Intro

This notebook is an example of how to use **Apache Flink** for processing simple data sets. We will take an open airline data set from stat-computing.org and find out who was the most popular carrier during 1998-2000 years. Next we will build a chart that shows flights distribution by months and look how it changes from year to year. We will use Zeppelin `%table` display system to build charts.

```
%md
### Getting the data
First we need to download and unpack the data. We will get three big data sets with flight details (one pack for each year) and a small one with carriers names. In total we will get for about 1,5 GB of d

<div class="markdown-body">
<h3>Getting the data</h3>
<p>First we need to download and unpack the data. We will get three big data sets with flight details (one pack for each year) and a small one with carriers names. In total we will get for about 1,5 GB o
</div>
```

```
%sh

rm /tmp/flights98.csv.bz2
curl -o /tmp/flights98.csv.bz2 "http://stat-computing.org/dataexpo/2009/1998.csv.bz2"
rm /tmp/flights98.csv
bzip2 -d /tmp/flights98.csv.bz2
chmod 666 /tmp/flights98.csv
```

```
rm: cannot remove '/tmp/flights98.csv.bz2': No such file or directory
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed

  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0 73.1M    0 64295    0     0  51646      0  0:24:44  0:00:01  0:24:43 51642
  0 73.1M    0  358k    0     0   160k      0  0:07:47  0:00:02  0:07:45  160k
  1 73.1M    1 1209k    0     0   373k      0  0:03:20  0:00:03  0:03:17  373k
  4 73.1M    4 3204k    0     0   773k      0  0:01:36  0:00:04  0:01:32  773k
  7 73.1M    7 5508k    0     0  1071k      0  0:01:09  0:00:05  0:01:04 1145k
 10 73.1M   10 7875k    0     0  1280k      0  0:00:58  0:00:06  0:00:52 1592k
 13 73.1M   13 10.1M    0     0  1458k      0  0:00:51  0:00:07  0:00:44 2049k
 17 73.1M   17 12.7M    0     0  1608k      0  0:00:46  0:00:08  0:00:38 2422k
 20 73.1M   20 14.9M    0     0  1671k      0  0:00:44  0:00:09  0:00:35 2413k
 23 73.1M   23 17.1M    0     0  1728k      0  0:00:43  0:00:10  0:00:33 2403k
 26 73.1M   26 19.4M    0     0  1787k      0  0:00:41  0:00:11  0:00:30 2411k
 29 73.1M   29 21.7M    0     0  1837k      0  0:00:40  0:00:12  0:00:28 2379k
 32 73.1M   32 24.1M    0     0  1879k      0  0:00:39  0:00:13  0:00:26 2322k
 36 73.1M   36 26.4M    0     0  1916k      0  0:00:39  0:00:14  0:00:25 2365k
 39 73.1M   39 28.5M    0     0  1930k      0  0:00:38  0:00:15  0:00:23 2341k
 41 73.1M   41 30.6M    0     0  1943k      0  0:00:38  0:00:16  0:00:22 2292k
 44 73.1M   44 32.6M    0     0  1947k      0  0:00:38  0:00:17  0:00:21 2215k
 47 73.1M   47 34.6M    0     0  1952k      0  0:00:38  0:00:18  0:00:20 2145k
 50 73.1M   50 36.6M    0     0  1960k      0  0:00:38  0:00:19  0:00:19 2082k
 52 73.1M   52 38.3M    0     0  1947k      0  0:00:38  0:00:20  0:00:18 1998k
