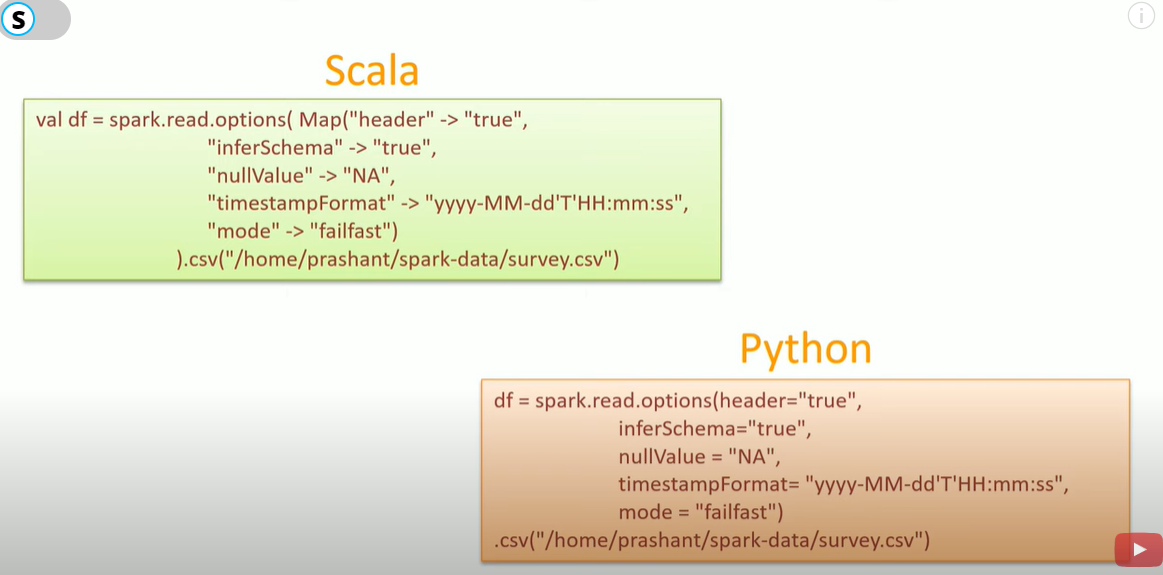
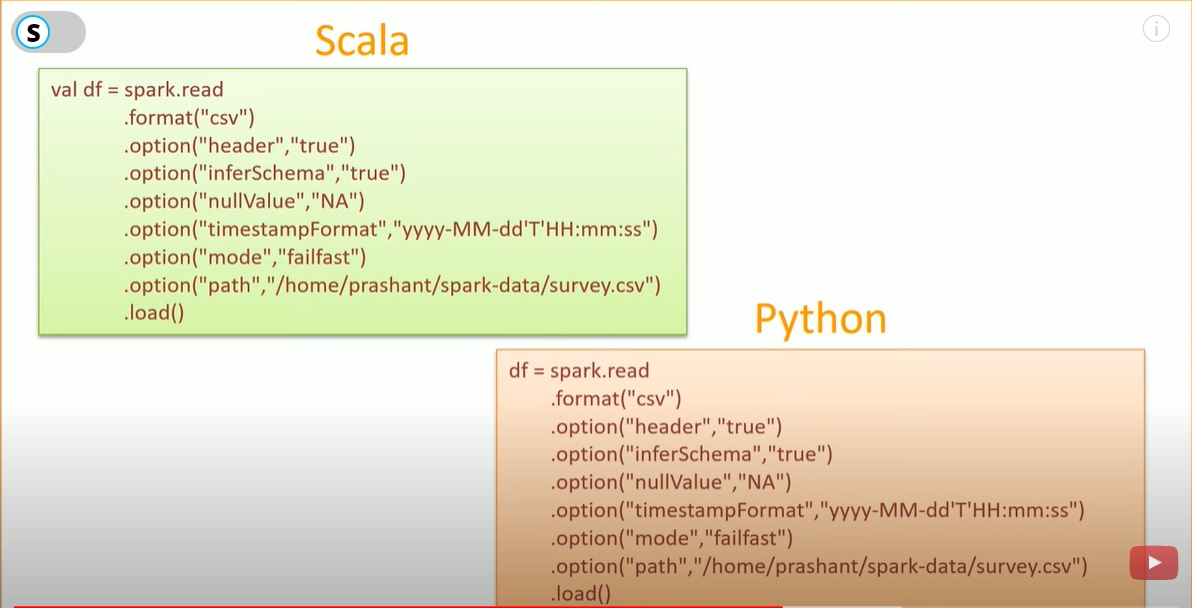


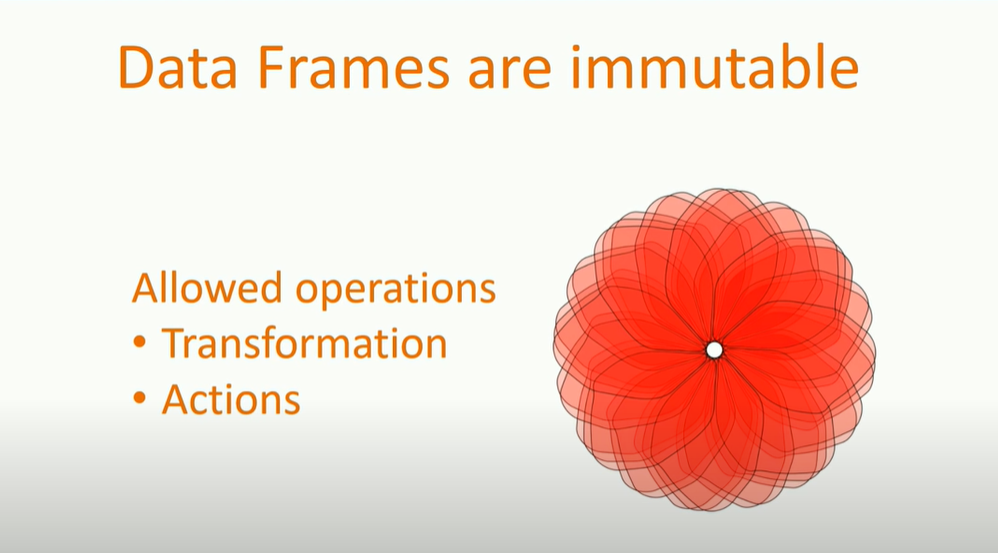


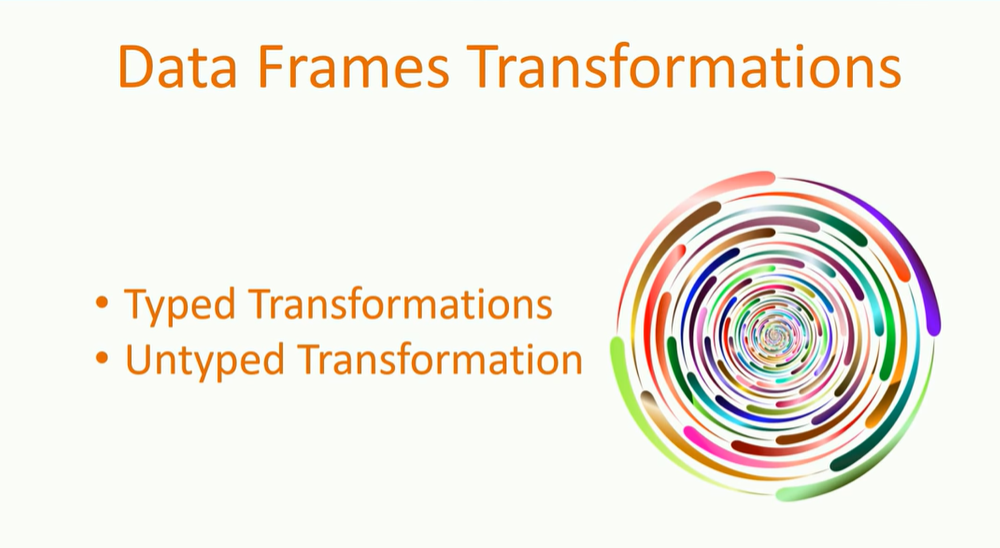
Code to load data from CSV file.



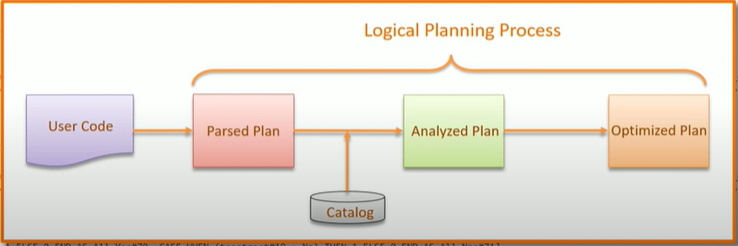
The above one is specific to csv file, we have more generic code ,refer the below code.







When there is a transformations followed by action the following plan takes place:



1.When we execute an action SPARK takes the user code.

2.It will parse the user code and generate PARSE LOGICAL PLAN.

3.Apache spark maintains a catalog of all the tables and dataframe information.

4.The analyzer makes a call to the catalogue and resolves the initial plan and resolve column names and their data types. Output is an ANALYZED LOGICAL PLAN.

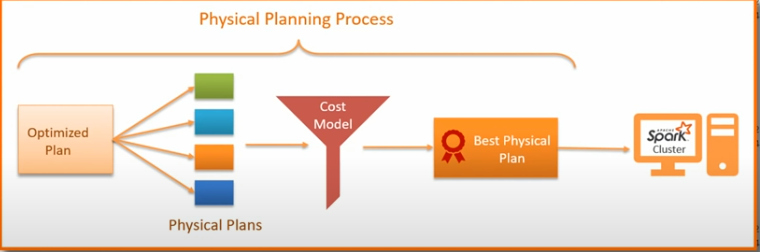
5 Optimizer picks the ANALYZED LOGICAL PLAN .Here optimizer performs two operations:

1.Pipelining :Combining multiple transformations together

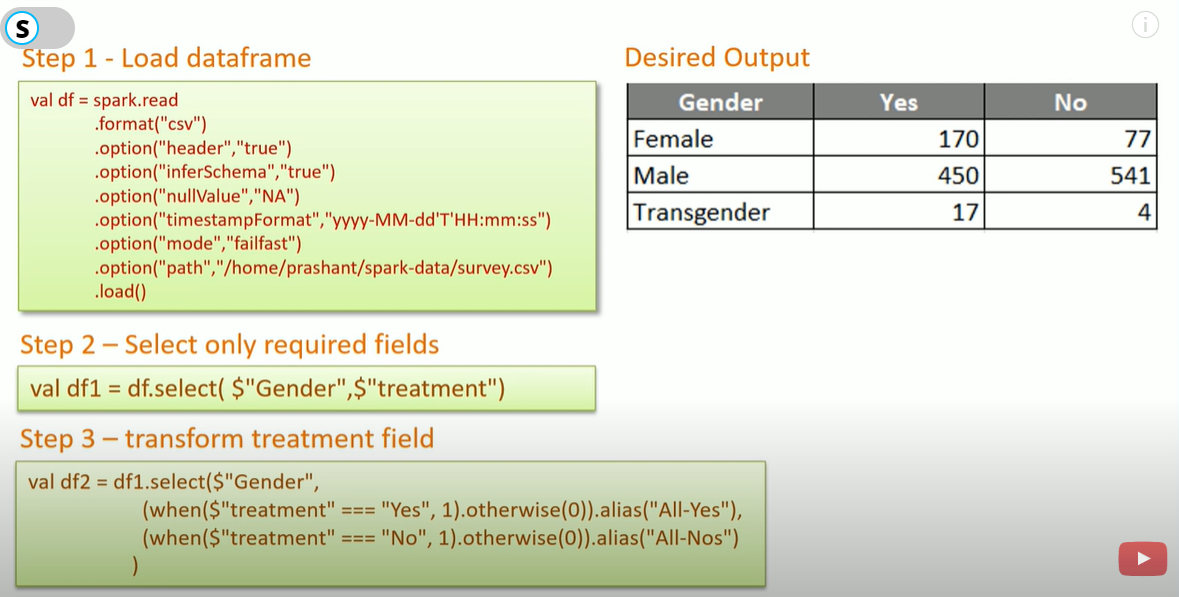
2.Predicate Pushdown : Pushing down the filter conditions at the early age rather than pushing it down at the end.

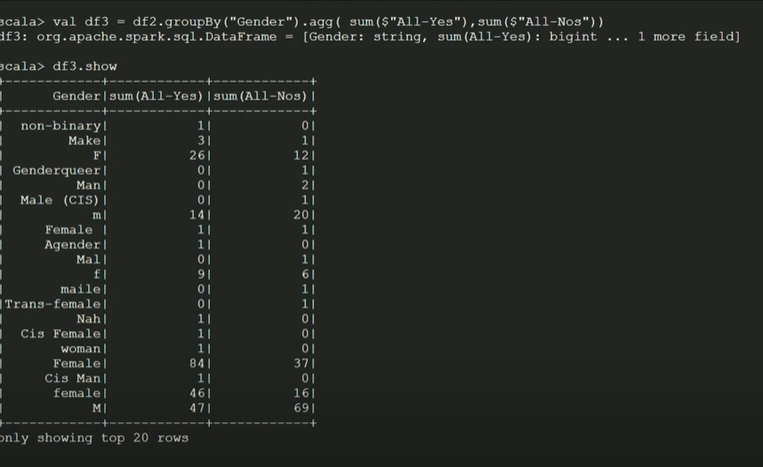
6.Optimized logical plan goes to the compiler that generates bunch of physical execution plans.

7.Physical execution plans are nothing but bunch of RDD transformations



8.Finally sparks settles for one best physical plan and sends it for execution

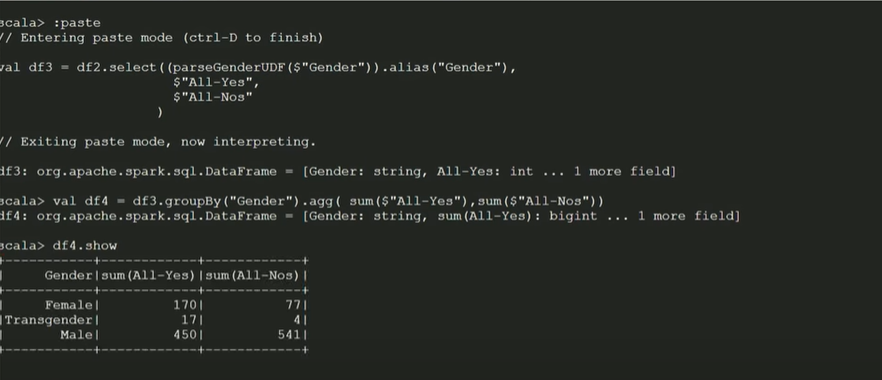


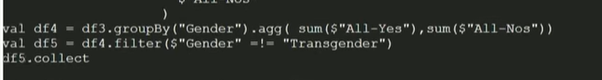


In the above output we can see that the o/p is not as per the desired one.Here is a data quality issue.



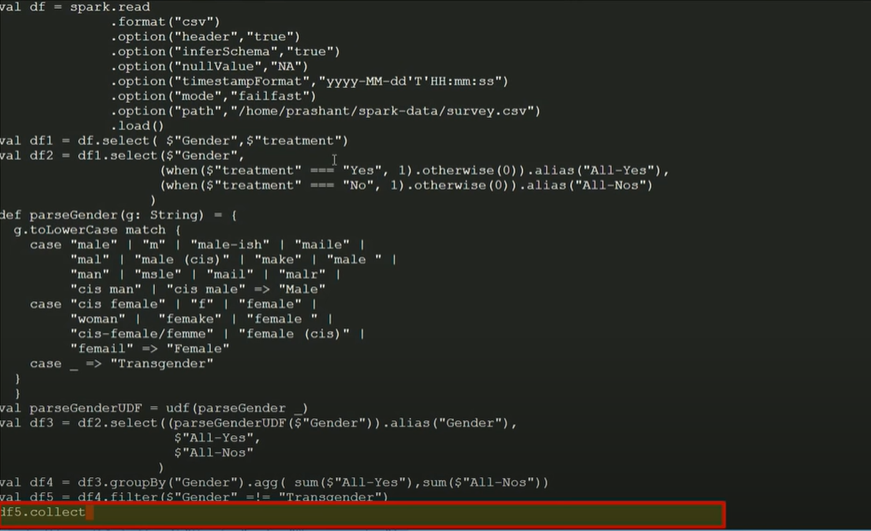
In the above we created a function to categories all the gender to three categories male,female and transgender.But I want this function to be available to all the executors.So in this case we will need to register the function as UDF .Spark will serialize the function on driver and transfer it over the network to all the executors.





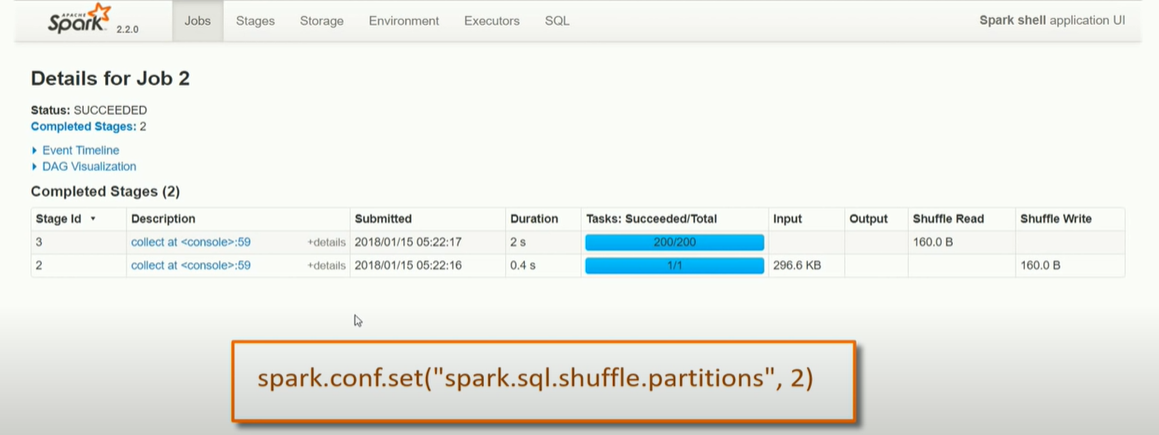
(before df4.show apply val df5 line)

Now how all this together will work



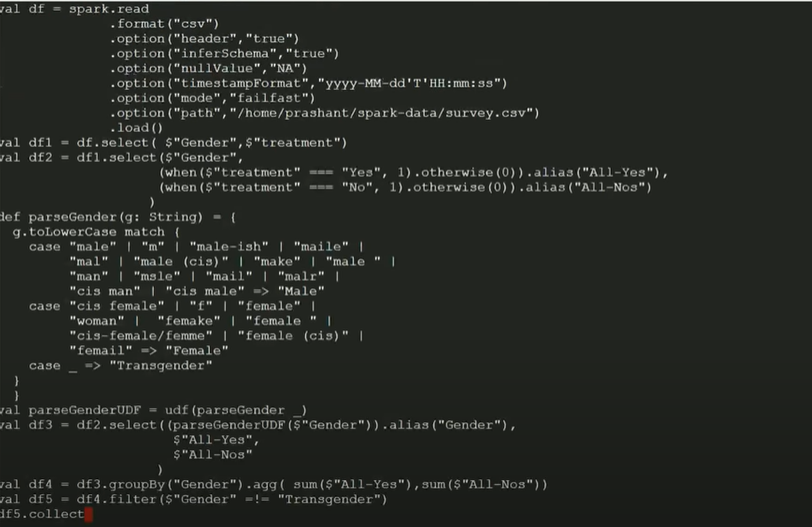
Here the operation takes place in two stages one before the shuffle another after the shuffle.

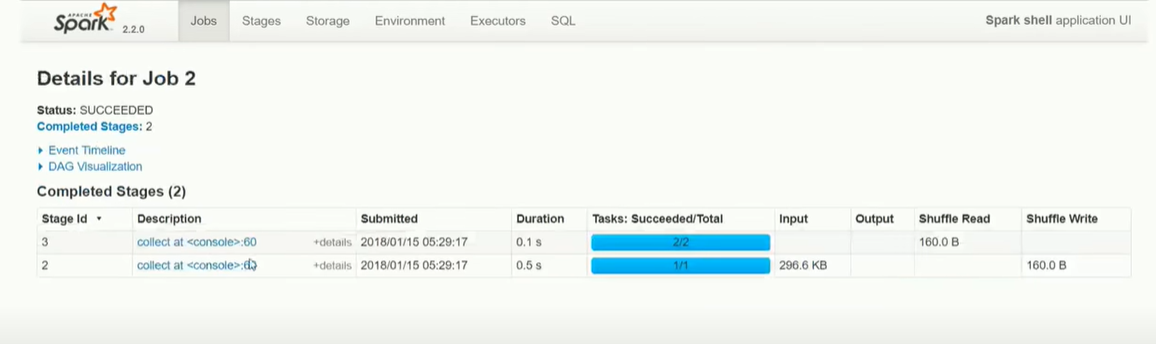
But we can see that 1st stage has 1 task or 1 partition but 2nd Stage has 200 task or partitions.



Now lets add this to the code(at the start)







Now we have 2 partitions or task instead of 200.