In [83]:
```
airports_path = "/data/students/bigdata_internet/lab5/airports.csv"
airlines_path = "/data/students/bigdata_internet/lab5/airlines.csv"
routes_path = "/data/students/bigdata_internet/lab5/routes.csv"
```

In [84]:
```
airports_df = spark.read.load(airports_path, format = "csv", header = True
airlines_df = spark.read.load(airlines_path, format = "csv", header = True
routes_df = spark.read.load(routes_path, format = "csv", header = True, in
```

2 Task 1: Which are the countries in the world with more than 100 airports? Reports these countries and their number of airports.

In [85]:
```
def countAirports(country):
    isinstanceairport = 1
    return isinstanceairport
```

In [86]:
```
spark.udf.register("isAirportInstance", lambda country: countAirports(coun
```

Out[86]:
```
<function __main__.<lambda>(country)>
```

In [87]:
```
country_total_airports_df = airports_df.selectExpr("country","isAirportIns
```

In [88]:
```
country_total_df = country_total_airports_df.groupBy("country").sum("insta
```

In [89]:
```
country_more_than_100 = country_total_df.filter("total_airports>=100")
```

In [90]:
```
country_more_than_100.show()
```

```
+--------------+--------------+
|       country|total_airports|
+--------------+--------------+
|        Russia|           264|
|       Germany|           249|
|        France|           217|
|         India|           148|
|         China|           241|
| United States|          1512|
|     Indonesia|           145|
|        Canada|           430|
|        Brazil|           264|
|         Japan|           123|
|     Australia|           334|
|United Kingdom|           167|
+--------------+--------------+
```

2 Task 2: Which are the Top-10 airlines by total number of flights? For each airline in the Top-10, provide airline name, airline icao code and number of flights.

In [91]:
```
def countFlights(airline):
    isFlight = 1
    return isFlight
```

In [92]:
```
spark.udf.register("isFlight",lambda airline_id: countFlights(airline_id),
```

Out[92]:
```
<function __main__.<lambda>(airline_id)>
```

In [93]:
```
airline_flights_df = routes_df.selectExpr("airline_id","isFLight(airline_i
```

In [94]:
```
airline_total_flights_df = airline_flights_df.groupBy("airline_id").sum("fl
```

In [95]:
```
airline_names = airlines_df.select("airline_id","name","icao")
```

In [96]:
```
airline_joined = airline_total_flights_df.join(airline_names,airline_total_
```

In [97]:
```
airlines_sorted = airline_joined.sort("total_flights",ascending = False)
```

In [98]:
```
airlines_sorted.show(10)
```

```
+--------------------+----+-------------+
|                name|icao|total_flights|
+--------------------+----+-------------+
|             Ryanair| RYR|         2484|
|   American Airlines| AAL|         2354|
|     United Airlines| UAL|         2180|
|     Delta Air Lines| DAL|         1981|
|          US Airways| USA|         1960|
|China Southern Ai...| CSN|         1454|
|China Eastern Air...| CES|         1263|
|           Air China| CCA|         1260|
|  Southwest Airlines| SWA|         1146|
|             easyJet| EZY|         1130|
+--------------------+----+-------------+
only showing top 10 rows
```

2 Task 3: Which are the Top-10 airports by number of departing flights? For each airport in the Top-10, provide its name, its iata code and the number of departing flights.

In [99]:
```
airline_source_df = routes_df.selectExpr("airport_source","isFLight(airlin
```

In [100…
```
airport_departures = airline_source_df.groupBy("airport_source").sum("fligh
```

In [101…
```
ordered_airport_departures = airport_departures.sort("total_departures",asc
```

In [102…
```
airportName_iata = airports_df.select("name","iata")
```

In [103…
```
airport_name_iata_departingflights = ordered_airport_departures\
.join(airportName_iata,ordered_airport_departures\
      .airport_source == airportName_iata.iata,"inner").select("name","iata
```

In [104…
```
airport_name_iata_departingflights.show(10)
```

```
+--------------------+----+---------------+
|                name|iata|total_departures|
+--------------------+----+---------------+
|Hartsfield Jackso...| ATL|            915|
|Chicago O'Hare In...| ORD|            558|
|Beijing Capital I...| PEK|            535|
|London Heathrow A...| LHR|            527|
|Charles de Gaulle...| CDG|            524|
|Frankfurt am Main...| FRA|            497|
|Los Angeles Inter...| LAX|            492|
|Dallas Fort Worth...| DFW|            469|
|John F Kennedy In...| JFK|            456|
|Amsterdam Airport...| AMS|            453|
+--------------------+----+---------------+
only showing top 10 rows
```