 55 73.1M   55 40.4M    0     0  1956k      0  0:00:38  0:00:21  0:00:17 1996k
 57 73.1M   57 42.2M    0     0  1951k      0  0:00:38  0:00:22  0:00:16 1965k
 60 73.1M   60 44.0M    0     0  1948k      0  0:00:38  0:00:23  0:00:15 1932k
 62 73.1M   62 45.4M    0     0  1927k      0  0:00:38  0:00:24  0:00:14 1803k
 63 73.1M   63 46.5M    0     0  1896k      0  0:00:39  0:00:25  0:00:14 1688k
 65 73.1M   65 47.7M    0     0  1868k      0  0:00:40  0:00:26  0:00:14 1496k
 66 73.1M   66 48.8M    0     0  1843k      0  0:00:40  0:00:27  0:00:13 1363k
 68 73.1M   68 50.0M    0     0  1820k      0  0:00:41  0:00:28  0:00:13 1227k
 69 73.1M   69 51.1M    0     0  1786k      0  0:00:41  0:00:29  0:00:12 1126k
 71 73.1M   71 52.0M    0     0  1769k      0  0:00:42  0:00:30  0:00:12 1131k
 72 73.1M   72 53.0M    0     0  1744k      0  0:00:42  0:00:31  0:00:11 1098k
 73 73.1M   73 54.0M    0     0  1723k      0  0:00:43  0:00:32  0:00:11 1070k
 75 73.1M   75 55.1M    0     0  1702k      0  0:00:43  0:00:33  0:00:10 1040k
 76 73.1M   76 56.0M    0     0  1681k      0  0:00:44  0:00:34  0:00:10 1048k
 77 73.1M   77 56.9M    0     0  1659k      0  0:00:45  0:00:35  0:00:10  993k
 79 73.1M   79 57.8M    0     0  1638k      0  0:00:45  0:00:36  0:00:09  972k
 80 73.1M   80 58.7M    0     0  1618k      0  0:00:46  0:00:37  0:00:09  946k
 81 73.1M   81 59.6M    0     0  1600k      0  0:00:46  0:00:38  0:00:08  921k
 82 73.1M   82 60.5M    0     0  1582k      0  0:00:47  0:00:39  0:00:08  906k
 83 73.1M   83 61.4M    0     0  1566k      0  0:00:47  0:00:40  0:00:07  917k
 85 73.1M   85 62.1M    0     0  1546k      0  0:00:48  0:00:41  0:00:07  887k
 86 73.1M   86 63.0M    0     0  1532k      0  0:00:48  0:00:42  0:00:06  892k
 87 73.1M   87 63.9M    0     0  1517k      0  0:00:49  0:00:43  0:00:06  882k
 88 73.1M   88 64.8M    0     0  1503k      0  0:00:49  0:00:44  0:00:05  878k
 89 73.1M   89 65.6M    0     0  1489k      0  0:00:50  0:00:45  0:00:05  872k
 91 73.1M   91 66.5M    0     0  1477k      0  0:00:50  0:00:46  0:00:04  904k
 92 73.1M   92 67.4M    0     0  1465k      0  0:00:51  0:00:47  0:00:04  897k
 93 73.1M   93 68.2M    0     0  1451k      0  0:00:51  0:00:48  0:00:03  889k
 94 73.1M   94 69.2M    0     0  1441k      0  0:00:51  0:00:49  0:00:02  897k
 95 73.1M   95 70.1M    0     0  1430k      0  0:00:52  0:00:50  0:00:02  904k
 97 73.1M   97 71.0M    0     0  1421k      0  0:00:52  0:00:51  0:00:01  910k
 98 73.1M   98 71.9M    0     0  1413k      0  0:00:52  0:00:52 --:--:--  923k
 99 73.1M   99 72.8M    0     0  1403k      0  0:00:53  0:00:53 --:--:--  941k
100 73.1M  100 73.1M    0     0  1401k      0  0:00:53  0:00:53 --:--:--  941k
rm: cannot remove '/tmp/flights98.csv': No such file or directory
```

```
%sh

rm /tmp/flights99.csv.bz2
curl -o /tmp/flights99.csv.bz2 "http://stat-computing.org/dataexpo/2009/1999.csv.bz2"
