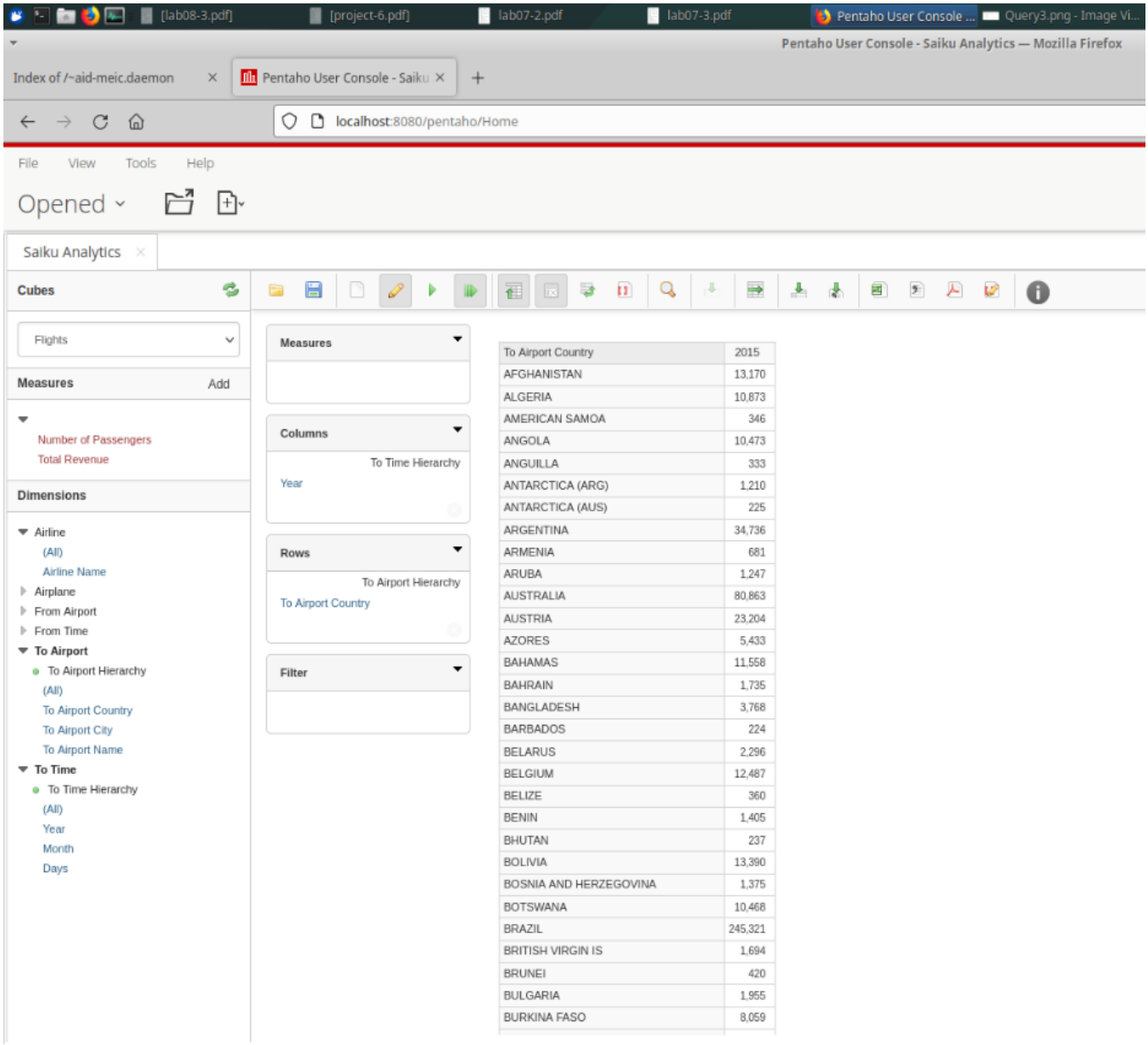
- Large-extra dataset



From Airport Country	Number of Passengers	Total Revenue
AFGHANISTAN	118,487	$ 29,733,054.04
ALGERIA	134,102	$ 33,575,847.65
AMERICAN SAMOA	3,672	$ 908,424.27
ANGOLA	141,529	$ 35,476,944.23
ANTARCTICA (ARG)	13,893	$ 3,448,611.26
ANTARCTICA (AUS)	11,118	$ 2,782,012.84
ANTIGUA	13,072	$ 3,290,431.24
ARGENTINA	543,291	$ 136,342,547.05
ARMENIA	4,108	$ 1,055,575.43
ASCENSION	3,562	$ 881,305.85
AUSTRALIA	795,622	$ 199,629,904.31
AUSTRIA	250,968	$ 62,959,199.66
AZERBAIJAN	12,336	$ 3,112,345.81
AZORES	44,203	$ 11,089,653.05
BAHAMAS	134,027	$ 33,591,860.64
BAHRAIN	3,833	$ 963,302.21
BANGLADESH	50,972	$ 12,756,406.28
BELARUS	76,650	$ 19,181,672.45
BELGIUM	204,773	$ 51,485,473.18
BELIZE	8,316	$ 2,096,556.78
BENIN	1,058	$ 260,207.44
BERMUDA	36,644	$ 9,189,028.08
BHUTAN	21,907	$ 5,476,323.07
BOLIVIA	139,558	$ 34,967,926.51
BOSNIA AND HERZEGOVINA	33,918	$ 8,516,489.07
BOTSWANA	106,291	$ 26,821,323.14
BRAZIL	3,488,138	$ 875,468,656.19
BRITISH VIRGIN IS	10,223	$ 2,557,258.12
BULGARIA	18,482	$ 4,628,949.86
BURKINA FASO	121,675	$ 30,547,731.99

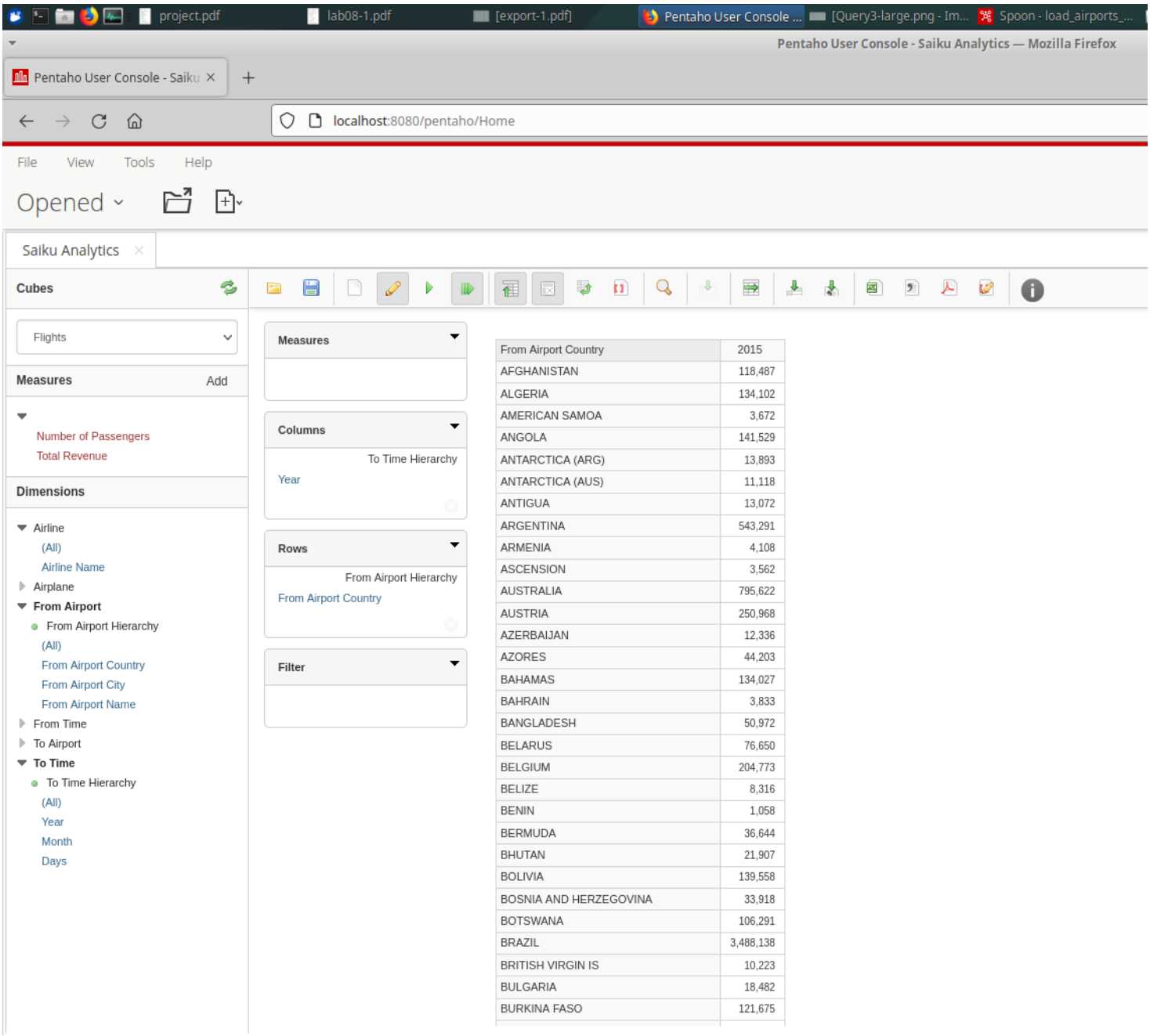
3- Number of flights by country of arrival per year