Removing erroneous Lines

In [105…
```
def finderroneouslines(a_source_id,a_dest_id,a_source,a_dest):
    ERRORCODE = str('\\\\N')
    if a_source_id == ERRORCODE or a_dest_id == ERRORCODE or a_source == E
        return False
    else:
        return True
```

In [106…
```
spark.udf.register("IsErroneous",lambda airport_source_id,airport_destinat
```

Out[106…
```
<function __main__.<lambda>(airport_source_id, airport_destination_id, airp
ort_source, airport_destination)>
```

In [107…
```
#erroneous_routes = routes_df.filter(lambda airport_source,airport_destina
non_erroneous_routes = routes_df.filter( "IsErroneous(airport_source_id,ai
```

In [108…
```
airports_for_nodes = airports_df.withColumn("id", airports_df.id.cast("str
```

In [109…
```
routes_for_edges= routes_df.withColumn("airport_source_id", routes_df.airpo
.withColumn("airport_destination_id", routes_df.airport_destination_id.cast
.withColumnRenamed("airport_destination_id","dst")
```

In [110…
```
nodes_df = airports_for_nodes
#nodes_df.show(1)
#nodes_df.printSchema()
edges_df = routes_for_edges
#edges_df.show(1)
#edges_df.printSchema()
from graphframes import GraphFrame
g = GraphFrame(nodes_df, edges_df)
```

4) Analyze and process the graph Task 1: Show top-10 airports by in and out degree. Please provide the name of the airport as well, its ID and its degree. In degree is the number of incoming edges (oriented graph) out degree is the number of outgoing edges

In [111…
```
airports_id_name = airports_df.select("id","name").withColumn("id", airport
```

In [112…
```
gInDeg = g.inDegrees
gOutDeg = g.outDegrees
```

In [113…
```
gInDegSorted=gInDeg.sort("inDegree", ascending=False)
gInDegSorted_And_Name = gInDegSorted.join(airports_id_name,gInDegSorted.id
gInDegSorted_And_Name.show(2)

gOutDegSorted=gOutDeg.sort("outDegree", ascending=False)
gOutDegSorted_And_Name = gOutDegSorted.join(airports_id_name,gOutDegSorted
gOutDegSorted_And_Name.show(2)
```

```
+----------+--------------------+--------+
|airport_id|                name|inDegree|
+----------+--------------------+--------+
|      3682|Hartsfield Jackso...|     911|
|      3830|Chicago O'Hare In...|     550|
+----------+--------------------+--------+
only showing top 2 rows

+----------+--------------------+---------+
|airport_id|                name|outDegree|
+----------+--------------------+---------+
|      3682|Hartsfield Jackso...|      915|
|      3830|Chicago O'Hare In...|      558|
+----------+--------------------+---------+
only showing top 2 rows
```

4) Analyze and process the graph Task 2: How many airports are reachable from Turin taking exactly 1 flight? What about taking 2 flights? And 3 flights? Hint: Use the motif finding functionality. Turin has id = 1526

