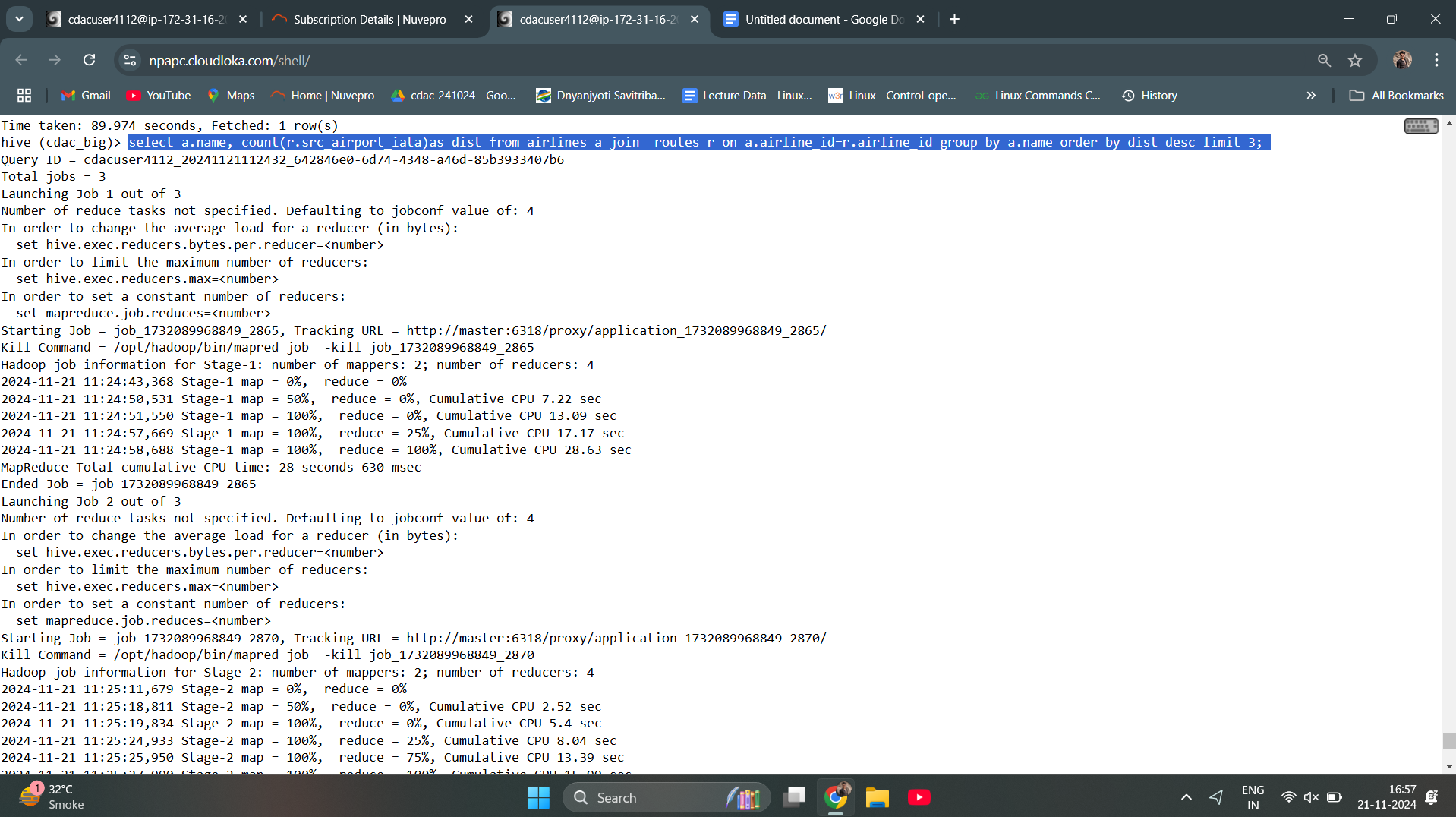
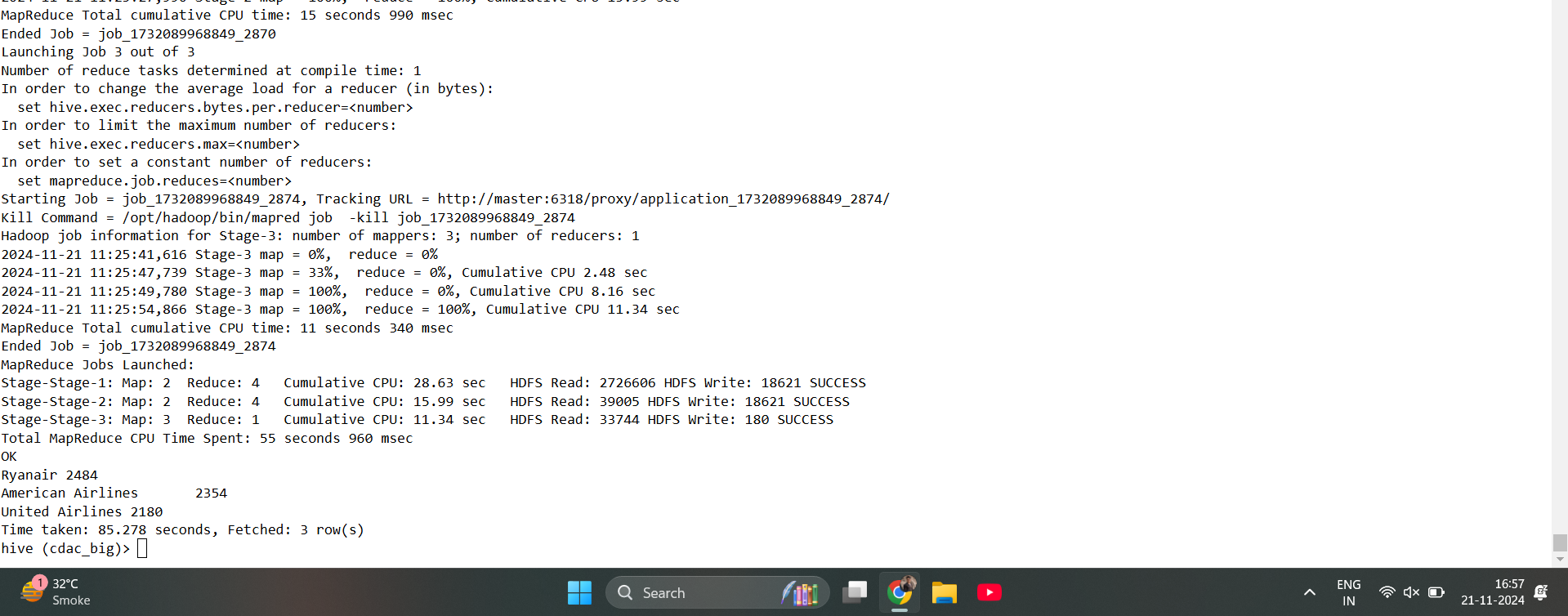
HIVE

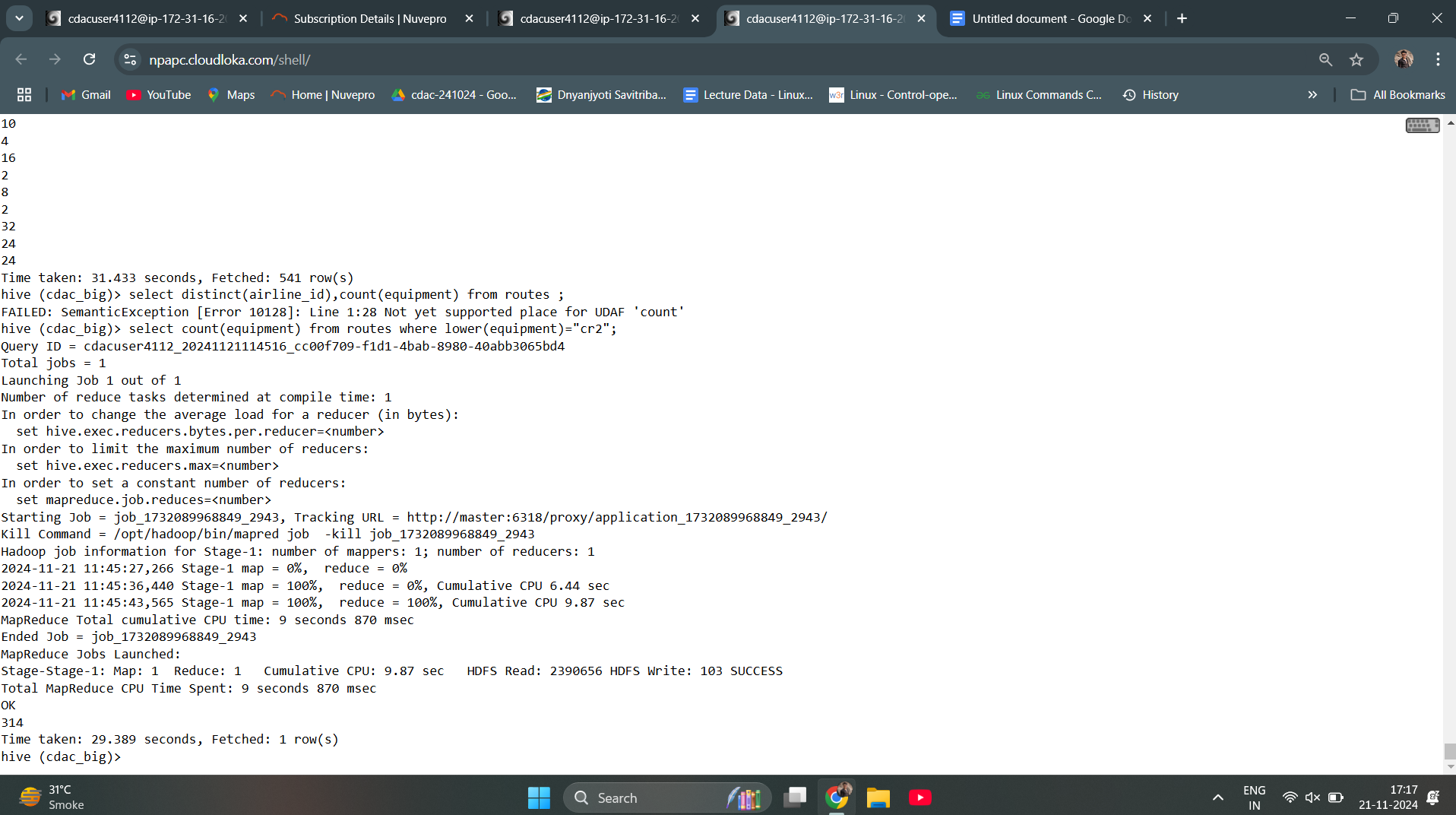
Question 1)

1. select a.name from airport a join routes r on a.iata =r.src\_airport\_iata where r.src\_airport\_iata is not null and r.dest\_airport\_iata is null limit 10;
2. select a.name, count(r.src\_airport\_iata)as dist from airlines a join routes r on a.airline\_id=r.airline\_id group by a.name order by dist desc limit 3;





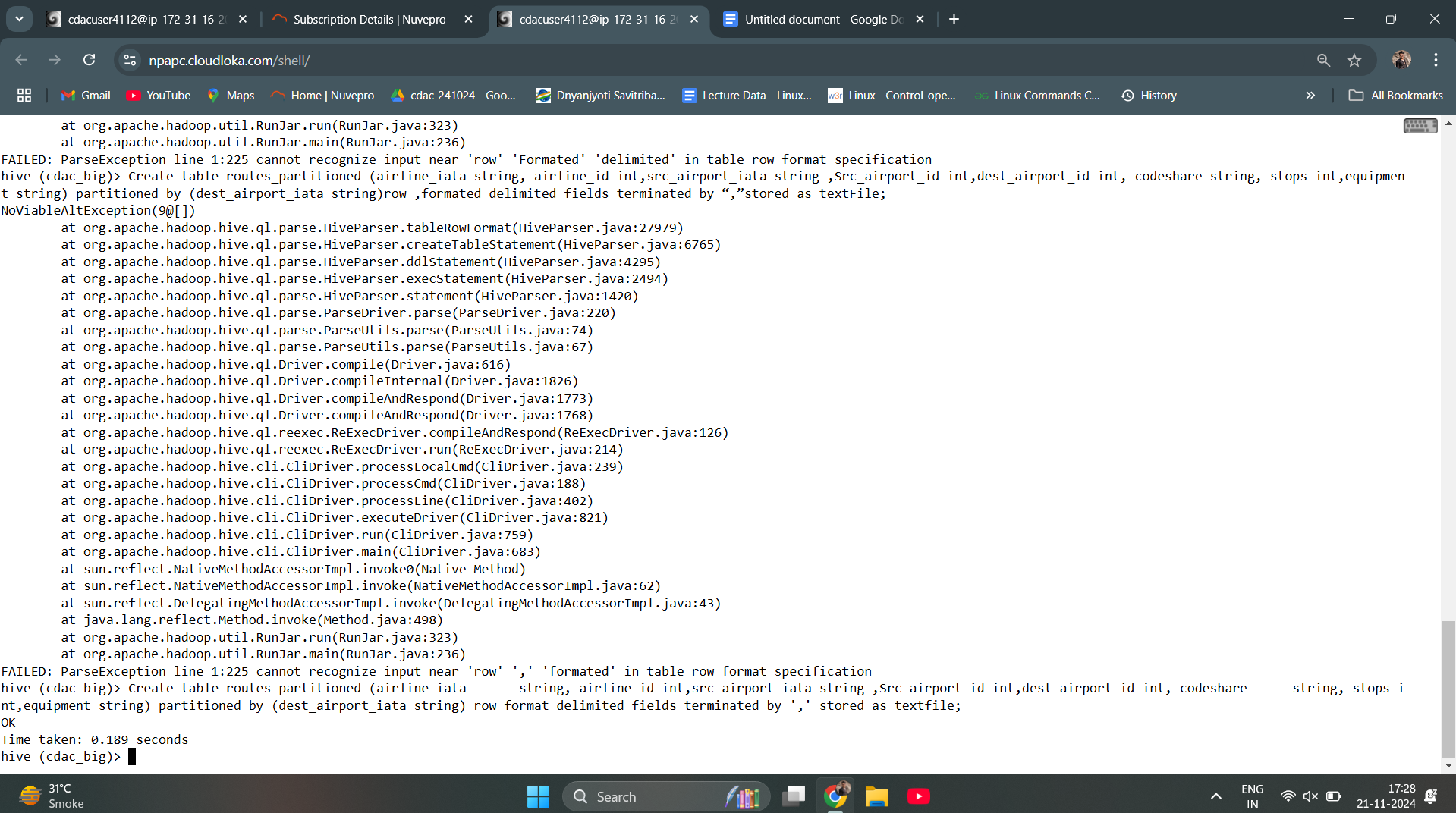
1. select count(equipment) from routes where lower(equipment)="cr2";

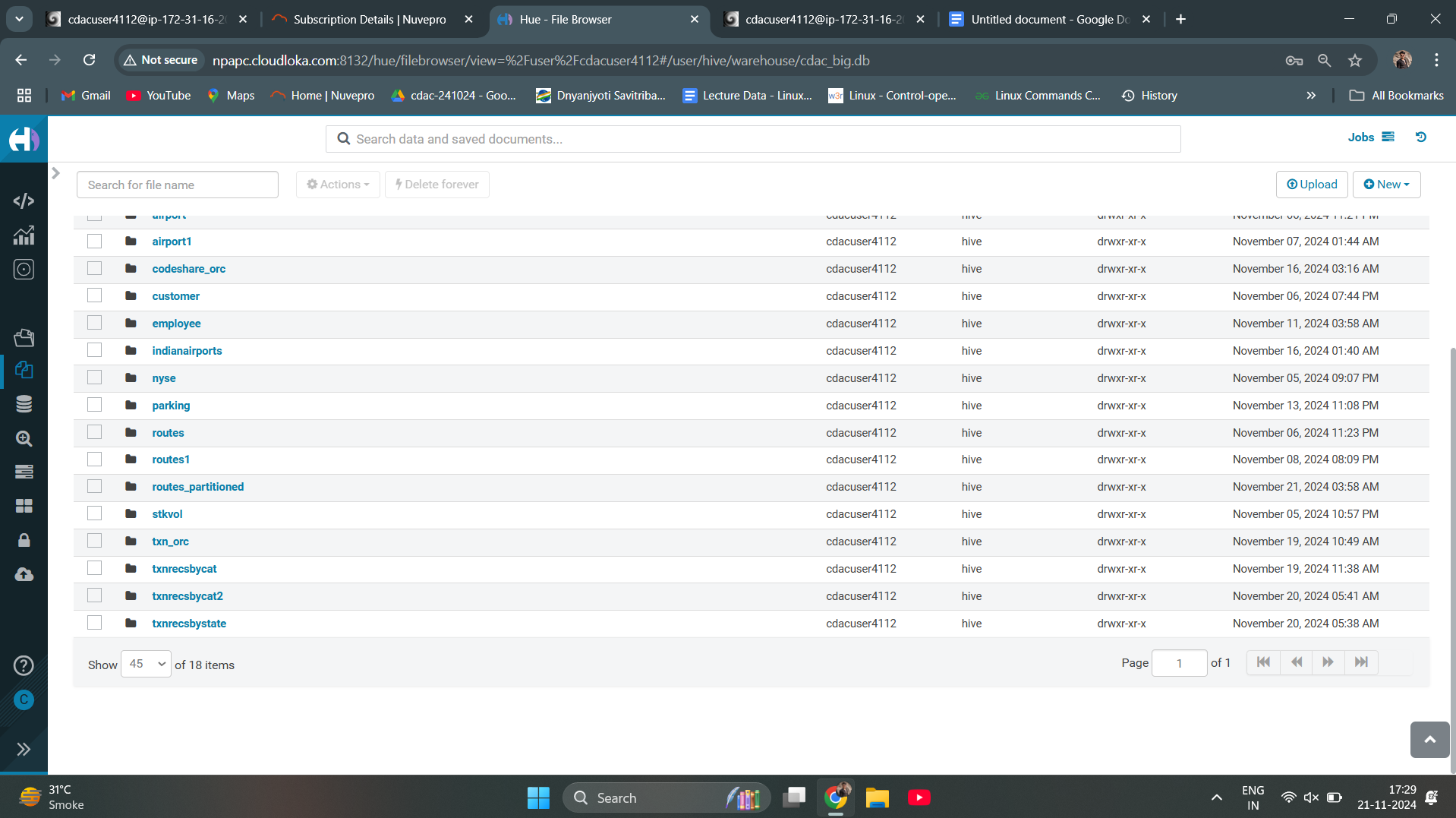


Question 2)

1. Create table routes\_partitioned (airline\_iata string, airline\_id int,src\_airport\_iata string

,Src\_airport\_id int,dest\_airport\_id int, codeshare string, stops int,equipment string) partitioned by (dest\_airport\_iata string) row format delimited fields terminated by ',' stored as textfile;





1. Insert overwrite table routes\_partitioned partition(r.dest\_airport\_iata) select r.airline\_iata string, r.airline\_id int,r.src\_airport\_iata string,r.Src\_airport\_id int,r.dest\_airport\_id int, r.codeshare string, r.stops int,r.equipment string from txnrecords distribute by r.dest\_airport\_iata subtring(r.ORD);

Insert overwrite table routes\_partitioned partition(dest\_airport\_iata) select r.airline\_iata string, r.airline\_id int,r.src\_airport\_iata string,r.Src\_airport\_id int,r.des

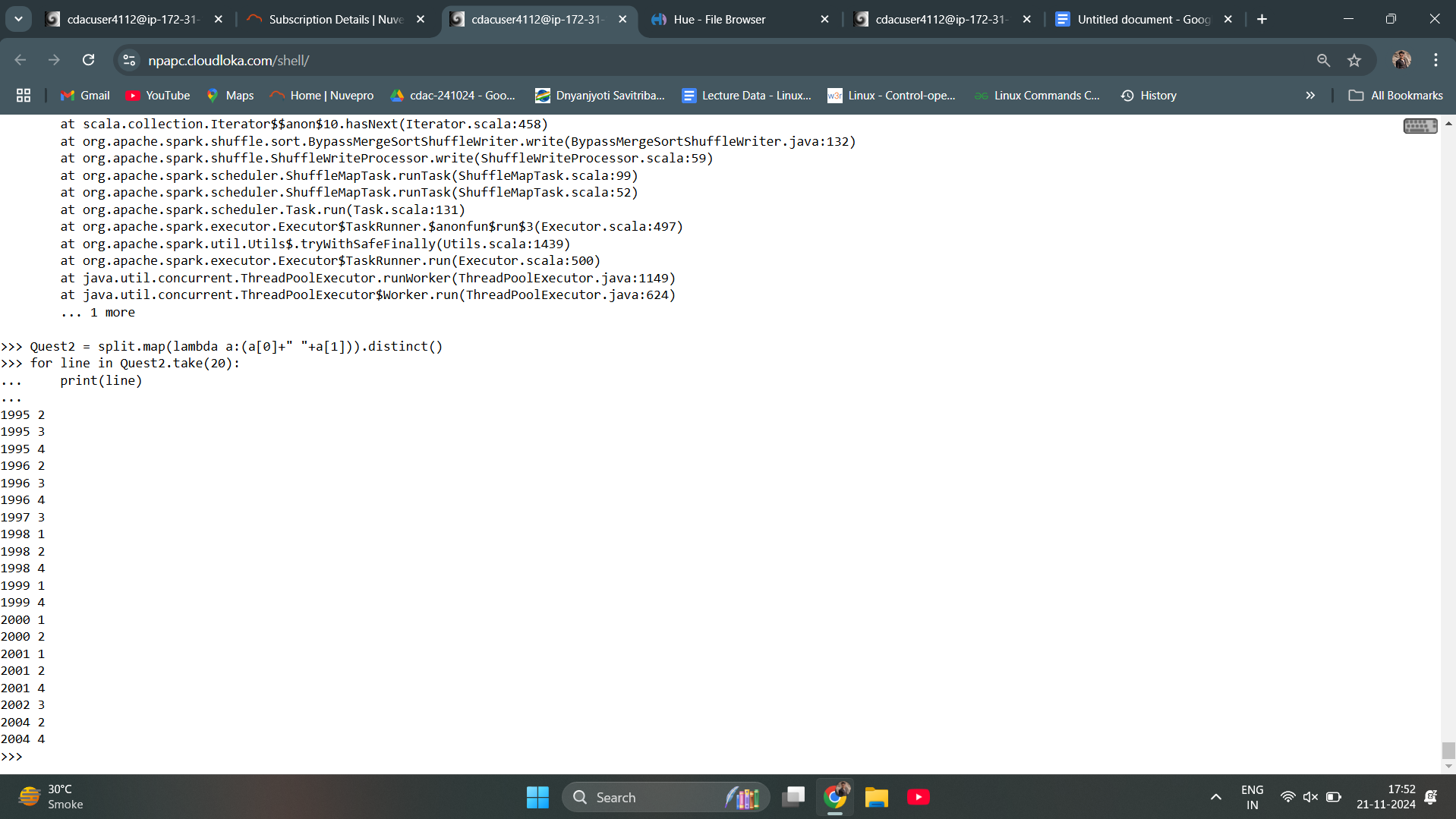
t\_airport\_id int, r.codeshare string, r.stops int,r.equipment string from txnrecords r distribute by r.dest\_airport\_iata subtring(r.ORD);

1. Select \* from routes\_partitioned where lower(r.dest\_airport\_iata)=”ord”;

SPARK

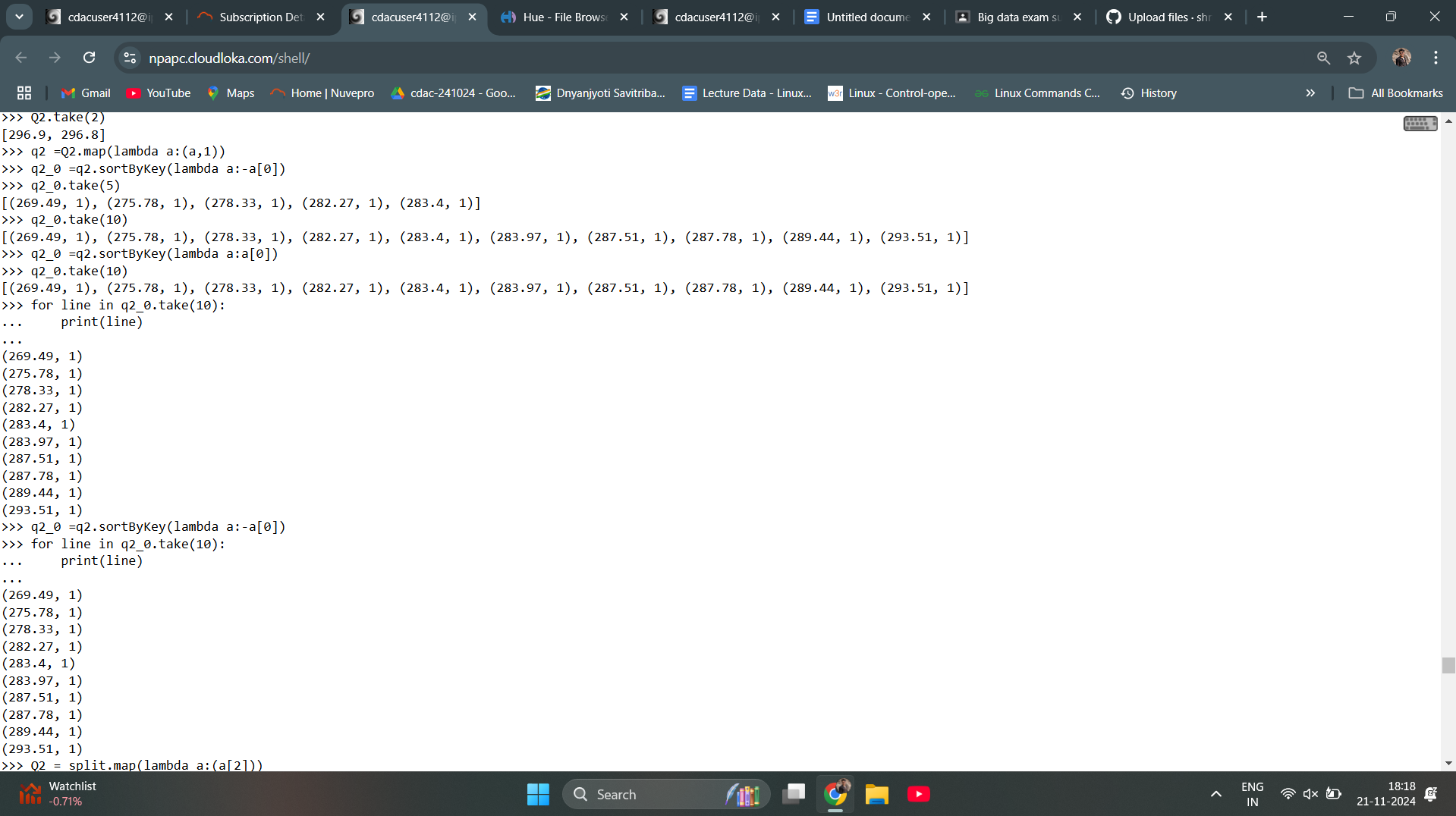
question 1)

1. Data = sc.textFile("/user/cdacuser4112/airlines.csv")
2. >>> header = data.first()
3. >>> filtered = data.filter(lambda a:(a !=header)
4. ... )
5. >>> filtered.count()

1. >>> split = filtered.map(lambda a:(a.split(",")[0],int(a.split(",")[1]),float(a.split(",")[2]),int(a.split(",")[3])))
2. >>> split.take(5)
3. >>> Quest2 = split.map(lambda a:(a[0]+" "+a[1])).distinct()
4. >>> for line in Quest2.take(20):
5. ... print(line)
6. 

**Question 2)**

2.



3. Q3 =split.map(lambda a:(a[1],a[3]))

>>> Q3.take(5)

[('1', 46561), ('2', 37443), ('3', 34128), ('4', 30388), ('1', 47808)]

>>> Q3\_0 =Q3.reduceByKey(lambda a,b:a+b)

>>> Q\_3.take(5)

Traceback (most recent call last):

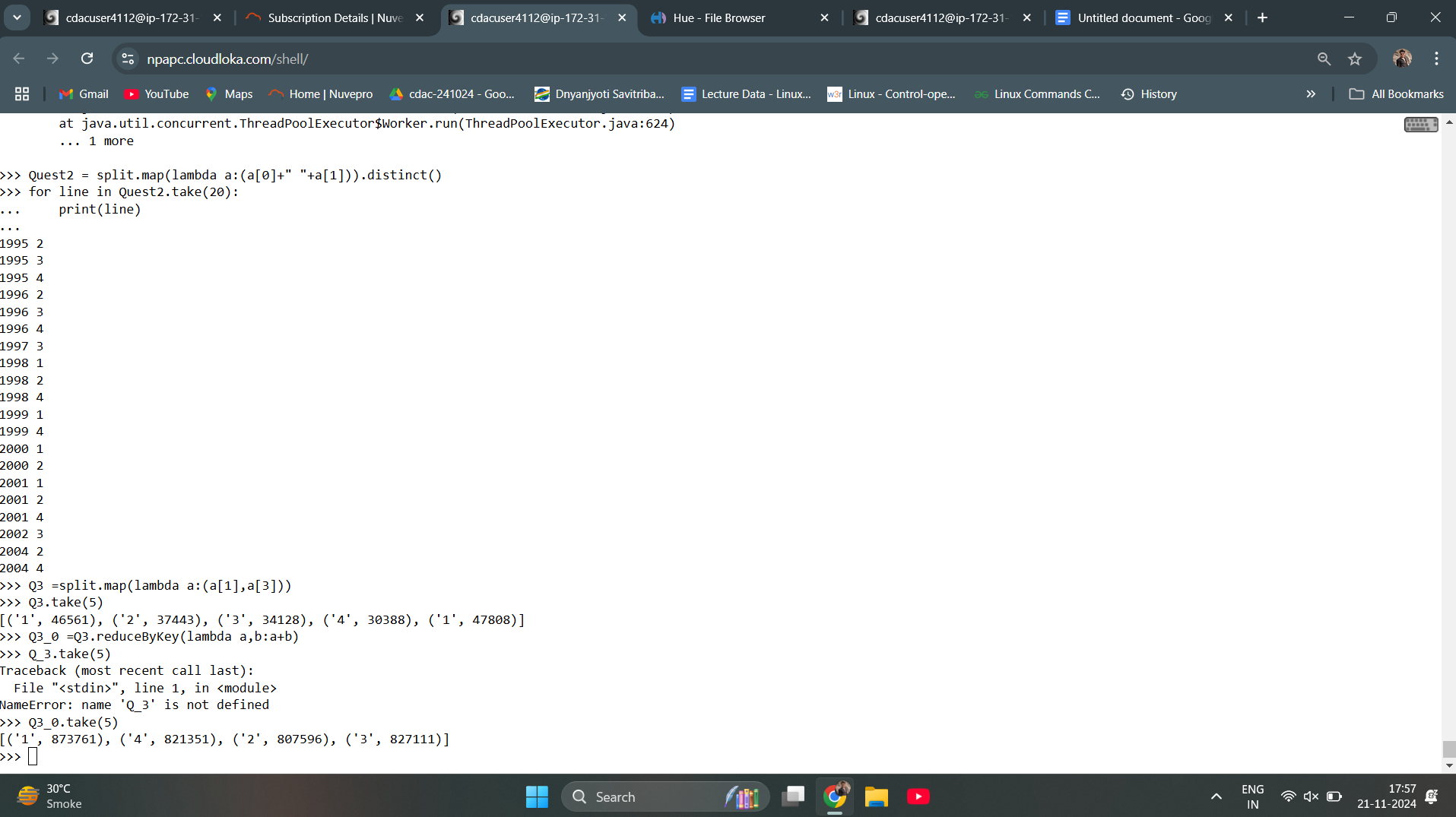
File "<stdin>", line 1, in <module>

NameError: name 'Q\_3' is not defined

>>> Q3\_0.take(5)

[('1', 873761), ('4', 821351), ('2', 807596), ('3', 827111)]

>>>



4.

>>> Q4 = split.map(lambda a:(a[0]))

>>> Q4.take(5)

['1995', '1995', '1995', '1995', '1996']

>>> q4 = Q4.map(lambda a:(a,1))

>>> q4.take(5)

[('1995', 1), ('1995', 1), ('1995', 1), ('1995', 1), ('1996', 1)]

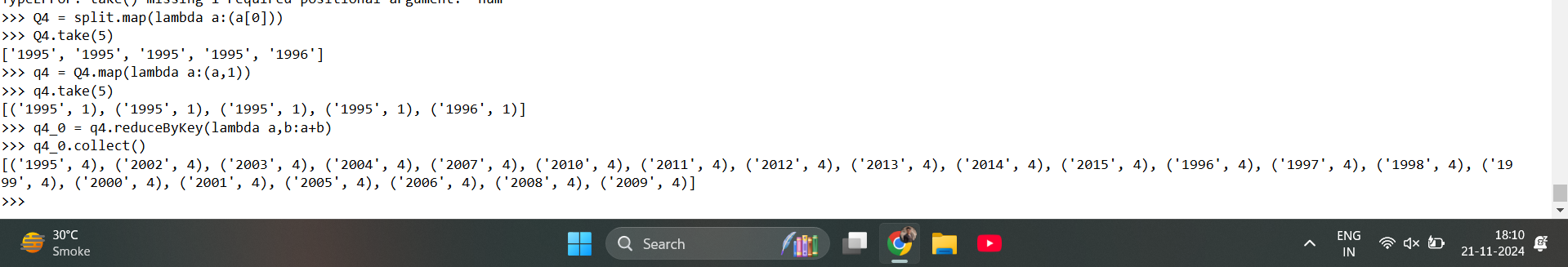
>>> q4\_0 = q4.reduceByKey(lambda a,b:a+b)

>>> q4\_0.collect()

[('1995', 4), ('2002', 4), ('2003', 4), ('2004', 4), ('2007', 4), ('2010', 4), ('2011', 4), ('2012', 4), ('2013', 4), ('2014', 4), ('2015', 4), ('1996', 4), ('1997', 4), ('1998', 4), ('19

99', 4), ('2000', 4), ('2001', 4), ('2005', 4), ('2006', 4), ('2008', 4), ('2009', 4)]

>>>



5.

>>> Q5 = split.map(lambda a:(a[0]+" "+a[1],a[2]))

>>> Q5.take(5)

[('1995 1', 296.9), ('1995 2', 296.8), ('1995 3', 287.51), ('1995 4', 287.78), ('1996 1', 283.97)]

>>> Q5 = split.map(lambda a:(a[0]))

>>> Q5.take(5)

['1995', '1995', '1995', '1995', '1996']

>>> Q5 = split.map(lambda a:(a[0],a[2]))

>>> Q5.take(5)

[('1995', 296.9), ('1995', 296.8), ('1995', 287.51), ('1995', 287.78), ('1996', 283.97)]

>>> q5 = Q5.reduceByKey(lambda a,b:a+b)

>>> q5.take(5)

[('1995', 1168.99), ('2002', 1250.1), ('2003', 1261.87), ('2004', 1223.5), ('2007', 1300.56)]