rm /tmp/flights99.csv
bzip2 -d /tmp/flights99.csv.bz2
chmod 666 /tmp/flights99.csv
```

```
rm: cannot remove '/tmp/flights99.csv.bz2': No such file or directory
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed

  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0 75.7M    0  5520    0     0   9851      0  2:14:25 --:--:--  2:14:25  9839
  0 75.7M    0 88819    0     0  64302      0  0:20:35  0:00:01  0:20:34 64268
  0 75.7M    0  181k    0     0  25316      0  0:52:18  0:00:07  0:52:11 25316
  0 75.7M    0  548k    0     0  67331      0  0:19:39  0:00:08  0:19:31 67327
  1 75.7M    1  817k    0     0  89344      0  0:14:49  0:00:09  0:14:40 89337
  1 75.7M    1 1042k    0     0   100k      0  0:12:54  0:00:10  0:12:44  105k
  3 75.7M    3 2461k    0     0   218k      0  0:05:55  0:00:11  0:05:44  239k
  6 75.7M    6 5069k    0     0   412k      0  0:03:08  0:00:12  0:02:56  985k
 11 75.7M   11 9165k    0     0   690k      0  0:01:52  0:00:13  0:01:39 1744k
 14 75.7M   14 11.2M    0     0   796k      0  0:01:37  0:00:14  0:01:23 2109k
 19 75.7M   19 14.8M    0     0   995k      0  0:01:17  0:00:15  0:01:02 2910k
 24 75.7M   24 18.6M    0     0  1174k      0  0:01:06  0:00:16  0:00:50 3331k
 29 75.7M   29 22.5M    0     0  1338k      0  0:00:57  0:00:17  0:00:40 3613k
 35 75.7M   35 26.5M    0     0  1486k      0  0:00:52  0:00:18  0:00:34 3603k
 40 75.7M   40 30.3M    0     0  1610k      0  0:00:48  0:00:19  0:00:29 4025k
 45 75.7M   45 34.2M    0     0  1731k      0  0:00:44  0:00:20  0:00:24 3980k
 50 75.7M   50 38.2M    0     0  1840k      0  0:00:42  0:00:21  0:00:21 4011k
 55 75.7M   55 42.2M    0     0  1940k      0  0:00:39  0:00:22  0:00:17 4020k
 60 75.7M   60 46.2M    0     0  2032k      0  0:00:38  0:00:23  0:00:15 4026k
 65 75.7M   65 49.9M    0     0  2106k      0  0:00:36  0:00:24  0:00:12 4017k
 70 75.7M   70 53.5M    0     0  2169k      0  0:00:35  0:00:25  0:00:10 3945k
 75 75.7M   75 57.2M    0     0  2229k      0  0:00:34  0:00:26  0:00:08 3884k
 80 75.7M   80 61.1M    0     0  2293k      0  0:00:33  0:00:27  0:00:06 3868k
 86 75.7M   86 65.5M    0     0  2372k      0  0:00:32  0:00:28  0:00:04 3956k
 92 75.7M   92 70.4M    0     0  2464k      0  0:00:31  0:00:29  0:00:02 4200k
100 75.7M  100 75.7M    0     0  2565k      0  0:00:30  0:00:30 --:--:-- 4585k
rm: cannot remove '/tmp/flights99.csv': No such file or directory
```

```
%sh

rm /tmp/flights00.csv.bz2
curl -o /tmp/flights00.csv.bz2 "http://stat-computing.org/dataexpo/2009/2000.csv.bz2"
rm /tmp/flights00.csv
bzip2 -d /tmp/flights00.csv.bz2
chmod 666 /tmp/flights00.csv
```

```
rm: cannot remove '/tmp/flights00.csv.bz2': No such file or directory
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed

  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  0     0    0     0    0     0      0      0 --:--:--  0:00:01 --:--:--     0
  0 78.7M    0  5520    0     0   3016      0  7:36:06  0:00:01  7:36:05  3014
  0 78.7M    0 39987    0     0  15337      0  1:29:41  0:00:02  1:29:39 15332
  0 78.7M    0 87755    0     0  24531      0  0:56:04  0:00:03  0:56:01 24526
  0 78.7M    0  157k    0     0  33950      0  0:40:31  0:00:04  0:40:27 33944
  0 78.7M    0  221k    0     0  40878      0  0:33:39  0:00:05  0:33:34 53734
  0 78.7M    0  308k    0     0  47250      0  0:29:06  0:00:06  0:29:00 63943
  0 78.7M    0  398k    0     0  52806      0  0:26:03  0:00:07  0:25:56 71903
  0 78.7M    0  437k    0     0  36667      0  0:37:31  0:00:12  0:37:19 41697
  0 78.7M    0  703k    0     0  57158      0  0:24:04  0:00:12  0:23:52 71137
  1 78.7M    1  851k    0     0  64259      0  0:21:24  0:00:13  0:21:11 80471
  1 78.7M    1 1171k    0     0  82442      0  0:16:41  0:00:14  0:16:27  109k
  1 78.7M    1 1546k    0     0  79861      0  0:17:13  0:00:19  0:16:54 97134
  3 78.7M    3 3181k    0     0   154k      0  0:08:41  0:00:20  0:08:21  327k
  4 78.7M    4 3466k    0     0   160k      0  0:08:21  0:00:21  0:08:00  308k
  4 78.7M    4 3565k    0     0   136k      0  0:09:50  0:00:26  0:09:24  216k
  8 78.7M    8 7196k    0     0   270k      0  0:04:57  0:00:26  0:04:31  501k
 10 78.7M   10 8459k    0     0   307k      0  0:04:22  0:00:27  0:03:55  894k
 11 78.7M   11 9386k    0     0   327k      0  0:04:06  0:00:28  0:03:38  768k
 15 78.7M   15 11.9M    0     0   413k      0  0:03:14  0:00:29  0:02:45 1093k
 18 78.7M   18 14.5M    0     0   487k      0  0:02:45  0:00:30  0:02:15 2553k
 22 78.7M   22 17.7M    0     0   574k      0  0:02:20  0:00:31  0:01:49 2195k
 25 78.7M   25 19.9M    0     0   626k      0  0:02:08  0:00:32  0:01:36 2375k
 28 78.7M   28 22.1M    0     0   676k      0  0:01:59  0:00:33  0:01:26 2726k
 31 78.7M   31 24.7M    0     0   734k      0  0:01:49  0:00:34  0:01:15 2643k
 34 78.7M   34 27.3M    0     0   789k      0  0:01:42  0:00:35  0:01:07 2638k
 38 78.7M   38 30.0M    0     0   841k      0  0:01:35  0:00:36  0:00:59 2513k
 40 78.7M   40 32.1M    0     0   874k      0  0:01:32  0:00:37  0:00:55 2457k
 43 78.7M   43 34.1M    0     0   906k      0  0:01:28  0:00:38  0:00:50 2445k
 45 78.7M   45 35.7M    0     0   925k      0  0:01:27  0:00:39  0:00:48 2250k
 47 78.7M   47 37.4M    0     0   946k      0  0:01:25  0:00:40  0:00:45 2062k
 49 78.7M   49 39.3M    0     0   968k      0  0:01:23  0:00:41  0:00:42 1907k
 52 78.7M   52 41.0M    0     0   987k      0  0:01:21  0:00:42  0:00:39 1859k
 54 78.7M   54 42.5M    0     0  1000k      0  0:01:20  0:00:43  0:00:37 1729k
 55 78.7M   55 43.9M    0     0  1008k      0  0:01:19  0:00:44  0:00:35 1651k
 57 78.7M   57 45.4M    0     0  1020k      0  0:01:18  0:00:45  0:00:33 1625k
 59 78.7M   59 46.6M    0     0  1027k      0  0:01:18  0:00:46  0:00:32 1512k
 60 78.7M   60 47.7M    0     0  1027k      0  0:01:18  0:00:47  0:00:31 1376k
 61 78.7M   61 48.6M    0     0  1024k      0  0:01:18  0:00:48  0:00:30 1236k
 62 78.7M   62 49.5M    0     0  1020k      0  0:01:18  0:00:49  0:00:29 1125k
 64 78.7M   64 50.4M    0     0  1021k      0  0:01:18  0:00:50  0:00:28 1027k
 65 78.7M   65 51.3M    0     0  1018k      0  0:01:19  0:00:51  0:00:28  941k
 66 78.7M   66 52.1M    0     0  1016k      0  0:01:19  0:00:52  0:00:27  910k
 67 78.7M   67 53.0M    0     0  1014k      0  0:01:19  0:00:53  0:00:26  909k
 68 78.7M   68 53.7M    0     0  1006k      0  0:01:20  0:00:54  0:00:26  868k
 69 78.7M   69 54.6M    0     0  1006k      0  0:01:20  0:00:55  0:00:25  858k
 70 78.7M   70 55.3M    0     0  1002k      0  0:01:20  0:00:56  0:00:24  831k
 71 78.7M   71 56.1M    0     0   998k      0  0:01:20  0:00:57  0:00:23  807k
 72 78.7M   72 56.9M    0     0   994k      0  0:01:21  0:00:58  0:00:23  787k
 73 78.7M   73 57.6M    0     0   991k      0  0:01:21  0:00:59  0:00:22  823k
 74 78.7M   74 58.4M    0     0   988k      0  0:01:21  0:01:00  0:00:21  784k
 75 78.7M   75 59.2M    0     0   985k      0  0:01:21  0:01:01  0:00:20  791k
 76 78.7M   76 60.0M    0     0   982k      0  0:01:22  0:01:02  0:00:20  797k
 77 78.7M   77 60.8M    0     0   980k      0  0:01:22  0:01:03  0:00:19  808k
 78 78.7M   78 61.6M    0     0   977k      0  0:01:22  0:01:04  0:00:18  812k
 79 78.7M   79 62.4M    0     0   975k      0  0:01:22  0:01:05  0:00:17  824k
 80 78.7M   80 63.4M    0     0   976k      0  0:01:22  0:01:06  0:00:16  870k
 82 78.7M   82 64.9M    0     0   984k      0  0:01:21  0:01:07  0:00:14 1006k
 85 78.7M   85 66.9M    0     0  1000k      0  0:01:20  0:01:08  0:00:12 1254k
 88 78.7M   88 69.4M    0     0  1022k      0  0:01:18  0:01:09  0:00:09 1602k
 92 78.7M   92 72.5M    0     0  1053k      0  0:01:16  0:01:10  0:00:06 2064k
 96 78.7M   96 76.1M    0     0  1089k      0  0:01:13  0:01:11  0:00:02 2600k
100 78.7M  100 78.7M    0     0  1116k      0  0:01:12  0:01:12 --:--:-- 3022k
rm: cannot remove '/tmp/flights00.csv': No such file or directory
```

```
%sh

rm /tmp/carriers.csv
curl -o /tmp/carriers.csv "http://stat-computing.org/dataexpo/2009/carriers.csv"
chmod 666 /tmp/carriers.csv
```

```
rm: cannot remove '/tmp/carriers.csv': No such file or directory
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed

  0     0    0     0    0     0      0      0 --:--:-- --:--:-- --:--:--     0
  9 43758    9  4140    0     0   7588      0  0:00:05 --:--:--  0:00:05  7582
100 43758  100 43758    0     0  46357      0 --:--:-- --:--:-- --:--:-- 46353
```

Preparing the data

The `flights<YY>.csv` contains various data but we only need the information about the year, the month and the carrier who served the flight. Let's retrieve this information and create `DataSets`.

```
%flink

case class Flight(year: Int, month: Int, carrierCode: String)
case class Carrier(code: String, name: String)

val flights98 = benv.readCsvFile[Flight]("/tmp/flights98.csv", ignoreFirstLine = true, includedFields = Array(0, 1, 8))
val flights99 = benv.readCsvFile[Flight]("/tmp/flights99.csv", ignoreFirstLine = true, includedFields = Array(0, 1, 8))
val flights00 = benv.readCsvFile[Flight]("/tmp/flights00.csv", ignoreFirstLine = true, includedFields = Array(0, 1, 8))
val flights = flights98.union(flights99).union(flights00)
val carriers = benv.readCsvFile[Carrier]("/tmp/carriers.csv", ignoreFirstLine = true, quoteCharacter = '"')
```

```
defined class Flight
defined class Carrier
flights98: org.apache.flink.api.scala.DataSet[Flight] = org.apache.flink.api.scala.DataSet@7cd81fd5
flights99: org.apache.flink.api.scala.DataSet[Flight] = org.apache.flink.api.scala.DataSet@58242e79
flights00: org.apache.flink.api.scala.DataSet[Flight] = org.apache.flink.api.scala.DataSet@13f866c0
flights: org.apache.flink.api.scala.DataSet[Flight] = org.apache.flink.api.scala.DataSet@2aad2530
carriers: org.apache.flink.api.scala.DataSet[Carrier] = org.apache.flink.api.scala.DataSet@148c977b
```

Choosing the carrier

Now we will search for the most popular carrier during the whole time period.

```
%flink

import org.apache.flink.api.common.operators.Order
import org.apache.flink.api.java.aggregation.Aggregations

case class CarrierFlightsCount(carrierCode: String, count: Int)
case class CountByMonth(month: Int, count: Int)

val carriersFlights = flights
  .map(f => CarrierFlightsCount(f.carrierCode, 1))
  .groupBy("carrierCode")
  .sum("count")

val maxFlights = carriersFlights
  .aggregate(Aggregations.MAX, "count")

val bestCarrier = carriersFlights
  .join(maxFlights)
  .where("count")
  .equalTo("count")
  .map(_._1)
```

```
val carrierName = bestCarrier
  .join(carriers)
  .where("carrierCode")
  .equalTo("code")
  .map(_._2.name)
  .collect
  .head
```

```
import org.apache.flink.api.common.operators.Order
import org.apache.flink.api.java.aggregation.Aggregations
defined class CarrierFlightsCount
defined class CountByMonth
carriersFlights: org.apache.flink.api.scala.AggregateDataSet[CarrierFlightsCount] = org.apache.flink.api.scala.AggregateDataSet@2c59be0b
maxFlights: org.apache.flink.api.scala.AggregateDataSet[CarrierFlightsCount] = org.apache.flink.api.scala.AggregateDataSet@53e5fad9
bestCarrier: org.apache.flink.api.scala.DataSet[CarrierFlightsCount] = org.apache.flink.api.scala.DataSet@64b7b1b3
carrierName: String = Delta Air Lines Inc.
```

```
%flink

println(s"""The most popular carrier is:
$carrierName
""")
```

```
The most popular carrier is:
Delta Air Lines Inc.
```

Calculating flights

The last step is to filter **Delta Air Lines** flights and group them by months.

```
%flink

def countFlightsPerMonth(flights: DataSet[Flight],
                         carrier: DataSet[CarrierFlightsCount]) = {
  val carrierFlights = flights
    .join(carrier)
    .where("carrierCode")
    .equalTo("carrierCode")
    .map(_._1)

  carrierFlights
    .map(flight => CountByMonth(flight.month, 1))
    .groupBy("month")
    .sum("count")
    .sortPartition("month", Order.ASCENDING)
}

val bestCarrierFlights_98 = countFlightsPerMonth(flights98, bestCarrier)
val bestCarrierFlights_99 = countFlightsPerMonth(flights99, bestCarrier)
val bestCarrierFlights_00 = countFlightsPerMonth(flights00, bestCarrier)
```

```
countFlightsPerMonth: (flights: org.apache.flink.api.scala.DataSet[Flight], carrier: org.apache.flink.api.scala.DataSet[CarrierFlightsCount])org.apache.flink.api.scala.DataSet[CountByMonth]
bestCarrierFlights_98: org.apache.flink.api.scala.DataSet[CountByMonth] = org.apache.flink.api.scala.PartitionSortedDataSet@2aa64309
bestCarrierFlights_99: org.apache.flink.api.scala.DataSet[CountByMonth] = org.apache.flink.api.scala.PartitionSortedDataSet@35fe60c4
bestCarrierFlights_00: org.apache.flink.api.scala.DataSet[CountByMonth] = org.apache.flink.api.scala.PartitionSortedDataSet@4621410f
```

```
%flink

def monthAsString(month: Int): String = {
  month match {
    case 1 => "Jan"
    case 2 => "Feb"
    case 3 => "Mar"
    case 4 => "Apr"
    case 5 => "May"
    case 6 => "Jun"
    case 7 => "Jul"
    case 8 => "Aug"
    case 9 => "Sept"
    case 10 => "Oct"
    case 11 => "Nov"
    case 12 => "Dec"
  }
}

// We should put all the results into a common DataFrame
// to show them in a common picture
val bestCarrierFlights = bestCarrierFlights_98
  .join(bestCarrierFlights_99)
  .where("month")
  .equalTo("month")
  .map(tuple => (tuple._1.month, tuple._1.count, tuple._2.count))
  .join(bestCarrierFlights_00)
  .where(0)
  .equalTo("month")
  .map(tuple => (tuple._1._1, tuple._1._2, tuple._1._3, tuple._2.count))
  .collect

var flightsByMonthTable = s"Month\t1998\t1999\t2000\n"
bestCarrierFlights.foreach(data => flightsByMonthTable += s"${monthAsString(data._1)}\t${data._2}\t${data._3}\t${data._4}\n")
```

```
monthAsString: (month: Int)String
bestCarrierFlights: Seq[(Int, Int, Int, Int)] = Buffer((1,78523,77745,78055), (2,71101,70498,71090), (3,78906,77812,78453), (4,75726,75343,75247), (5,77937,77226,76797), (6,75432,75840,74846), (7,77521,7
flightsByMonthTable: String =
"Month  1998    1999    2000
"
```

```
%flink

println(s"""%table
$flightsByMonthTable
""")
```

```
Month   1998    1999    2000
Jan     78523   77745   78055
Feb     71101   70498   71090
Mar     78906   77812   78453
Apr     75726   75343   75247
May     77937   77226   76797
Jun     75432   75840   74846
Jul     77521   77264   75776
Aug     78104   78141   77654
Sept    74840   75067   73696
Oct     76145   77829   77425
Nov     73552   74411   73659
Dec     77308   76954   75331
```

Results

Looking at this chart we can say that February is the most unpopular month, but this is only because it has less days (28 or 29) than the other months (30 or 31). To receive more fair picture we should calculate the average flights count per day for each month.

```
%flink

def daysInMonth(month: Int, year: Int): Int = {
  month match {
    case 1 => 31
    case 2 => if (year % 4 == 0) {
        29
    } else {
        28
    }
    case 3 => 31
    case 4 => 30
```

```
    case 5 => 31
    case 6 => 30
    case 7 => 31
    case 8 => 31
    case 9 => 30
    case 10 => 31
    case 11 => 30
    case 12 => 31
  }
}


var flightsByDayTable = s"Month\t1998\t1999\t2000\n"

bestCarrierFlights.foreach(data => flightsByDayTable += s"${monthAsString(data._1)}\t${data._2/daysInMonth(data._1,1998)}\t${data._3/daysInMonth(data._1,1999)}\t${data._4/daysInMonth(data._1,2000)}\n")

daysInMonth: (month: Int, year: Int)Int
flightsByDayTable: String =
"Month  1998    1999    2000
"

%flink

println(s"""%table
$flightsByDayTable
""")

Month   1998    1999    2000
Jan     2533    2507    2517
Feb     2539    2517    2451
Mar     2545    2510    2530
Apr     2524    2511    2508
May     2514    2491    2477
Jun     2514    2528    2494
Jul     2500    2492    2444
Aug     2519    2520    2504
Sept    2494    2502    2456
Oct     2456    2510    2497
Nov     2451    2480    2455
Dec     2493    2482    2430

%flink
```