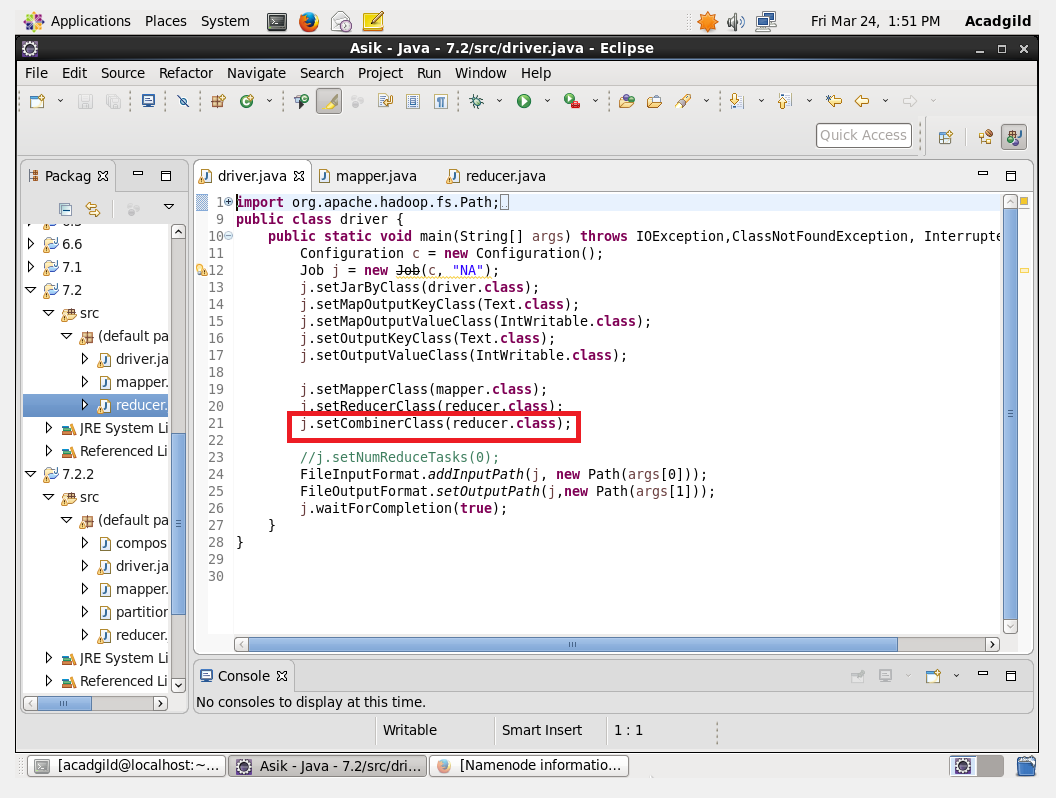
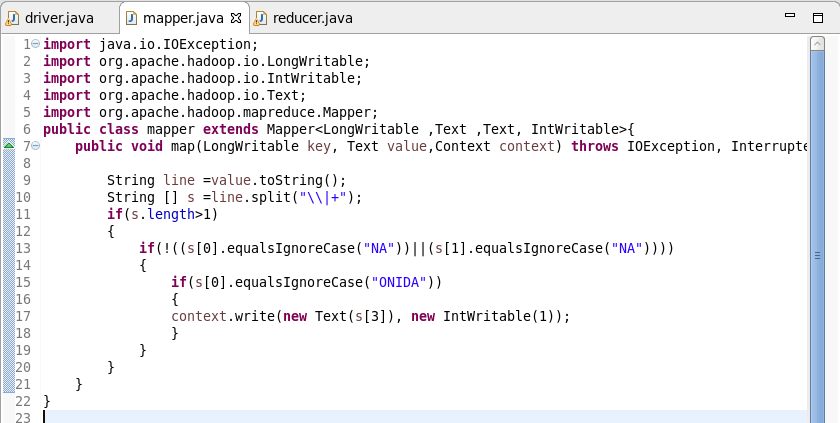
**ASSIGNMENT 7.2:**

1. Sales of different TV Task 6 : Modify Sales of different TV Task 3 (refer session 5, assignment 2) to take advantage of Combiner.

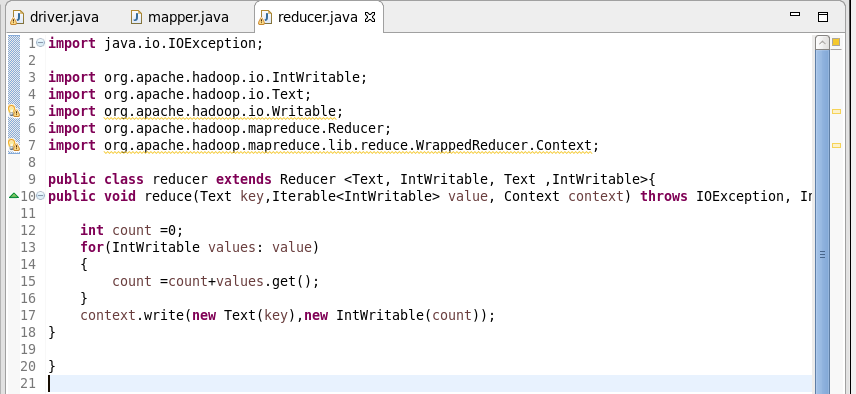
Driver Class:



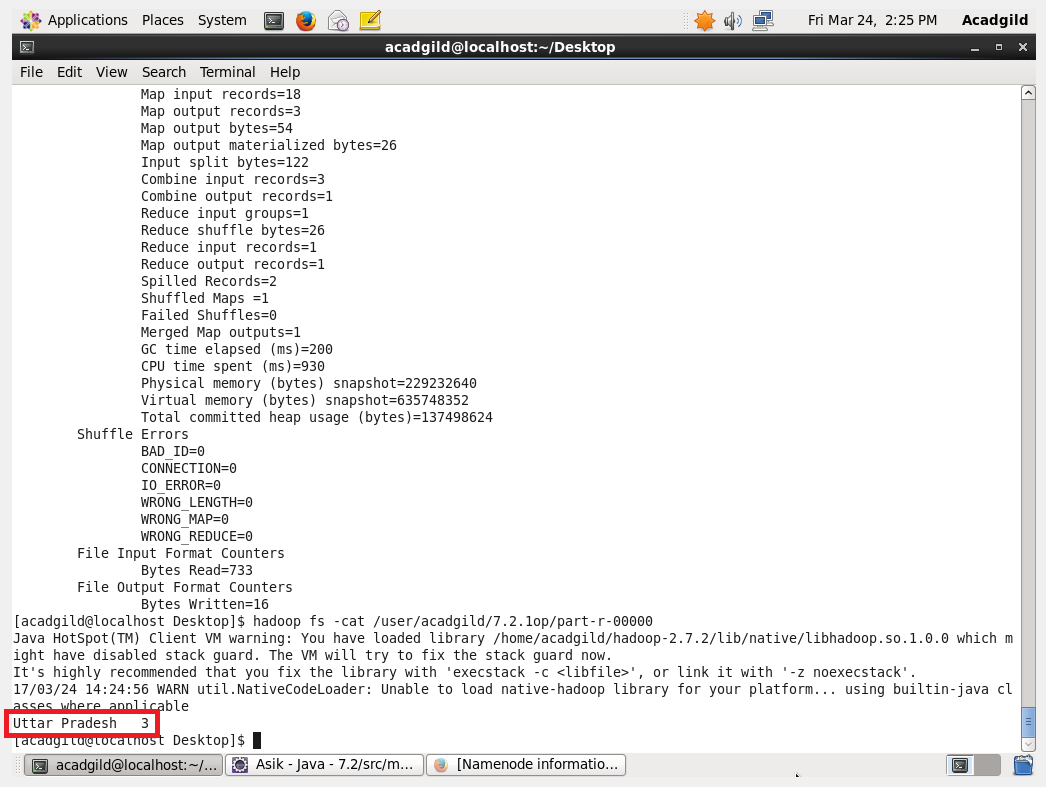
Mapper Class:



Reducer Class:



Output:

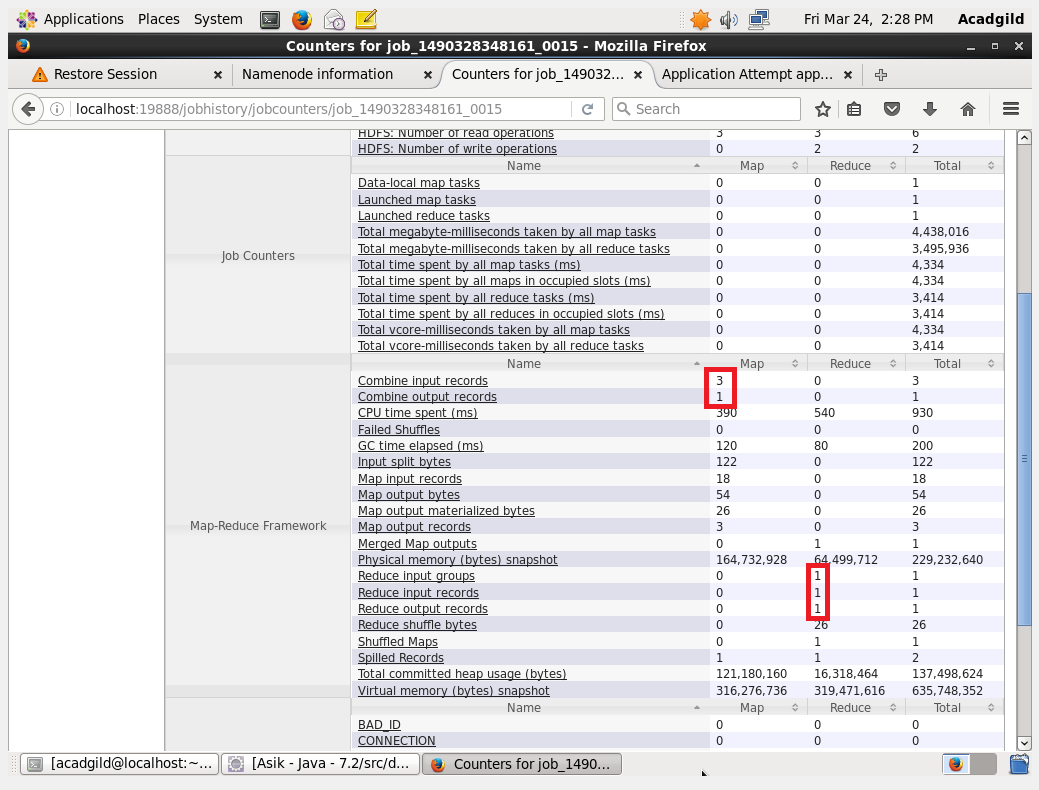


Job History Information:

Use of Combiner:

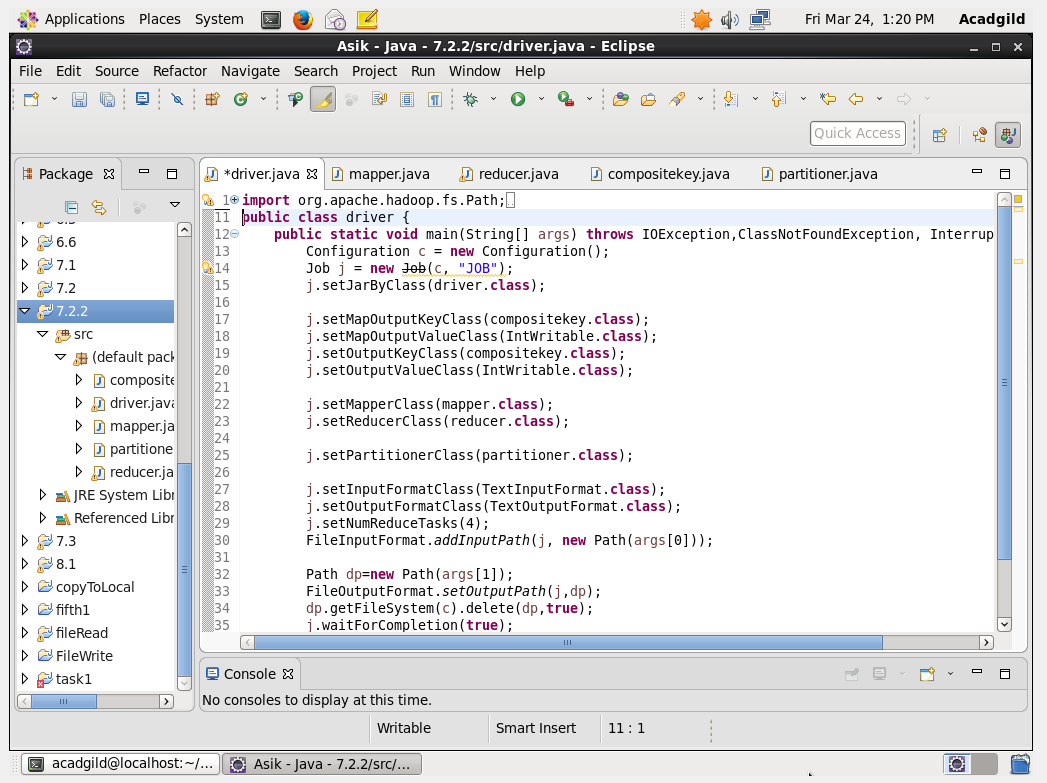
The Combiner class is used in between the Map class and the Reduce class to reduce the volume of data transfer between Map and Reduce. Usually, the output of the map task is large and the data transferred to the reduce task is high.

Here the Map Output records is 3 and the Reduce Input records in only 1. This is because of the use of combiner class.



1. Write a Mapreduce program to view the total sales for each product for every Company corresponding to each size. Make sure that all records for a single company goes to a single reducer and inside every reducer, keys must be sorted in descending order of the size. You may write a custom WritableComparable for this purpose.

Driver Class:



Composite Key:

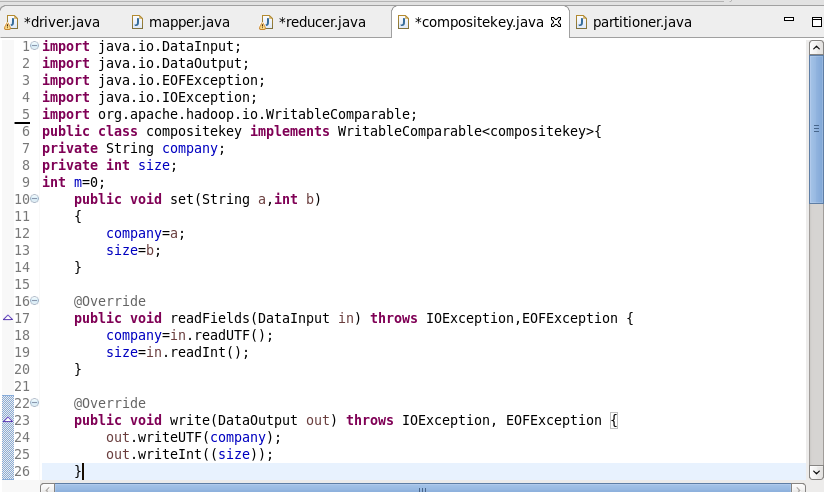
Step 1 : Overriding HashCode returning hashcode of company

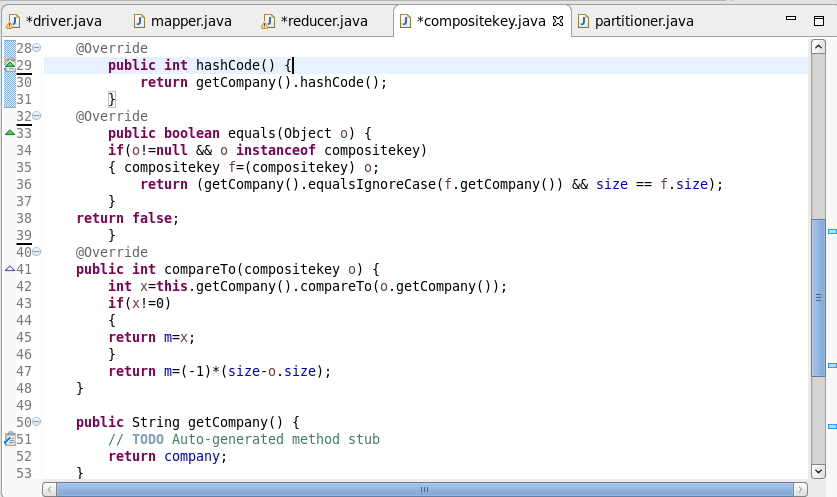
Step 2 : Overriding equals Method: checks the company and size are equal for every object if equal it will send to the corresponding HashCode location else compare to will be done

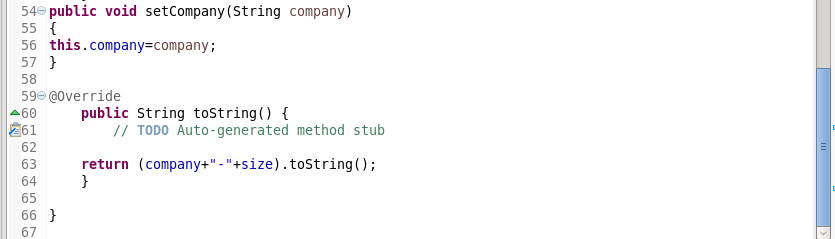
Step 3 : Overriding Compare to method: if both the company are equal if its same it will return (-1)(size-object’s size)(**-1 will reverse the sort descending based on size**) and it will be send and framework sorting will be made on it or else if different it will send the company and framework sorting will be made on it

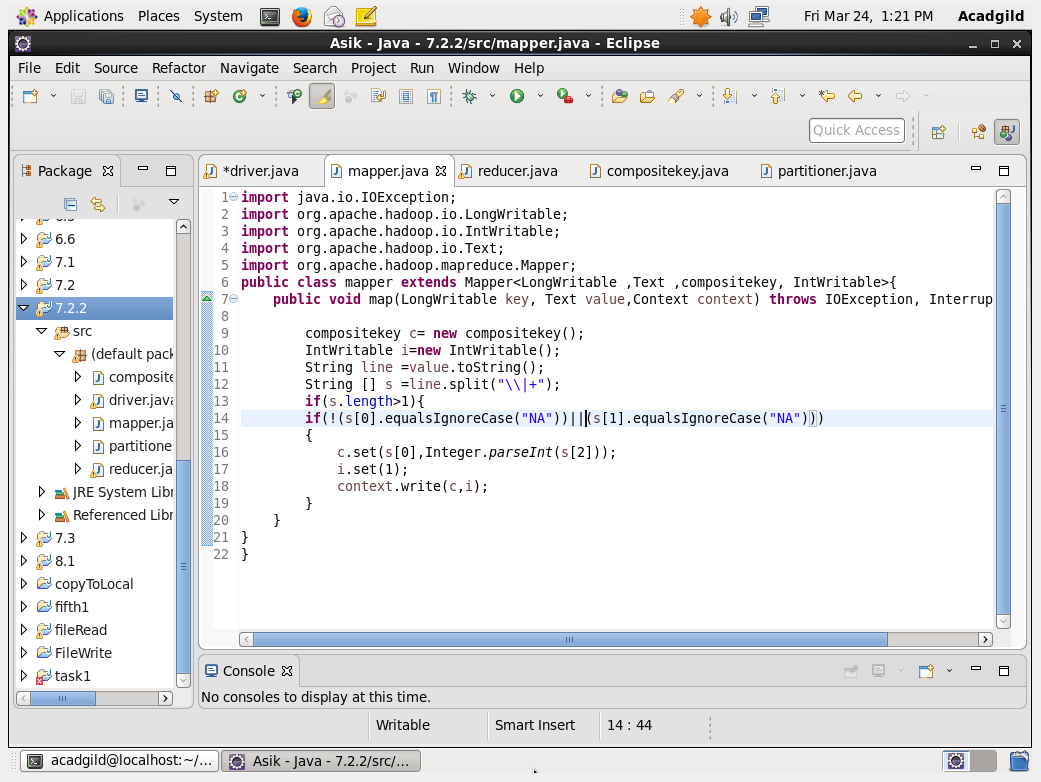
Step 4 : overriding toString method to print the company and size.

COMPOSITE KEY CLASS:

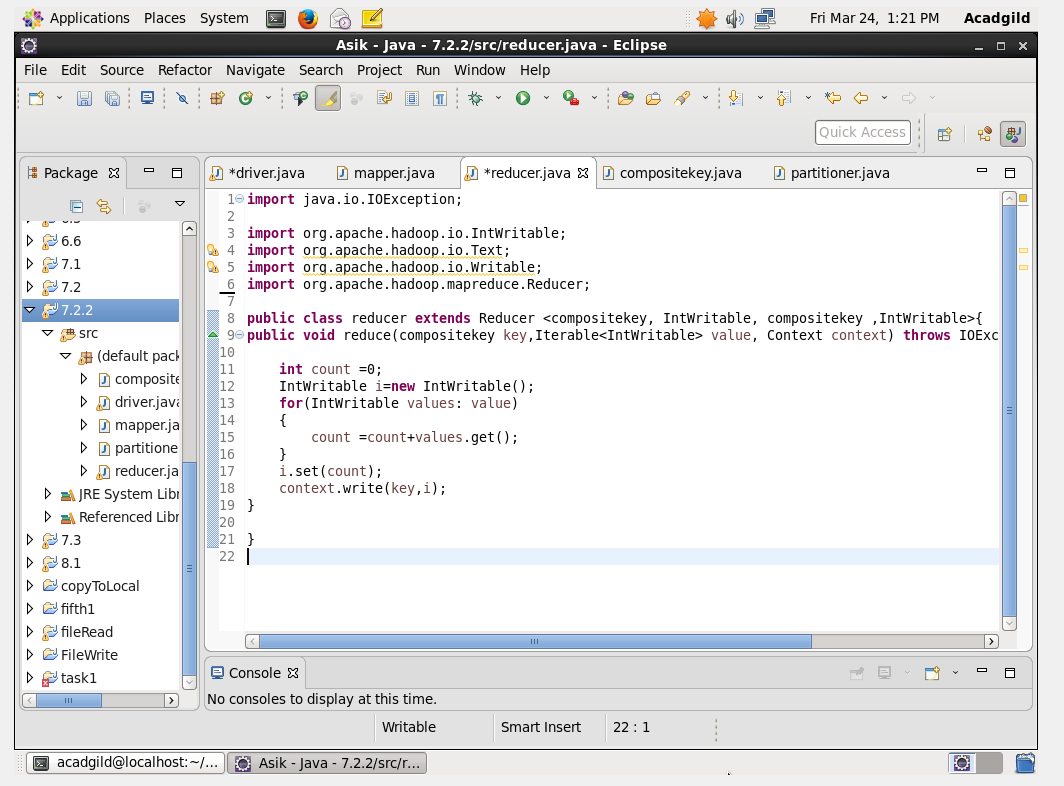




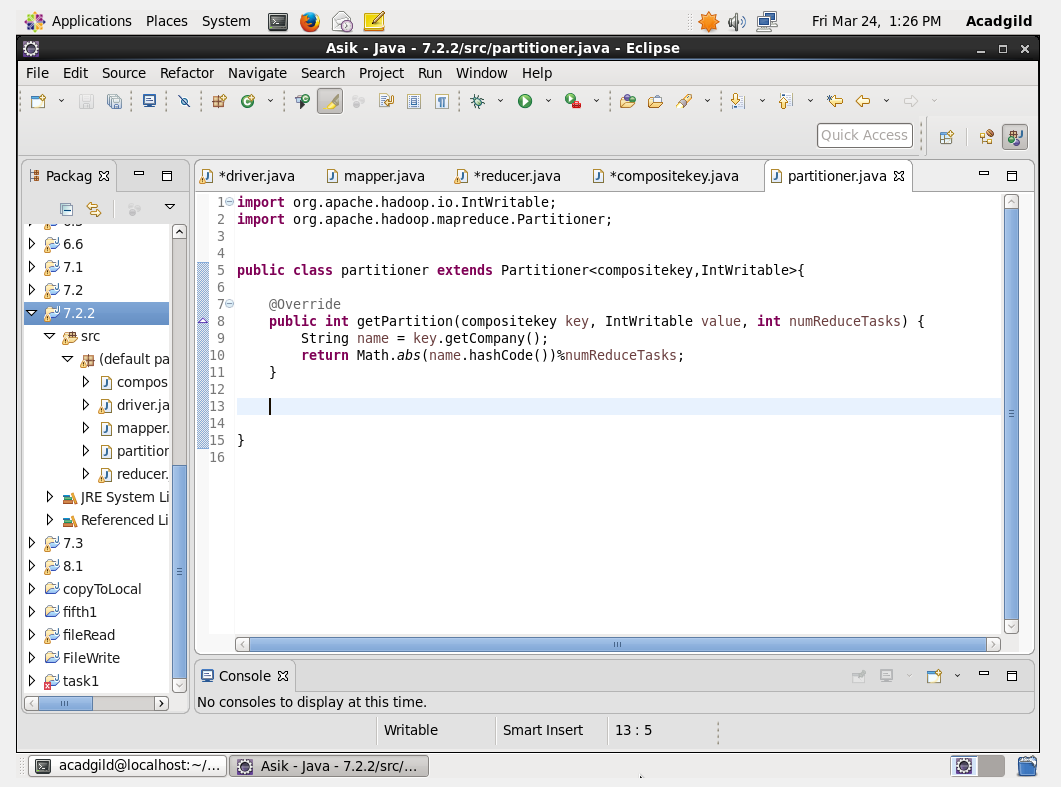


Mapper Class:

Reducer Class:



Partitioner Class:



Output: