import org.apache.spark.\_

import org.apache.spark.rdd.RDD

import org.apache.spark.graphx.\_

val vertices = Array((1L,("A")),(2L,("B")),(3L,("C")))

val vRDD = sc.parallelize(vertices)

vRDD.take(1)

vRDD.take(2)

val edges = Array(Edge(1L,2L,1800),Edge(2L,3L,800),Edge(3L,1L,1400))

val eRDD = sc.parallelize(edges)

eRDD.take(2) val nowhere = "nowhere"

val graph = Graph(vRDD,eRDD,nowhere)

#To check number of Airports

val numairports = graph.numVertices

#To check routes

val numairports = graph.numEdges

#Route having distance > 1000

(graph.edges.filter{case Edge(src,dst,prop)=>prop>1000}.collect.foreach(println))

#Triplet Information

graph.triplets.take(3).foreach(println)

#Indegree

val i = graph.inDegrees i.collect()

#Outdegrees

val o = graph.outDegrees o.collect()

#Total Degree

val t = graph.degrees t.collect()