In [114…
```
motifs = g.find("(a)-[e]->(b)")
turin_motif = motifs.filter("a.id = 1526")
turin_motif_distinct = turin_motif.select("a","b").distinct()
turin_motif_distinct.show(200)
turin_motif_distinct.count()
```

```
+--------------------+--------------------+
|                   a|                   b|
+--------------------+--------------------+
|[1526, Turin Airp...|[1606, Malta Inte...|
|[1526, Turin Airp...|[1508, Lamezia Te...|
|[1526, Turin Airp...|[1655, Iaşi Airpo...|
|[1526, Turin Airp...|[1515, Vincenzo F...|
|[1526, Turin Airp...|[1561, Naples Int...|
|[1526, Turin Airp...|[340, Frankfurt a...|
|[1526, Turin Airp...|[1218, Barcelona ...|
|[1526, Turin Airp...|[1520, Olbia Cost...|
|[1526, Turin Airp...|[1506, Brindisi —...|
|[1526, Turin Airp...|[502, London Gatw...|
|[1526, Turin Airp...|[548, London Stan...|
|[1526, Turin Airp...|[1555, Leonardo d...|
|[1526, Turin Airp...|[345, Düsseldorf ...|
|[1526, Turin Airp...|[1514, Reggio Cal...|
|[1526, Turin Airp...|[580, Amsterdam A...|
|[1526, Turin Airp...|[1509, Catania-Fo...|
|[1526, Turin Airp...|[1701, Atatürk In...|
|[1526, Turin Airp...|[1074, Mohammed V...|
|[1526, Turin Airp...|[1512, Falcone—Bo...|
|[1526, Turin Airp...|[1382, Charles de...|
|[1526, Turin Airp...|[1519, Cagliari E...|
|[1526, Turin Airp...|[1517, Alghero-Fe...|
|[1526, Turin Airp...|[1501, Bari Karol...|
|[1526, Turin Airp...|[302, Brussels Ai...|
|[1526, Turin Airp...|[1229, Adolfo Suá...|
|[1526, Turin Airp...|[304, Brussels So...|
|[1526, Turin Airp...|[346, Munich Airp...|
|[1526, Turin Airp...|[1678, Zürich Air...|
|[1526, Turin Airp...|[1190, Tirana Int...|
+--------------------+--------------------+
```

Out[114… 29

In [115…

```
motifsTwoFlights = g.find("(a)-[e]->(b); (b)-[e2]->(c)")
turin_motifTwoFlights = motifsTwoFlights.filter("a.id = 1526")
turin_motifTwoFlights.show(1)



turin_motifTwoFlights_distinct = turin_motifTwoFlights.select("a","c").dis
turin_motifTwoFlights_distinct.show(5)
turin_motifTwoFlights_distinct.count()
```

```
+--------------------+--------------------+--------------------+----------
---------+--------------------+
|                   a|                   e|                   b|
e2|                   c|
+--------------------+--------------------+--------------------+----------
---------+--------------------+
|[1526, Turin Airp...|[4U, 2548, TRN, 1...|[345, Düsseldorf ...|[YM, 3539,
DUS, 3...|[1741, Podgorica ...|
+--------------------+--------------------+--------------------+----------
---------+--------------------+
only showing top 1 row

+--------------------+--------------------+
|                   a|                   c|
+--------------------+--------------------+
|[1526, Turin Airp...|[1606, Malta Inte...|
|[1526, Turin Airp...|[286, Monastir Ha...|
|[1526, Turin Airp...|[2121, Esfahan Sh...|
|[1526, Turin Airp...|[3751, Denver Int...|
|[1526, Turin Airp...|[1206, Split Airp...|
+--------------------+--------------------+
only showing top 5 rows
```

Out[115… 590

In [116…

```
motifsThreeFlights = g.find("(a)-[e]->(b); (b)-[e2]->(c); (c)-[e3]->(d)")
turin_motifThreeFlights = motifsThreeFlights.filter("a.id = 1526")
turin_motifThreeFlights.show(1)


turin_motifThreeFlights_distinct = turin_motifThreeFlights.select("a","d")
turin_motifThreeFlights_distinct.show(5)
turin_motifThreeFlights_distinct.count()
```

```
+--------------------+--------------------+--------------------+----------
---------+--------------------+--------------------+--------------------+
|                   a|                   e|                   b|
e2|                   c|                  e3|                   d|
+--------------------+--------------------+--------------------+----------
---------+--------------------+--------------------+--------------------+
|[1526, Turin Airp...|[4U, 2548, TRN, 1...|[345, Düsseldorf ...|[YM, 3539,
DUS, 3...|[1741, Podgorica ...|[YM, 3539, TGD, 1...|[1678, Zürich Air...|
+--------------------+--------------------+--------------------+----------
---------+--------------------+--------------------+--------------------+
only showing top 1 row

+--------------------+--------------------+
|                   a|                   d|
+--------------------+--------------------+
|[1526, Turin Airp...|[1606, Malta Inte...|
|[1526, Turin Airp...|[286, Monastir Ha...|
|[1526, Turin Airp...|[2121, Esfahan Sh...|
|[1526, Turin Airp...|[2955, Abakan Air...|
|[1526, Turin Airp...|[1374, Châlons-Va...|
+--------------------+--------------------+
only showing top 5 rows
```

Out[116… 2210

4. Analyze and process the graph Task 3: Compute the shortest path length from each airport in the dataset to Turin airport (id = 1526). Which are the 10 airports that are farther from Turin, in terms of number of hops? For each of these airports, report its name, its city and country, and the shortest path length to Turin (i.e., number of hops).