- Large dataset



To Airport Country	2015
AFGHANISTAN	13,170
ALGERIA	10,873
AMERICAN SAMOA	346
ANGOLA	10,473
ANGUILLA	333
ANTARCTICA (ARG)	1,210
ANTARCTICA (AUS)	225
ARGENTINA	34,736
ARMENIA	681
ARUBA	1,247
AUSTRALIA	80,863
AUSTRIA	23,204
AZORES	5,433
BAHAMAS	11,558
BAHRAIN	1,735
BANGLADESH	3,768
BARBADOS	224
BELARUS	2,296
BELGIUM	12,487
BELIZE	360
BENIN	1,405
BHUTAN	237
BOLIVIA	13,390
BOSNIA AND HERZEGOVINA	1,375
BOTSWANA	10,468
BRAZIL	245,321
BRITISH VIRGIN IS	1,694
BRUNEI	420
BULGARIA	1,955
BURKINA FASO	8,059

- Large-extra dataset



From Airport Country	2015
AFGHANISTAN	118,487
ALGERIA	134,102
AMERICAN SAMOA	3,672
ANGOLA	141,529
ANTARCTICA (ARG)	13,893
ANTARCTICA (AUS)	11,118
ANTIGUA	13,072
ARGENTINA	543,291
ARMENIA	4,108
ASCENSION	3,562
AUSTRALIA	795,622
AUSTRIA	250,968
AZERBAIJAN	12,336
AZORES	44,203
BAHAMAS	134,027
BAHRAIN	3,833
BANGLADESH	50,972
BELARUS	76,650
BELGIUM	204,773
BELIZE	8,316
BENIN	1,058
BERMUDA	36,644
BHUTAN	21,907
BOLIVIA	139,558
BOSNIA AND HERZEGOVINA	33,918
BOTSWANA	106,291
BRAZIL	3,488,138
BRITISH VIRGIN IS	10,223
BULGARIA	18,482
BURKINA FASO	121,675