INDIAN INSTITUTE OF TECHNOLOGY PATNA

NPTEL

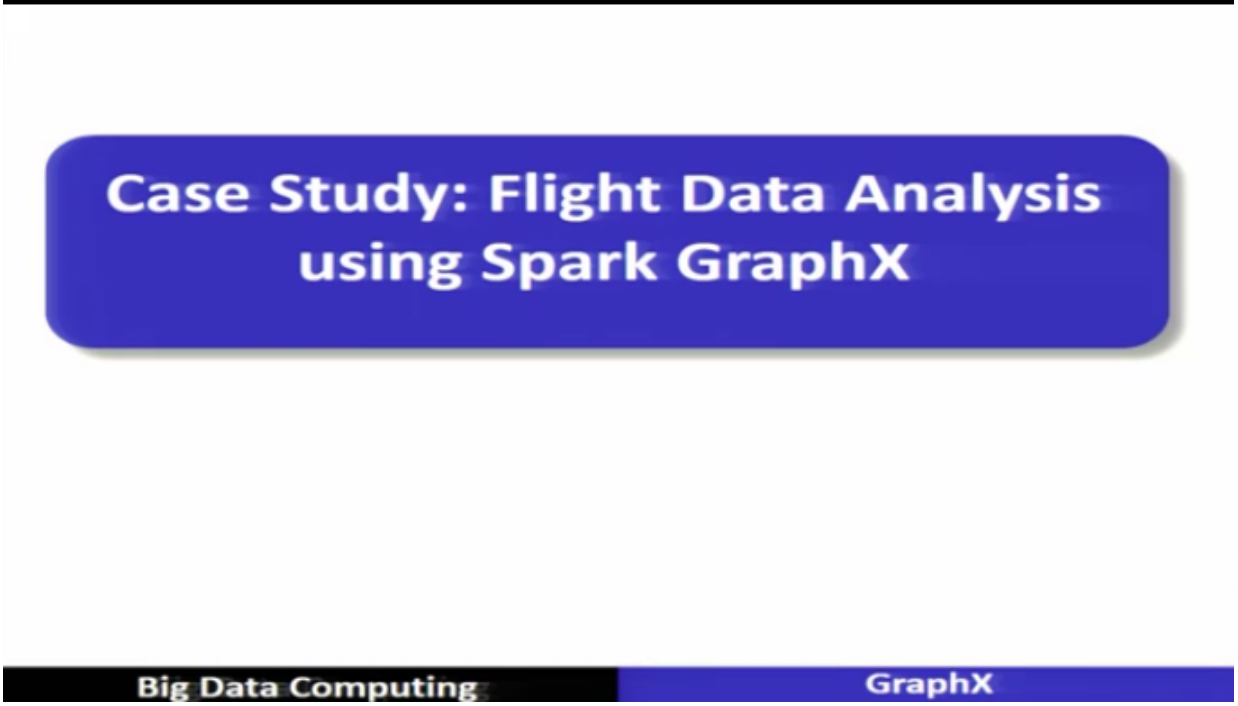
NATIONAL PROGRAMME ON TECHNOLOGY ENHANCED LEARNING

COURSE TITLE
BIG DATA COMPUTING

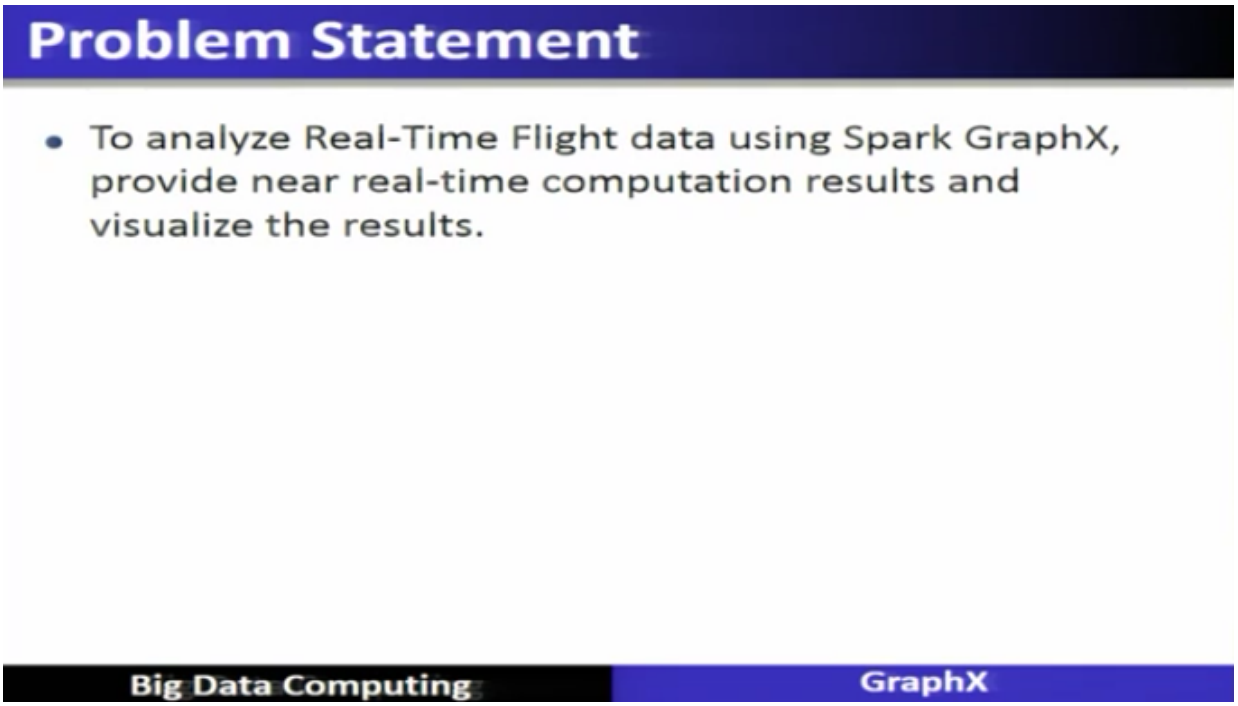
LECTURE-34
CASE STUDY: FLIGHT DATA ANALYSIS USING SPARK GRAPH-X

BY
DR. RAJIV MISRA
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY PATNA

(Refer Slide Time: 00:16)

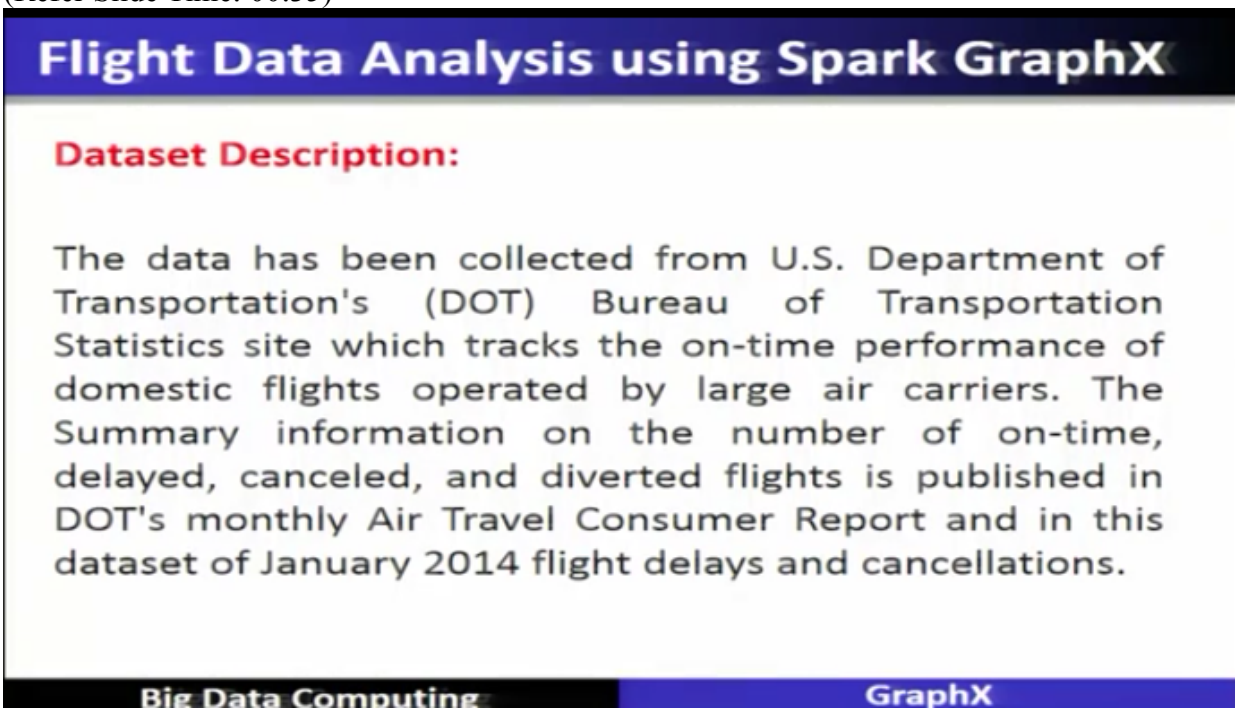


Cast Study, Flight Data Analysis using Spark GraphX.
(Refer Slide Time: 00:20)



The problem statement, to analyze the Real-Time Flight data using Spark GraphX, and to provide near real-time computation results and visualize the results.

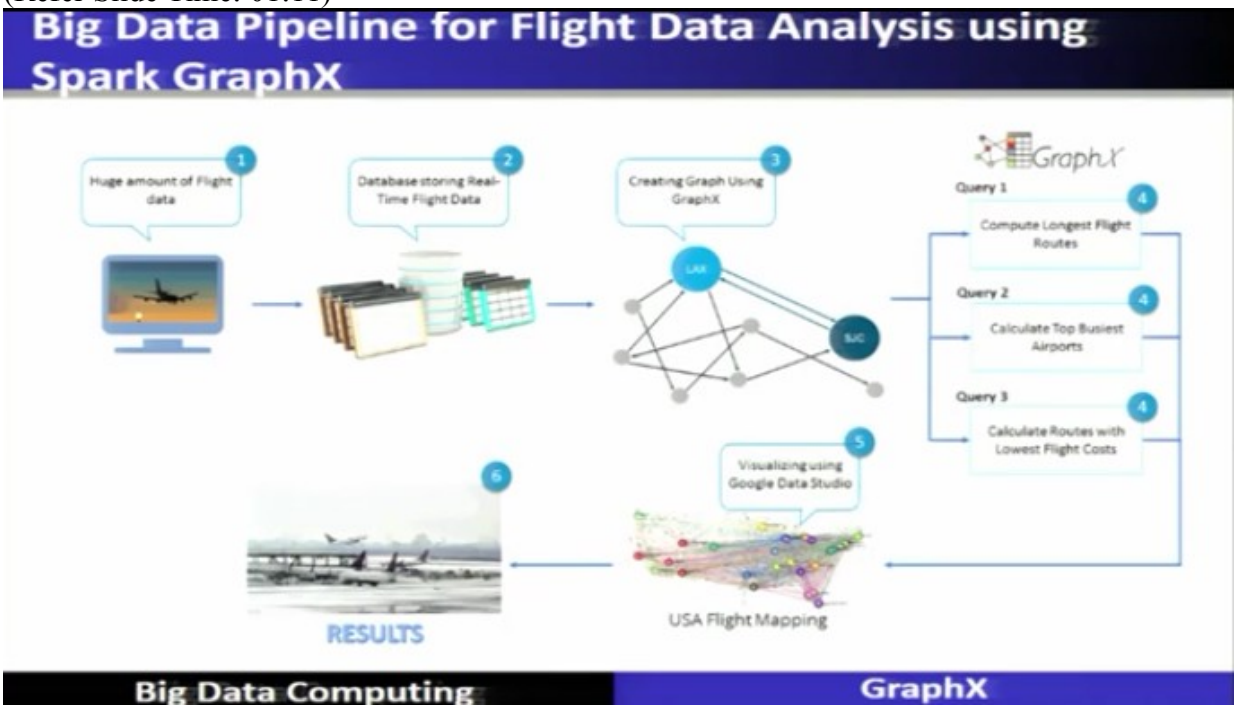
(Refer Slide Time: 00:35)



So for this let us see the dataset description, so data has been collected from US Department of Transportation DOT, Bureau of Transportation Statistics site which tracks on-time performance of domestic flights operated by large carriers. The summary information on the number of on-time delayed, canceled, and diverted flights is published in DOT's monthly Air Travel

Consumer Report and in this dataset of January 2014 flight delays and cancellations are been considered.

(Refer Slide Time: 01:11)

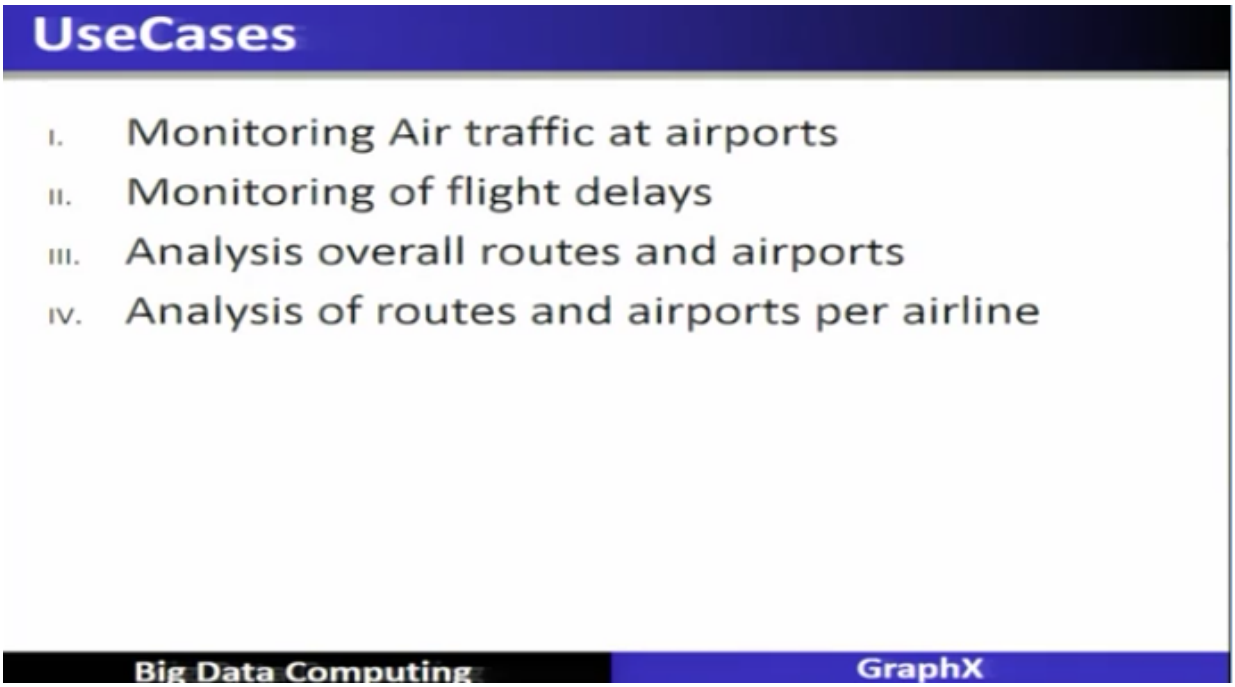


So we are going to analyze this dataset that is a called a flight dataset using the pipeline, that is called big data pipeline for Flight Data Analysis using Spark GraphX, so the first stage, first step in the pipeline is we have collected the dataset, so we have obtained the dataset which is a huge amount of flight data information.

Now this particular data base storing real-time flight data is being applied or being used into the data base, so after getting the data or a car pass or a big data set, we converted into the form of a graph, so that means the data is converted into the form of a table, and from table which will be converted into the form of a graph, the second pipeline, so creating graph using GraphX we will create a graph for the entire data.

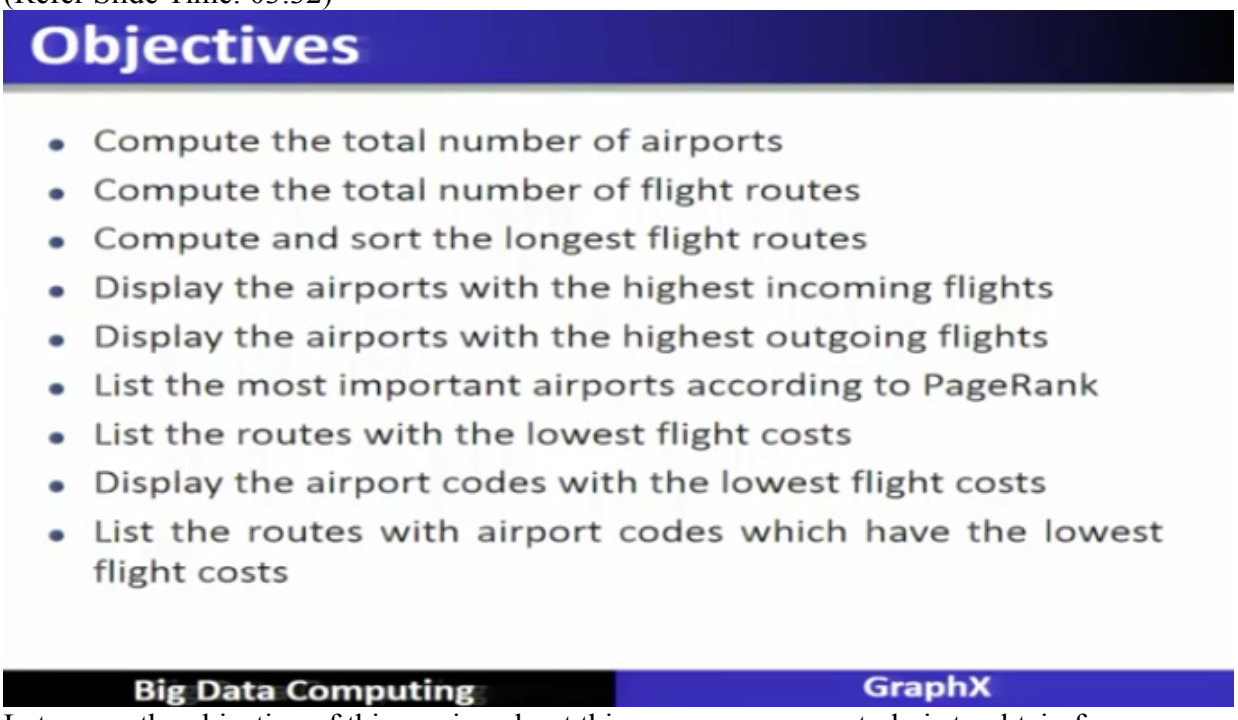
So after creating the graph we can apply various queries based on the graph algorithms, so the queries are we have to compute the longest flights, longest flight routes or we can compute the top busiest airports or we can compute the routes with the lowest flight costs, so after doing all these queries we can also do the visualization of a flight mapping on the maps, and then we will use this particular result for this.

Let us see how this steps which is called the big data pipeline is use in doing this analyzing this flight data information.
(Refer Slide Time: 03:11)



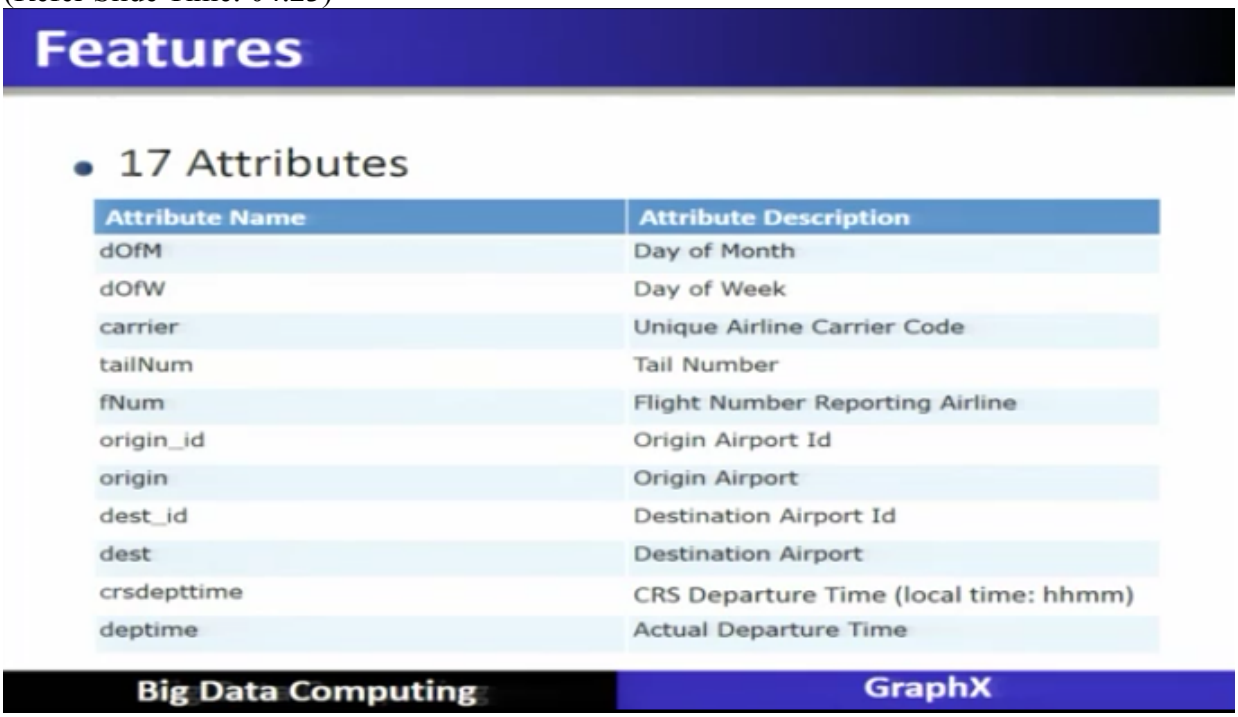
So this particular study will have the use cases such as monitoring air traffic at airports, monitoring of flight delays, analysis of overall routes and airports, analysis of routes and airports per airline and so on.

(Refer Slide Time: 03:32)



Let us see the objective of this session about this use case or case study is to obtain for following objectives, first is that how to compute the total number of airports in this dataset, how to find out the total number of flight routes, how to compute and sort the longest flight routes, how to display the airports with the highest incoming flights, how to display the airports with the highest outgoing flights, how to list the most important airports according to the

PageRank, and list the routes with the lowest flight costs, display the airport codes and the lowest flight costs, list the routes with the airport codes which have the lowest flight costs, let us see how these queries or how these objectives are to be fulfilled using the given dataset.
(Refer Slide Time: 04:23)



This dataset, if we go in more detail and see the features, there are 17 different attributes are available in this particular dataset, for example DOFM that is day of the month, DOFW means day of the week, carrier unique airline carrier code, tail number is the tail number, and F number is the flight number reporting airlines, and origin ID is the origin airport ID, origin and origin airport, destination ID is the destination airport ID, destination is destination airport, CRS departure time, departure time is actual departure time, and departure delay minutes difference in the minutes between the schedule and actual departure and earlier departure is set to 0.

(Refer Slide Time: 05:04)

Features

Attribute Name	Attribute Description
depdelaymins	Difference in minutes between scheduled and actual departure time. Early departures set to 0.
crsarrtime	CRS Arrival Time (local time: hhmm)
arrtime	Actual Arrival Time (local time: hhmm)
arrdelaymins	Difference in minutes between scheduled and actual arrival time. Early arrivals set to 0.
crselapsedtime	CRS Elapsed Time of Flight, in Minutes
dist	Distance between airports (miles)

Big Data Computing GraphX

Similarly there are arrival times and actual arrival time difference between, in the minutes between the scheduled and actual arrival time that is arrival delay in the minutes, and similarly the elapse time of the flight and the distance between the airport.

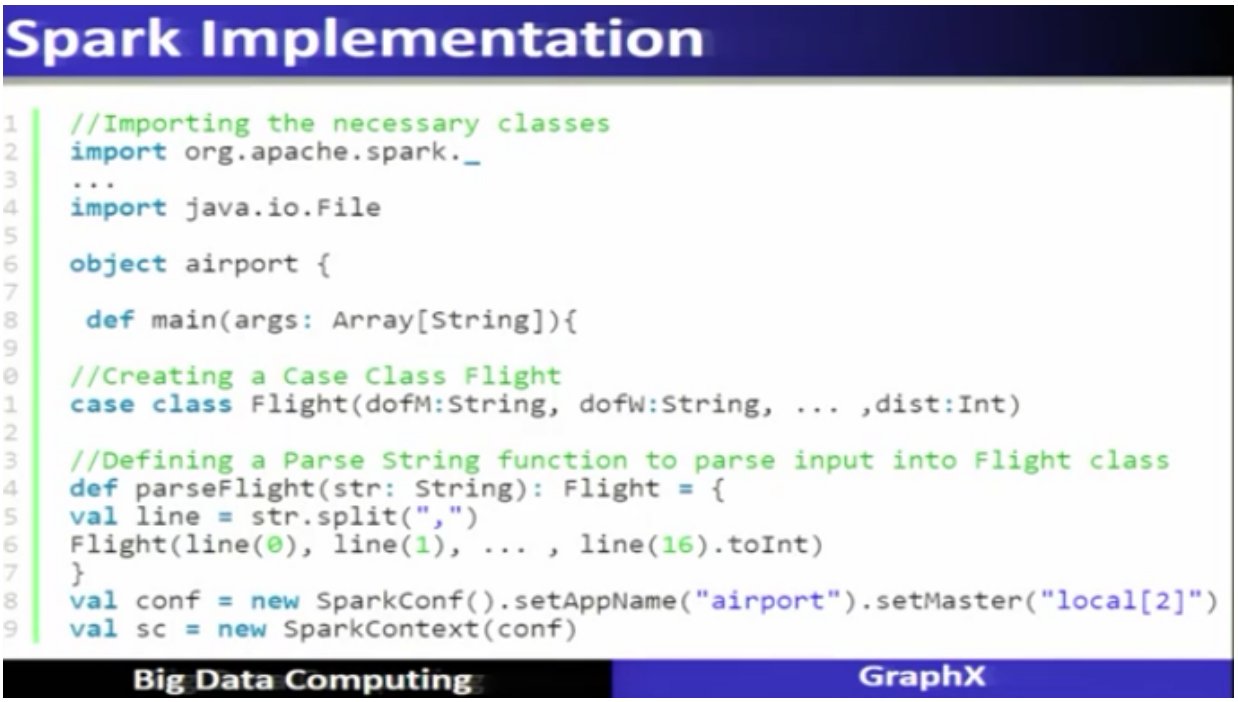
(Refer Slide Time: 05:33)

Sample Dataset

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	dOfM	dOfW	carrier	tailNum	flNum	origin_id	origin	dest_id	dest	crsdeptime	deptime	depdelaymins	crsarrtime	arrtime	arrdelaymins	crselapsedtime	dist
2	1	3	AA	N338AA	1	12478	JFK	12892	LAX	900	914	14	1225	1238	13	385	2475
3	2	4	AA	N338AA	1	12478	JFK	12892	LAX	900	857	0	1225	1226	1	385	2475
4	4	6	AA	N327AA	1	12478	JFK	12892	LAX	900	1005	65	1225	1324	59	385	2475
5	5	7	AA	N323AA	1	12478	JFK	12892	LAX	900	1050	110	1225	1415	110	385	2475
6	6	1	AA	N319AA	1	12478	JFK	12892	LAX	900	917	17	1225	1217	0	385	2475
7	7	2	AA	N328AA	1	12478	JFK	12892	LAX	900	910	10	1225	1212	0	385	2475
8	8	3	AA	N323AA	1	12478	JFK	12892	LAX	900	923	23	1225	1215	0	385	2475
9	9	4	AA	N339AA	1	12478	JFK	12892	LAX	900	859	0	1225	1204	0	385	2475
10	10	5	AA	N319AA	1	12478	JFK	12892	LAX	900	929	29	1225	1245	20	385	2475

Big Data Computing GraphX

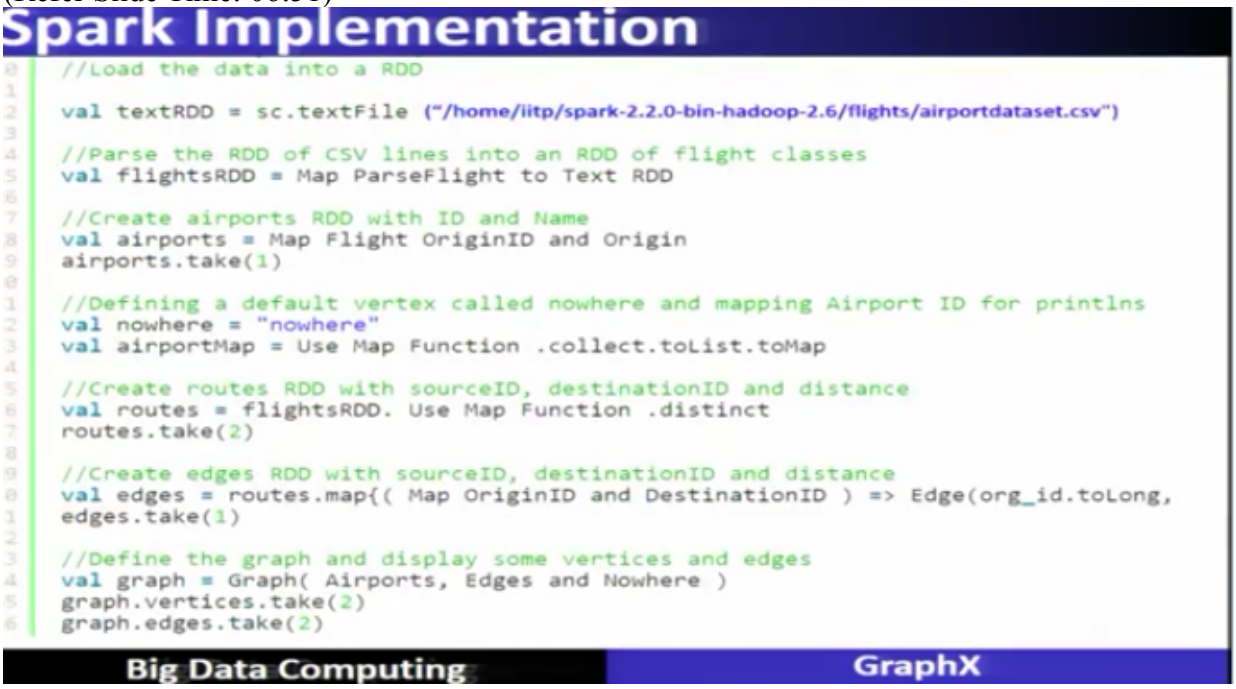
So this is the sample dataset with all the 17 features shown over here for our analysis use case,
(Refer Slide Time: 05:40)



so here we can see here the part of the spark code which implements this flight data analysis using GraphX, so let us understand some of the code before we go into more detail of it.

So that means first we have to create a class which is called a flight class with all these details which are provided as 17 different features, then we can pass the string function to pass the input into the flight class data, so for that these functions are there, and then we will create the application name and we have to create the spark context, so this will create the spark context.

(Refer Slide Time: 06:51)

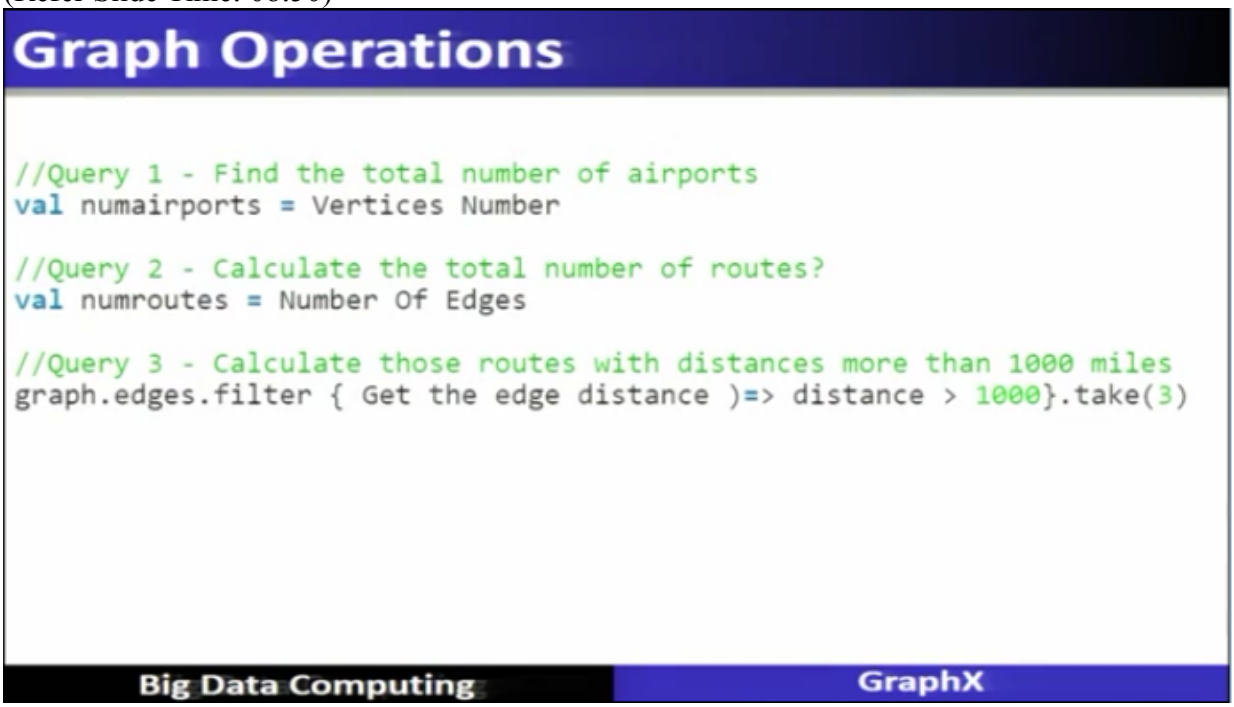


Now the next step is to read the file that is the text file, once it is read then it will become and RDD, so the file is read and after reading it becomes an RDD into the spark system, and now we have to pass the RDD into an RDD flight class using this map function, and it will create the airport RDD's with the ID and names using this particular function.

After defining default vertex called nowhere and mapping airport ID's for printlines, so here all these details are listed over here, now we have to create the routes RDD's with the source ID and destination and the distance using the map function, and create the edge RDD's with the source ID and the distance and so on.

So as RDD is the vertex IDD's routes RDD's are now created, now we can define the graph as the airports as the vertices, and the routes and edges, so routes will be the edges, and the airports will be the vertices, the vertices, edges and nowhere, so after taking them now we will convert it into an edge RDD and vertex RDD.

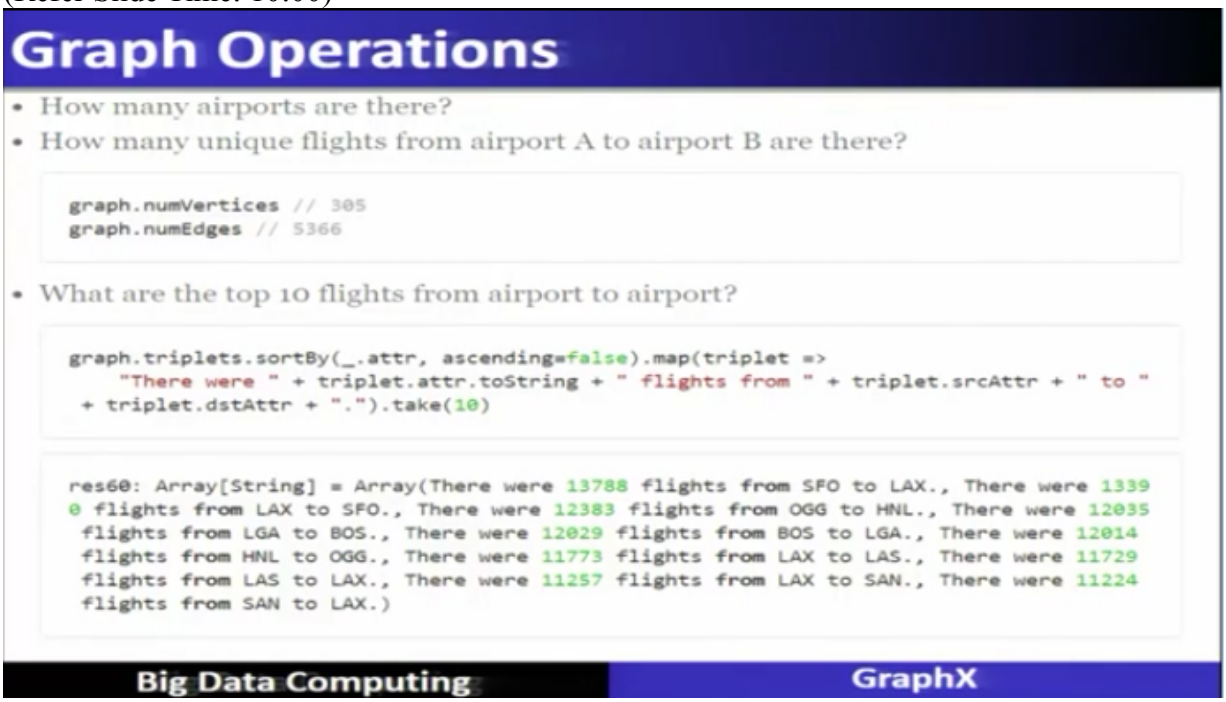
(Refer Slide Time: 08:50)



So using vertices and a graphs we can now, we have computed, we have form the graph of that particular dataset, now we have to perform various queries, so the query 1 is to find out how many number of airports are there in that, so just we have to calculate, we have to find out the number of vertices in the graph. If you want to find out how many number of routes are there, we have to calculate how many different edges are there in the graph.

Now if you want to find out, calculate those routes with the distance more than 1000 miles, so easily we can apply the filter function of this GraphX on this particular graph, so after applying the filter function where in the distance is greater than 1000, it will generate graph that is called as subgraph. Now this whole function will not delete the portions the graph, but it will use tombstone and will generate the subgraph out of the main graph.

(Refer Slide Time: 10:00)



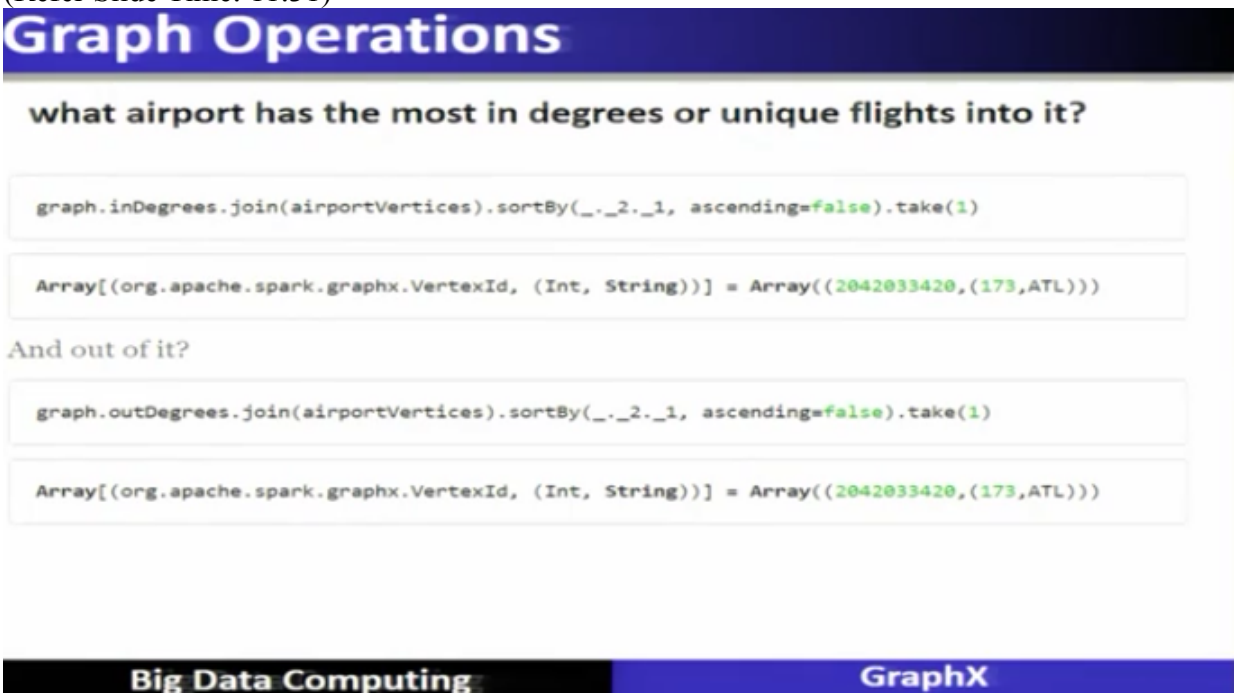
Similarly if you want to find out how many airports are there, so airports are represented as the number of vertices, so we have to run this particular function that is graph.number of vertices so we can find out that number of vertices are 304, 305.

Similarly how many unique flights from airport A to B are there, so we have to find out the number of edges here and if you want to find out the top 10 flights from airport to airport, so we can find out using the graph triplets and we can sort the flights, top 10 flights and now we can take these 10 data out of this sorted top 10 flights, we can use it, so we can see here that there are so many number of flights from source destination and by taking the triplets and doing the sort and we can easily get this information out of this particular graph.

(Refer Slide Time: 11:20)

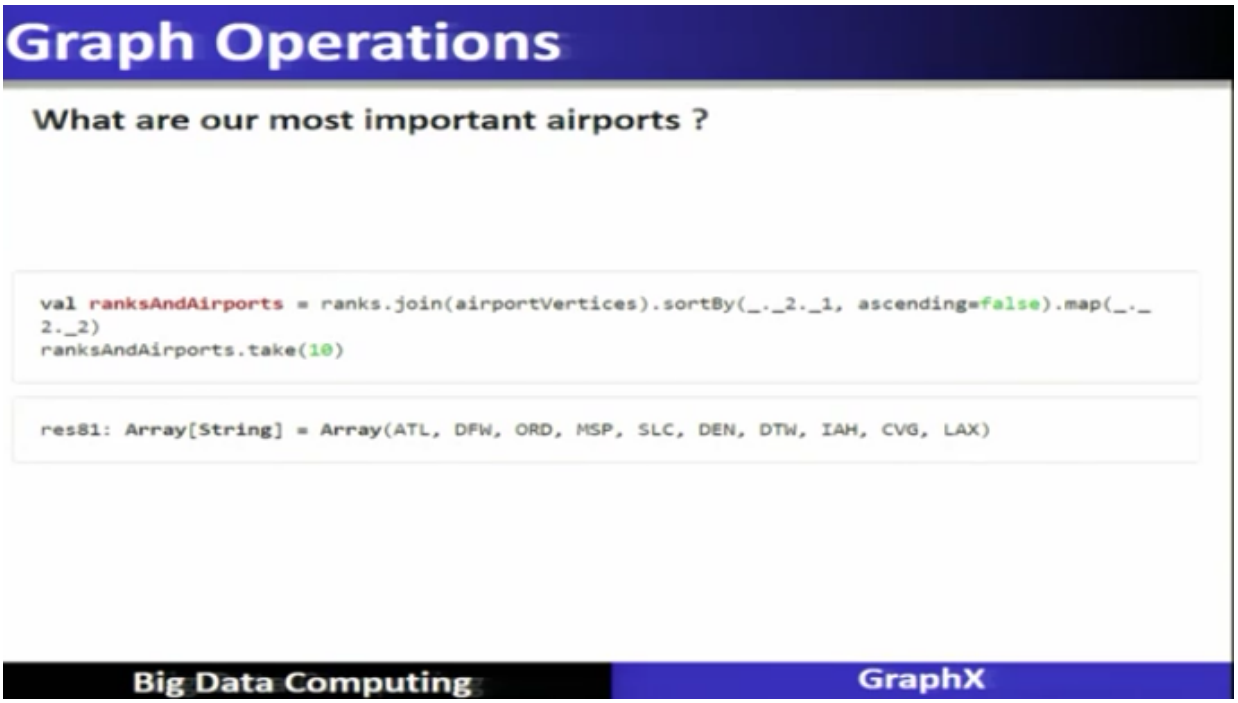
So what are the lowest 10 flights from airport to airport using some triplets and by sorting it in the descending order we can obtain this data also.

(Refer Slide Time: 11:31)



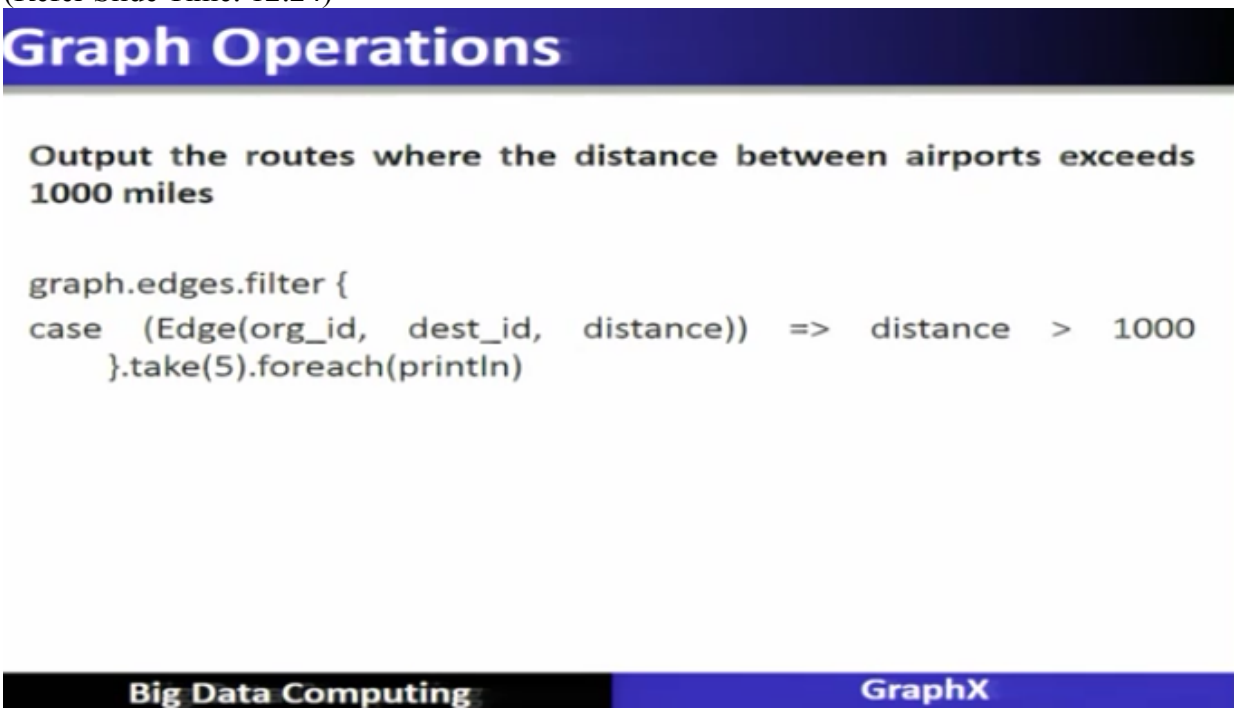
Now if you want to find out which, what airport has the most in degree and or unique flights into it, then using in degrees and the joint airport vertices we can obtain this amount of information, and out of it what are the flights which are going out of the airport, most of the degrees so that also we can find out using out degree, and after applying the joint operation.

(Refer Slide Time: 12:07)



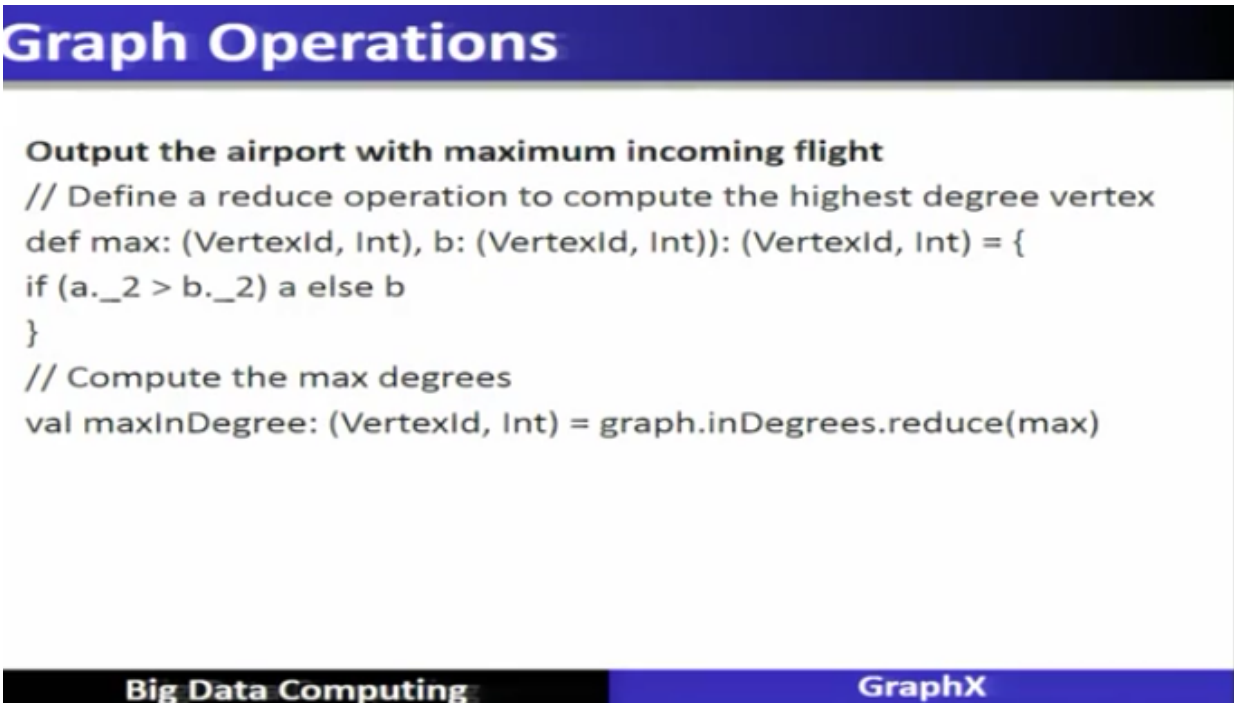
What are the most important airports? So for that using PageRank what we can ranks and, not using PageRank, but ranks and airport means it will sort and find out the airports,

(Refer Slide Time: 12:24)



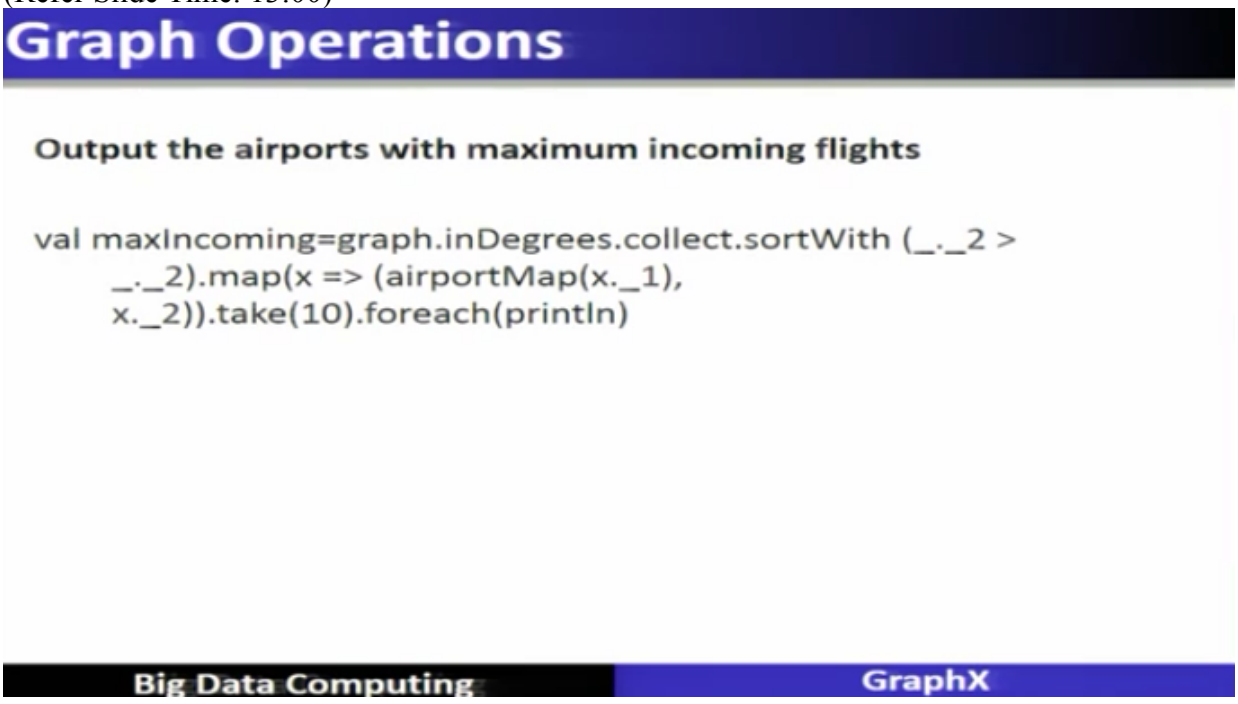
so graph operations output the routes where the distance between the airport exceeds 1000 that we have seen, it will just apply the filter and get this output of the routes where the distance between the airport exceeds 1000 miles.

(Refer Slide Time: 12:42)



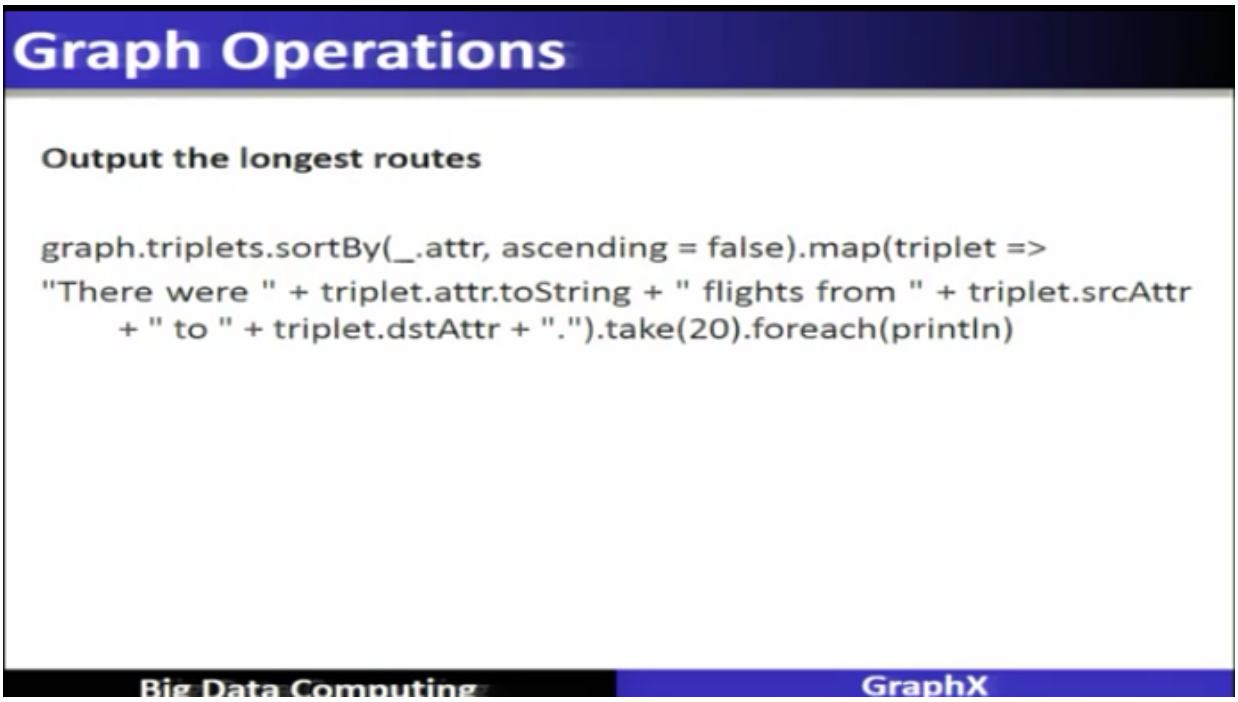
Similarly if you want to output the airport with the maximum incoming flight, so here in degree we can find out with the maximum of it, and this will give the maximum incoming flight.

(Refer Slide Time: 13:00)



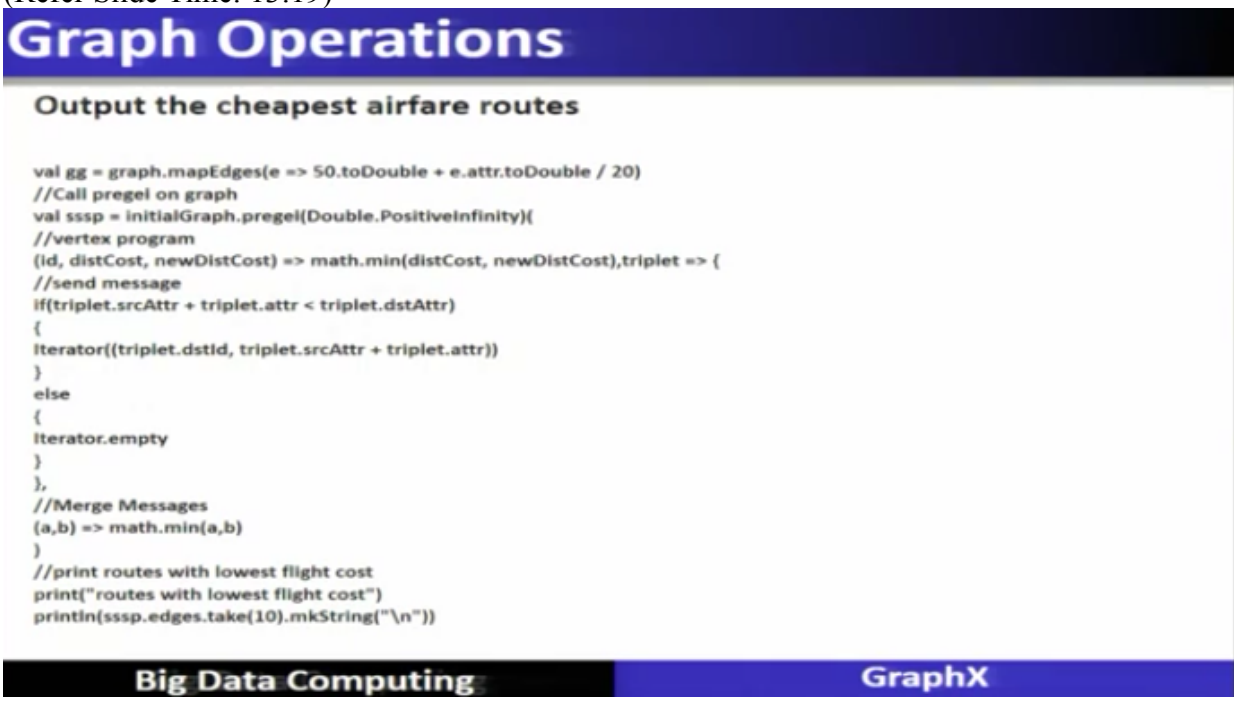
Similarly maximum incoming flights we can get using this particular piece of code.

(Refer Slide Time: 13:08)



Similarly we have to output the longest routes by doing this kind of sorting, and applying on the graph triplets.

(Refer Slide Time: 13:19)

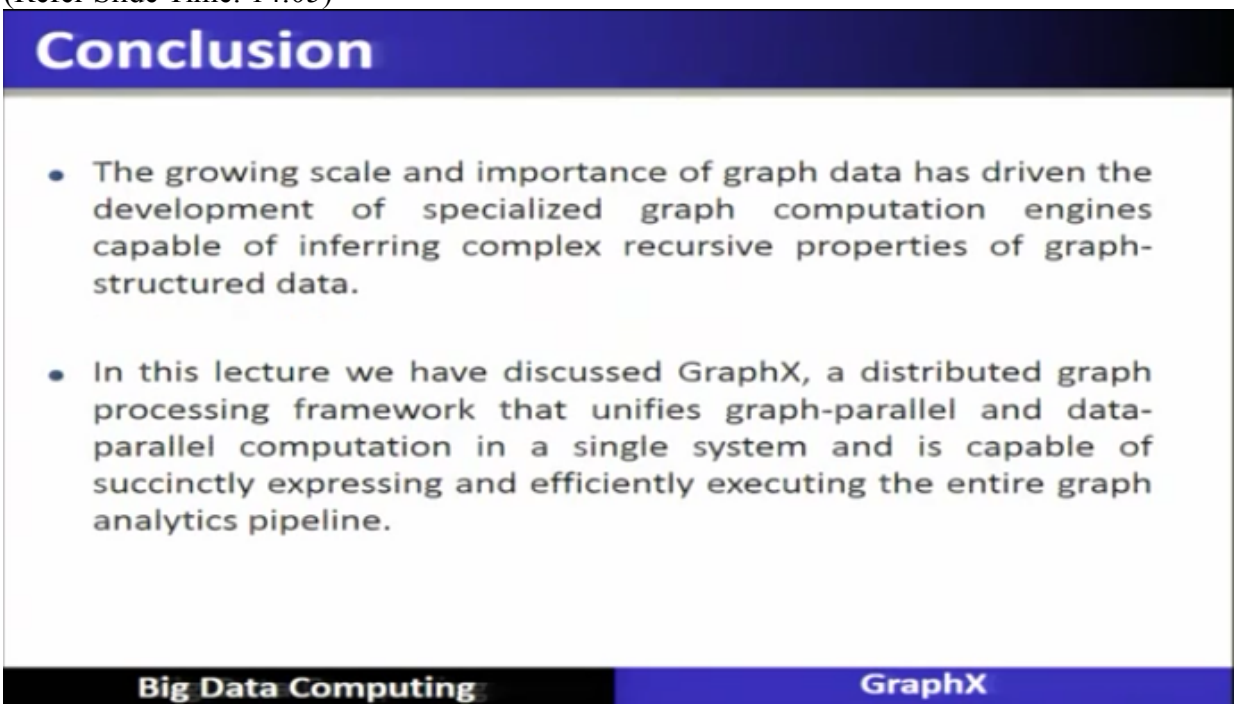


So if you want to find out the cheapest airline route, then this is the code and using this particular code we can obtain this.

(Refer Slide Time: 13:29)

Now we can find out the most influential airport using the PageRank, so we can apply the PageRank algorithm over the vertices, and we can then join the airports and then take a temp, and top 10 and then it will become the most influential airport using PageRank, so output the most influential airport from the most influential to the latest, so further more we can sort and we can obtain this.

(Refer Slide Time: 14:05)



So conclusion, the growing scale and importance of graph data has driven the development of specialized graph computation engines, capable of inferring complex recursive properties of graph-structured data.

In this lecture we have discussed GraphX, a distributed processing framework that unifies the graph-parallel and data-parallel computation in a single system and is capable of succinctly expressing and efficiently executing the entire graph data, graph analytics pipeline. Thank you.

Acknowledgement

Ministry of Human Resource & Development

Prof. Satyaki Roy

Co-coordinator, IIT Kanpur

NPTEL Team

Sanjay Pal
Bharat Lal
Ashish Singh
Badal Pradhan
Tapobrata Das
K. K. Mishra
Ashutosh Gairola
Puneet Kumar Bajpai
Bhadro Rao
Shikha Gupta
Aradhana Singh Rajawat
Sweta
Nipa Sikdar
Anupam Mishra
Ram Chandra
Manoj Shrivastava
Dilip Tripathi
Padam Shukla
Sharwan K Verma
Sanjay Mishra
Shubham Rawat
Santosh Nayak
Praduyman Singh Chauhan
Mahendra Singh Rawat
Tushar Srivastava
Uzair Siddiqui
Lalty Dutta

Murali Krishnan
Ganesh Rana
Ajay Kanaujia
Ashwani Srivastava
M.L. Benerjee

an IIT Kanpur Production