In [117…
```
#list of landmark nodes
landmarks=["1526"]
results = g.shortestPaths(landmarks=landmarks)
```

In [118…
```
res = results.select("id","name","city","country","distances")
res.show(1)
res.count()
```

```
+----+------------------+---------+---------+-----------+
|  id|              name|     city|  country|  distances|
+----+------------------+---------+---------+-----------+
|6240|Birdsville Airport|Birdsville|Australia|[1526 -> 7]|
+----+------------------+---------+---------+-----------+
only showing top 1 row
```

Out[118…
```
7698
```

In [119…
```
def numberOfHops(dist_dict):
    if len(dist_dict)>0:
        v = dist_dict["1526"]
    else:
        v = 0
    return (int(v))
```

In [120…
```
spark.udf.register("NHops",lambda distances: numberOfHops(distances),"inte
```

Out[120…
```
<function __main__.<lambda>(distances)>
```

In [121…
```
num_hops = res.selectExpr("id","name","city","country","distances","NHops(
```

In [122…
```
num_hops.sort("n_hops",ascending = False).show(10)
```

```
+----+--------------------+------------+-------------+-----------+------+
|  id|                name|        city|      country|  distances|n_hops|
+----+--------------------+------------+-------------+-----------+------+
|5522|   Peawanuck Airport|   Peawanuck|       Canada|[1526 -> 8]|     8|
|5482|Attawapiskat Airport|Attawapiskat|       Canada|[1526 -> 7]|     7|
|  10|      Thule Air Base|       Thule|    Greenland|[1526 -> 7]|     7|
|8199|   Nightmute Airport|   Nightmute|United States|[1526 -> 7]|     7|
|6321|    Portland Airport|    Portland|    Australia|[1526 -> 7]|     7|
|6329|Thargomindah Airport|Thargomindah|    Australia|[1526 -> 7]|     7|
|6240|  Birdsville Airport|  Birdsville|    Australia|[1526 -> 7]|     7|
|5535|     Salluit Airport|     Salluit|       Canada|[1526 -> 7]|     7|
|6333|    Windorah Airport|    Windorah|    Australia|[1526 -> 6]|     6|
|5893|   Mota Lava Airport|       Ablow|      Vanuatu|[1526 -> 6]|     6|
+----+--------------------+------------+-------------+-----------+------+
only showing top 10 rows
```

4. Analyze and process the graph Task 4: Given Turin airport (id==1526) and Belo Horizonte airport (id = 2537), compute: from how many airports in the world you can reach Turin using less hops than to reach

Belo Horizonte from how many airports in the world you can reach Belo Horizonte using less hops than to reach Turin from how many airports in the world you can reach with the same number of hops Turin and Belo Horizonte

In [123…
```
#list of landmark nodes
landmarks=["1526","2537"]
results_2 = g.shortestPaths(landmarks=landmarks)
res_2 = results_2.select("id","name","city","country","distances")
#res_2.show(2)
```

In [124…
```
def getdictofdistances(distances_map):
    if len(distances_map)>0:
        v = distances_map["2537"]
    else:
        v = 0
    return (int(v))
```

In [125…
```
spark.udf.register("getHops", lambda distances: getdictofdistances(distance
```

Out[125… <function __main__.<lambda>(distances)>

In [126…
```
turin_belo_n_hops = res_2.selectExpr("id","name","city","country","distance
.withColumnRenamed("getHops(distances)","n_hops_belo").withColumnRenamed("I
```

In [127…
```
#turin_belo_n_hops.show(5)
```

In [128…
```
turin_less_belo = turin_belo_n_hops.filter("n_hops_turin<n_hops_belo")
#turin_less_belo.show(5)
turin_less_belo.count()
```

Out[128… 1278

In [129…
```
belo_less_turin = turin_belo_n_hops.filter("n_hops_turin > n_hops_belo")
#belo_less_turin.show(5)
belo_less_turin.count()
```

Out[129… 281

In [130…
```
belo_same_turin = turin_belo_n_hops.filter("n_hops_turin = n_hops_belo")
#belo_same_turin.show(5)
belo_same_turin.count()
```

Out[130… 1608

4. Analyze and process the graph Task 5: How many connected components of at least two airports are there in the graph? Report the number of connected components and their sizes. Hint: First, drop the isolated vertices.

In [131…
```
sc.setCheckpointDir("tmp_ckpts")
gg = g.dropIsolatedVertices()
connected = gg.connectedComponents()
```

In [132…
```
cc = connected.select("id", "component").filter("component>2").orderBy("cor
```

In [133…
```
nComp=connected.select("component").distinct().count()
print("Number of connected components: ", nComp)
```

```
Number of connected components:  14
```

4. Analyze and process the graph Task 6: Consider only the subgraph of the flights that are performed either by AirDolomiti (icao = DLA,iata = EN) or by Sky Airline (icao = SKU). Can you plot this subgraph? Report the name of the cities (of the airports) in the graph. Hint: use Graphvix

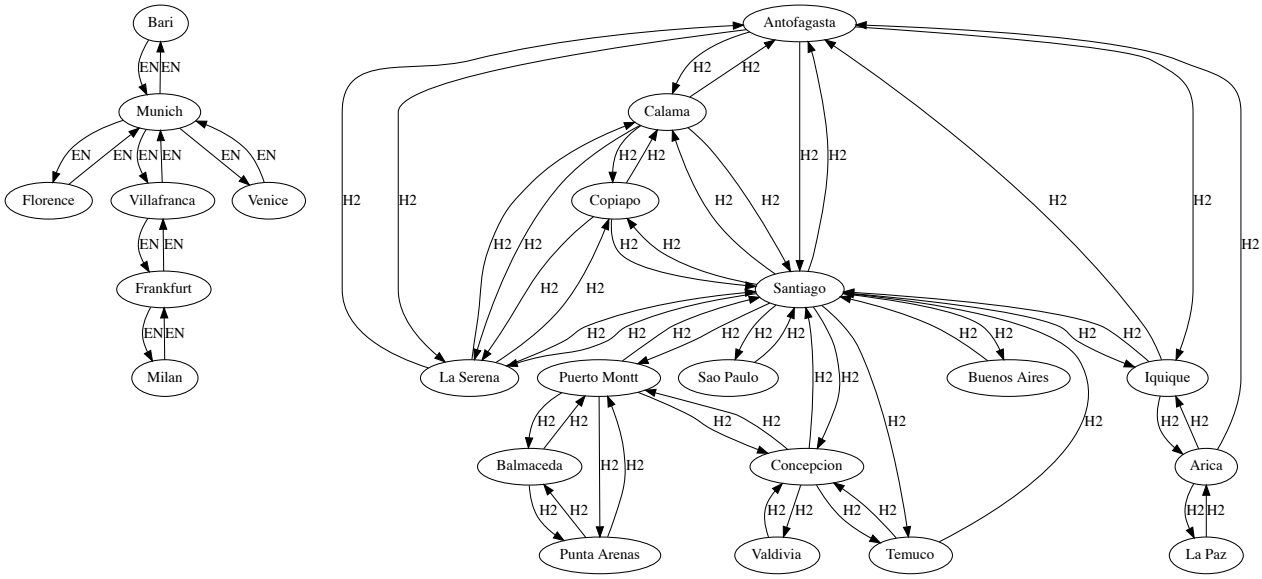
In [134…
```
subg = g.filterEdges("airline_iata == 'EN' or airline_iata =='H2'").dropIs
print(g.vertices.count(), g.edges.count())
print(subg.vertices.count(), subg.edges.count())
```

```
7698 67663
23 61
```

In [135…
```
from graphviz import Digraph
def vizGraph(edge_list,node_list):
    Gplot=Digraph()
    edges=edge_list.collect()
    nodes=node_list.collect()
    for row in edges:
        Gplot.edge(row['src'],row['dst'],label=row['airline_iata'])
    for row in nodes:
        Gplot.node(row['id'],label=row['city'])
    return Gplot
Gplot=vizGraph(subg.edges,subg.vertices)
```

In [136…
```
Gplot
```

Out[136…



In [ ]:

In [ ]: