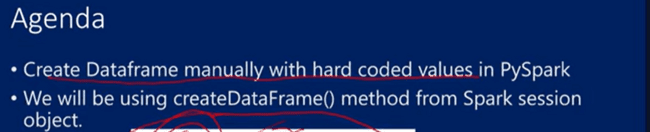
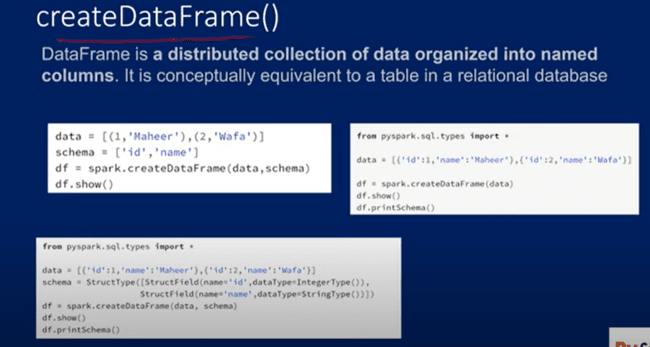
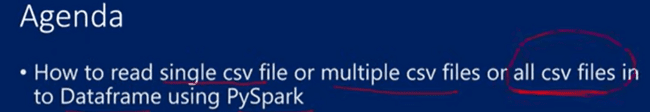


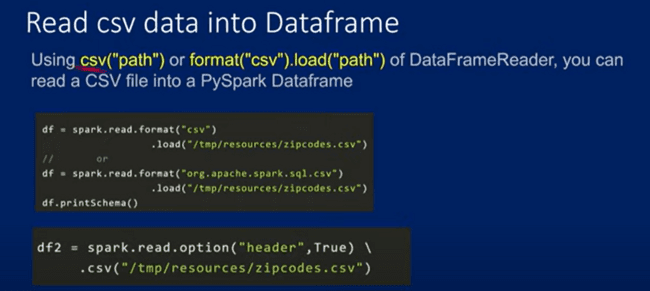
2.





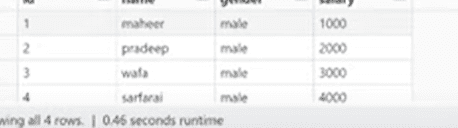
3.

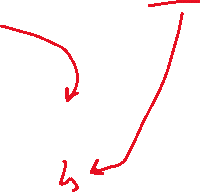


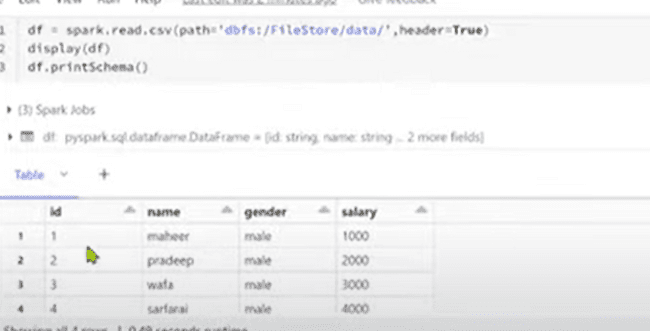










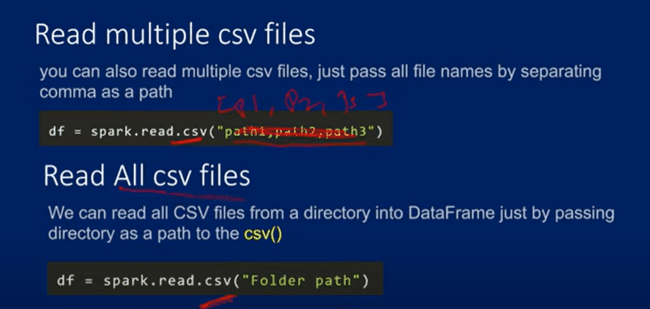




StructType.add( )

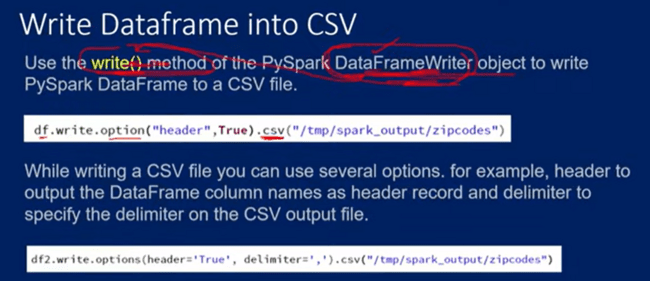


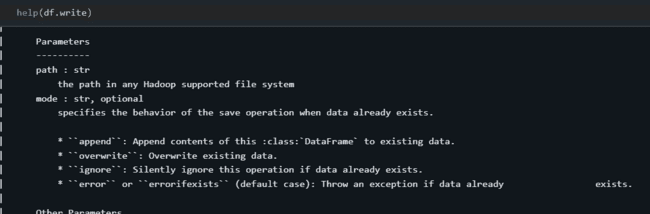




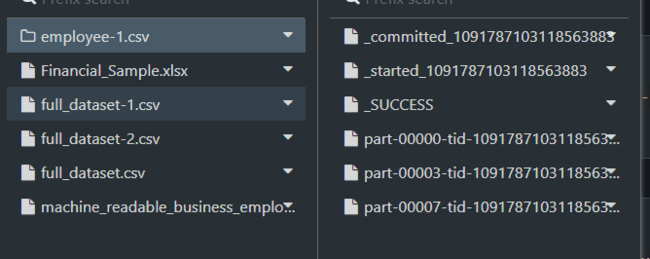
4. 





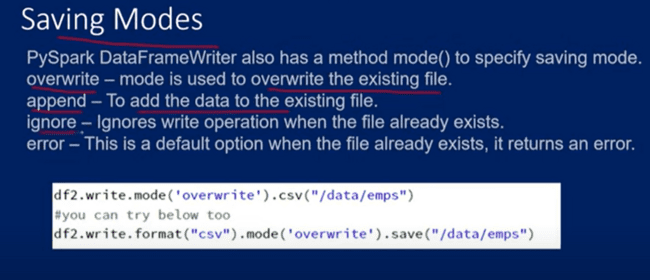


The worker nodes 🡪 here are seen when we create from df to csv

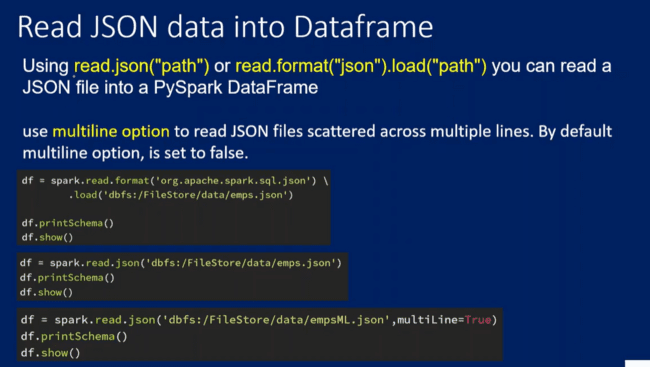
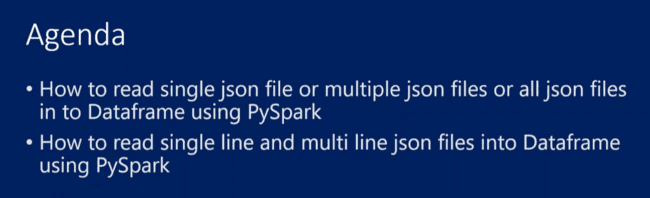


When we repartiontion into 1 file ,  **df.write create a folder** then data will be there inside like 3

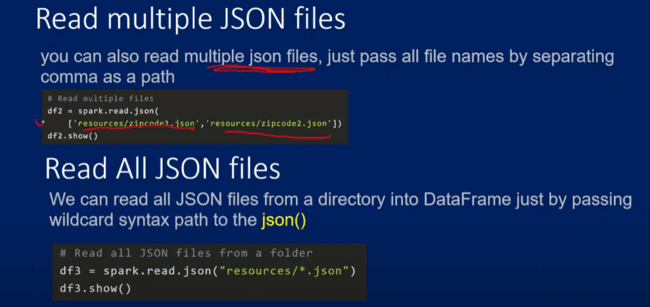
Df.repartioton. csv()



5.

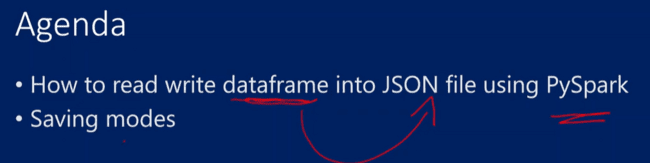


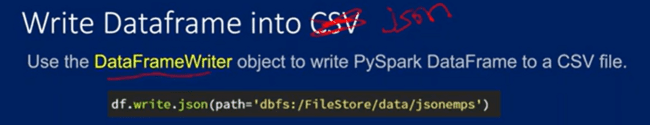


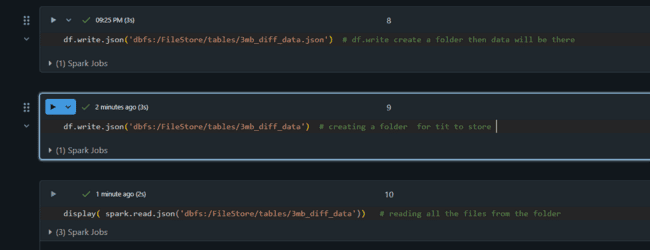




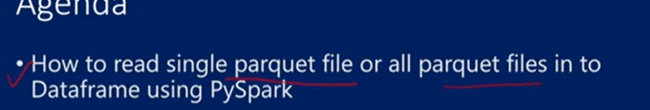
6

****

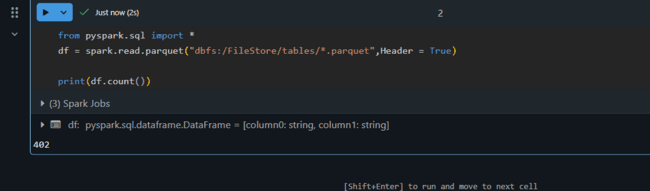
****

****

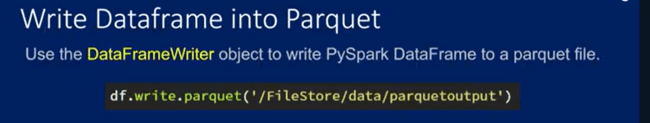
**7.**

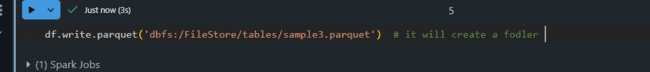
****

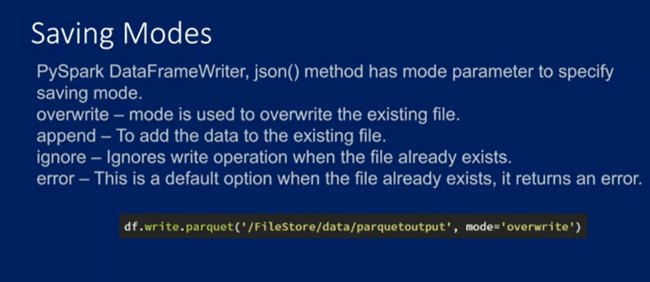
****

****

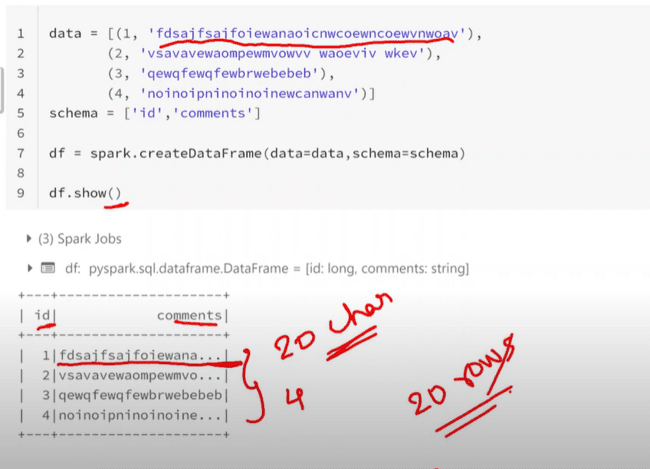
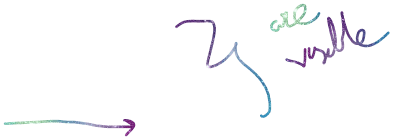
**8.**

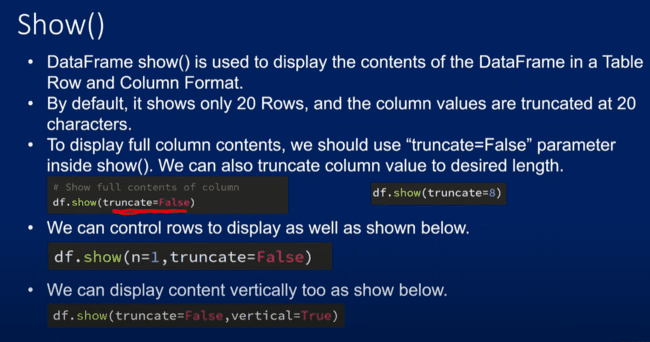
****

****

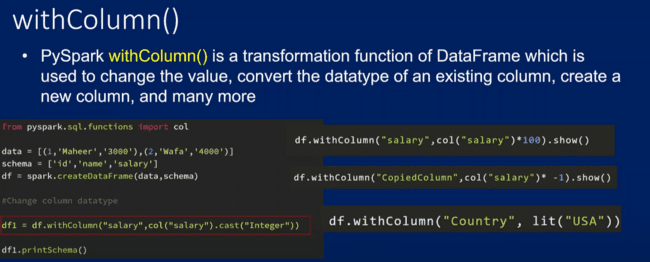
****

**9.**

****

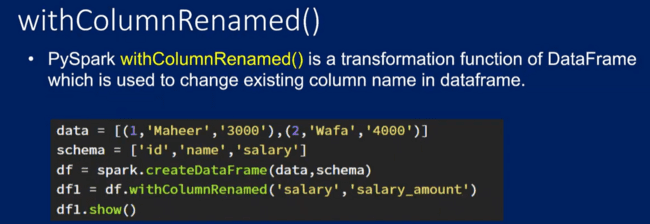
****

**10.**

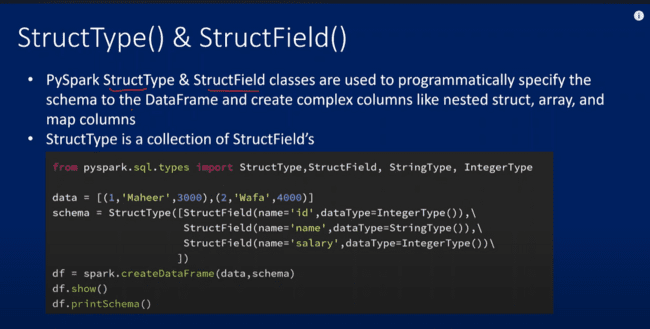
****

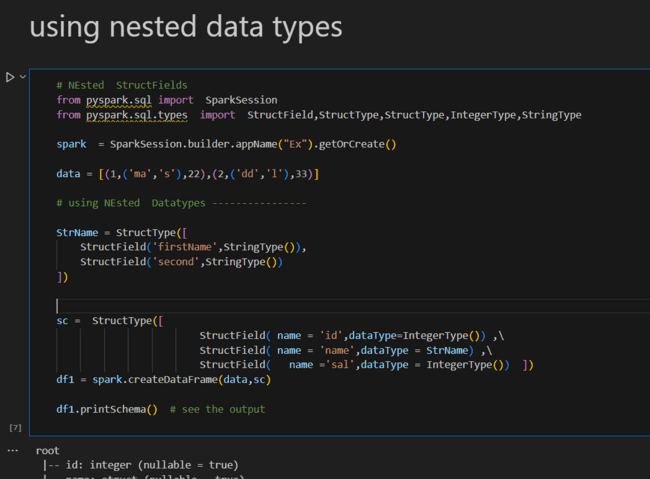


**11.**

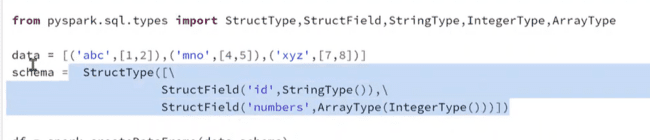
****

**12**

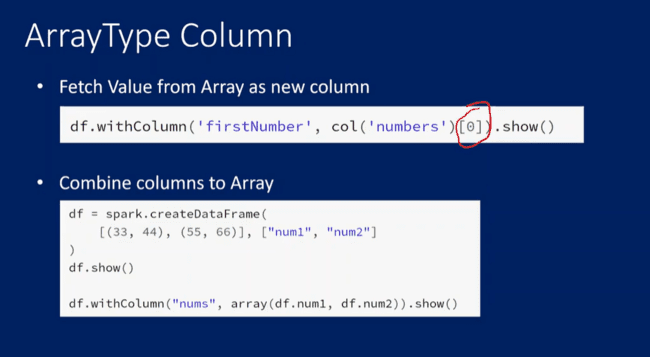
****

****

**13.**

****



****

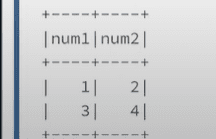


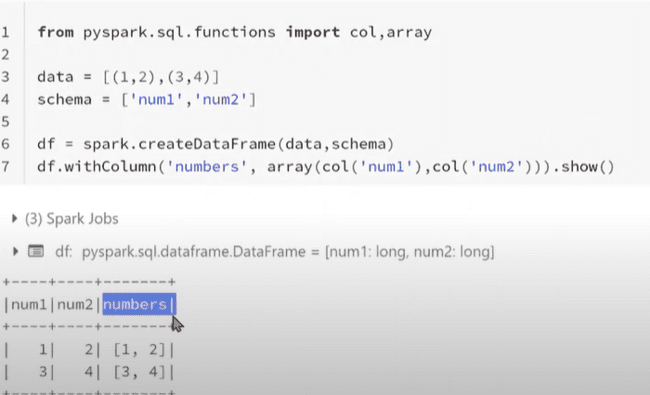
****



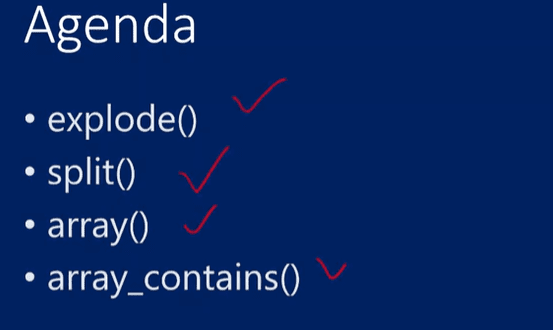
****

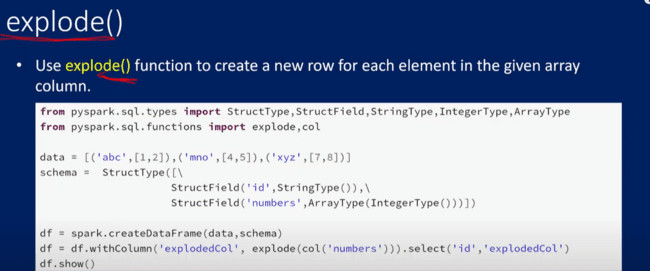
**Combining 🡪 to form array**

****

****

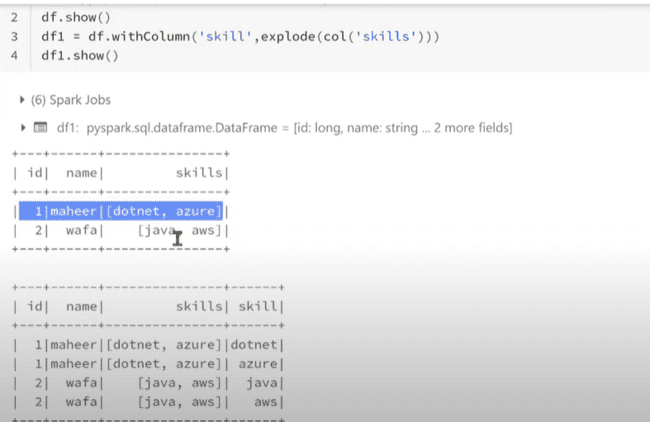
**14.**

****

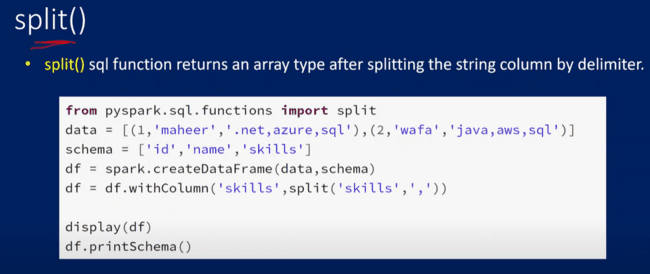




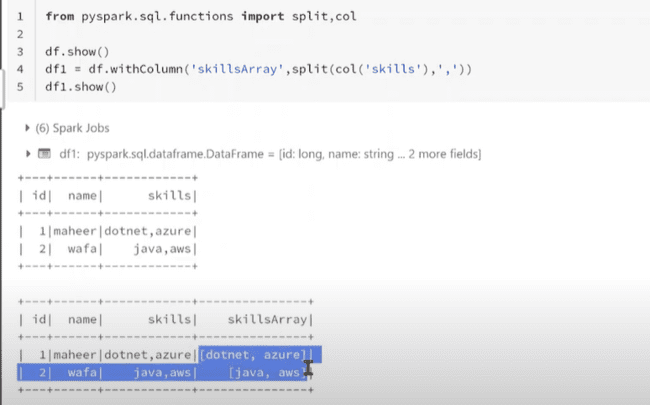
Ex: 🡪 ex



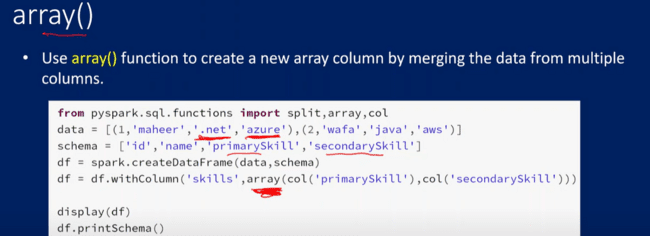




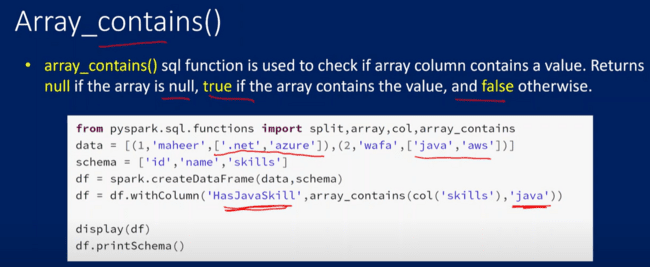






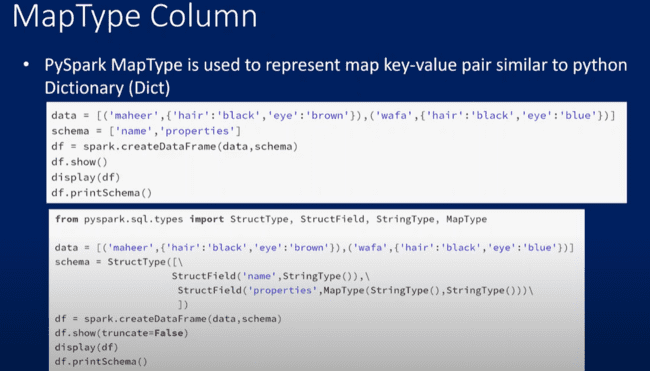




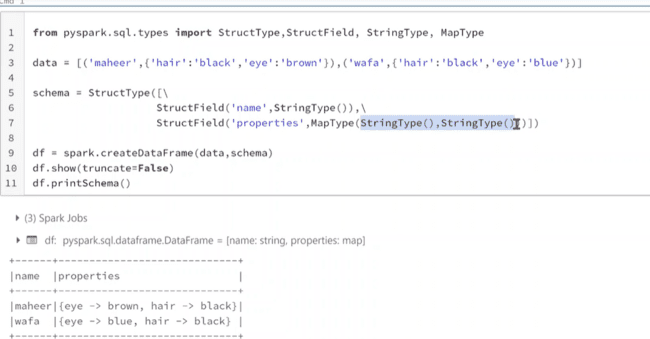


Check if java is there , if java is there return true

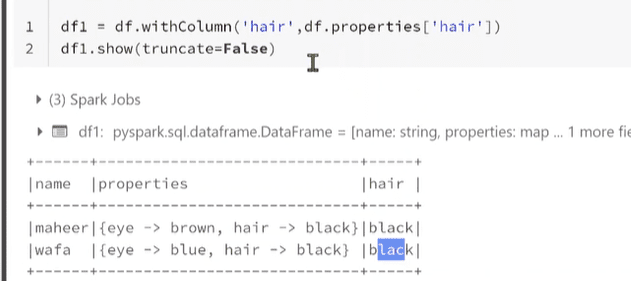
15.



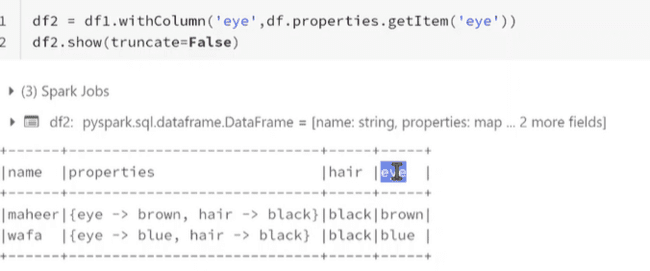






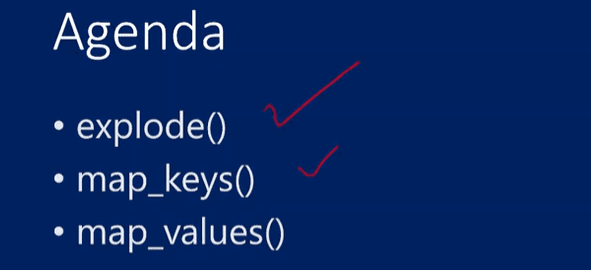


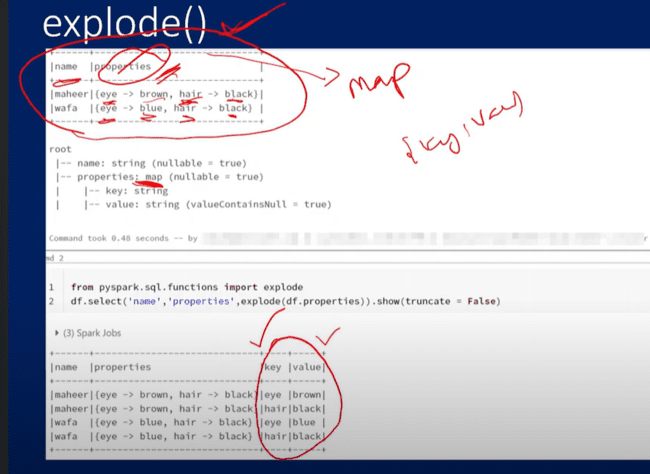




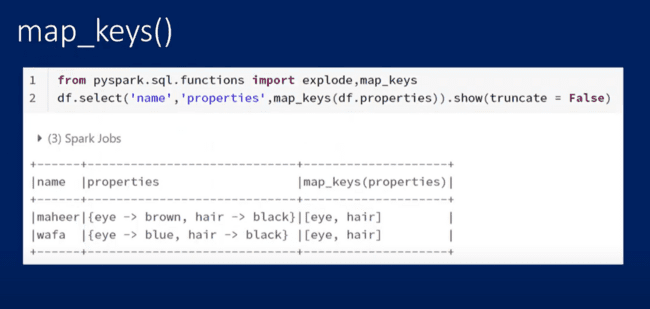


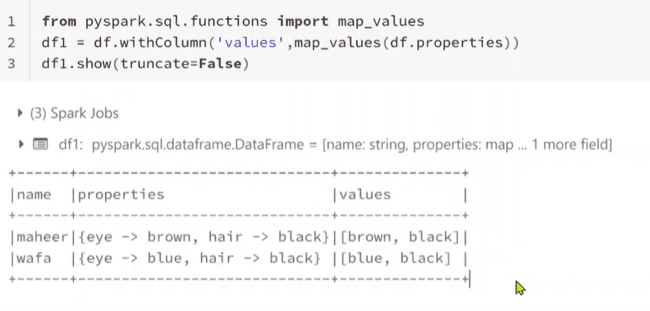
16.





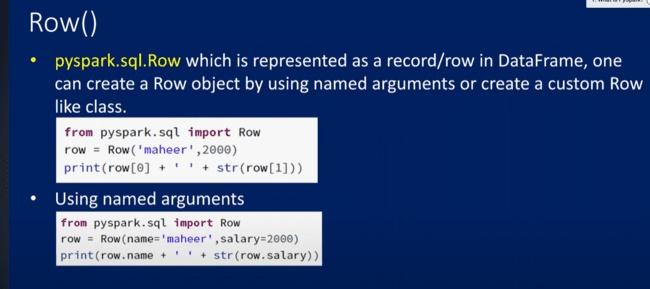


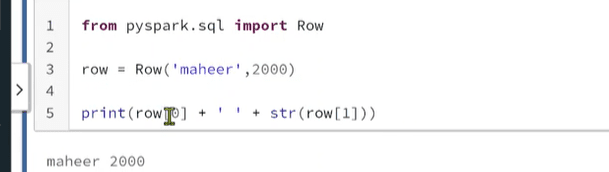


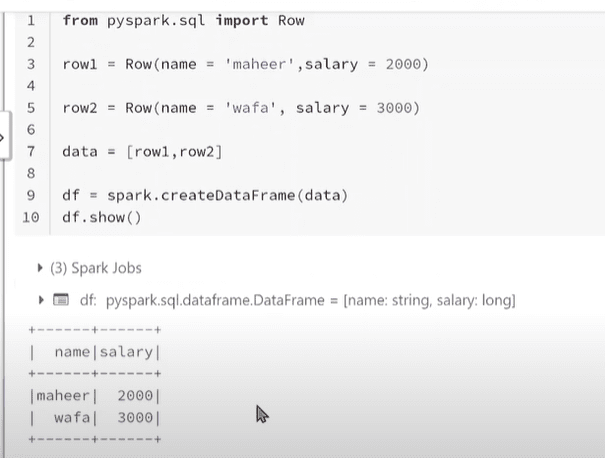


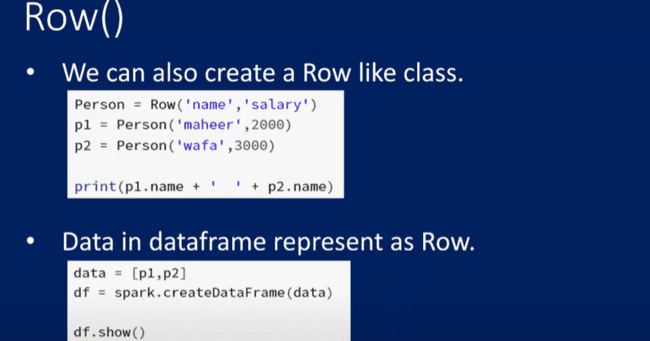


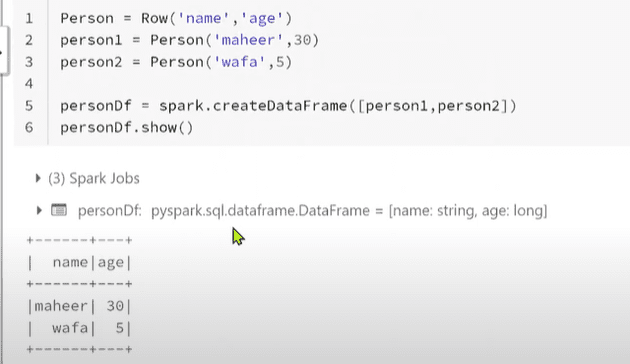
17.

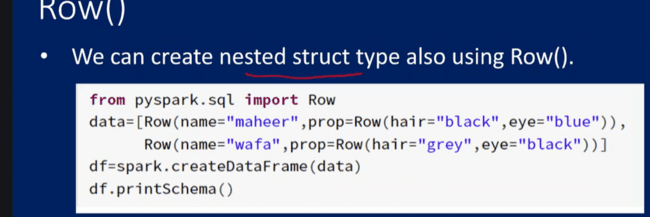






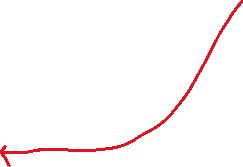




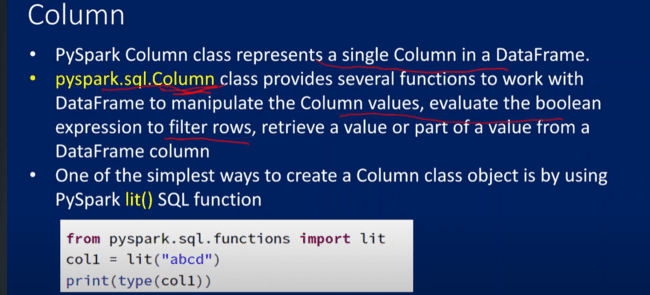


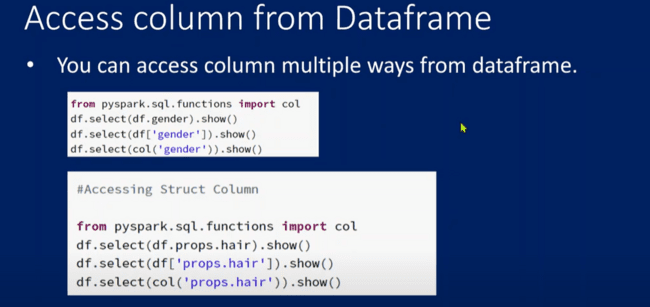




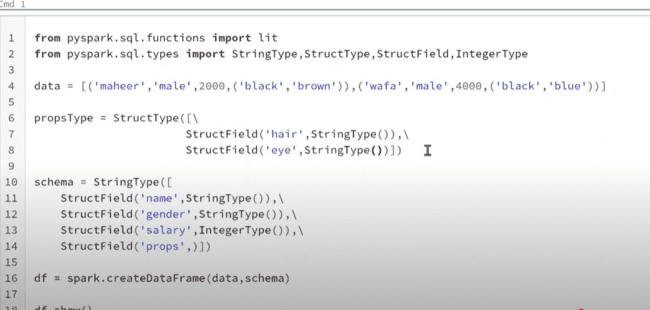


18.





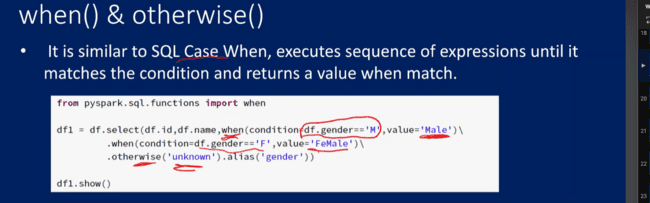


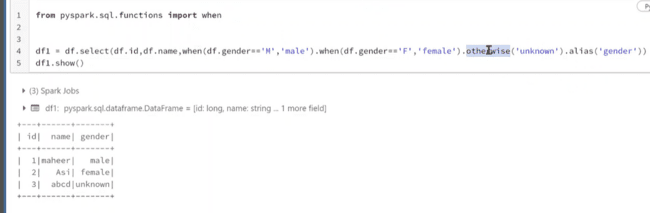


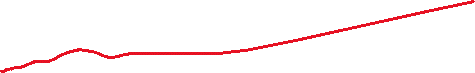




19.

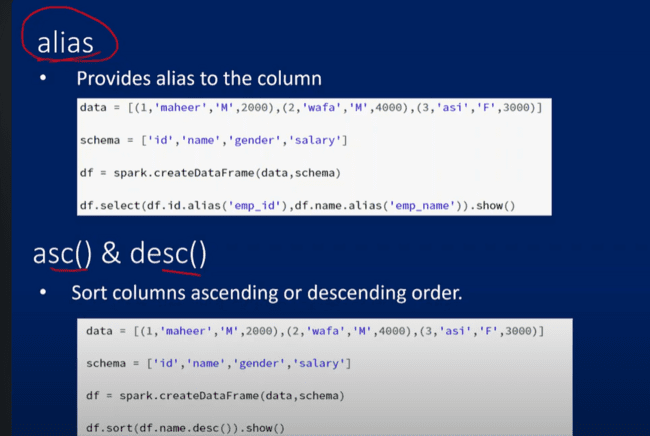




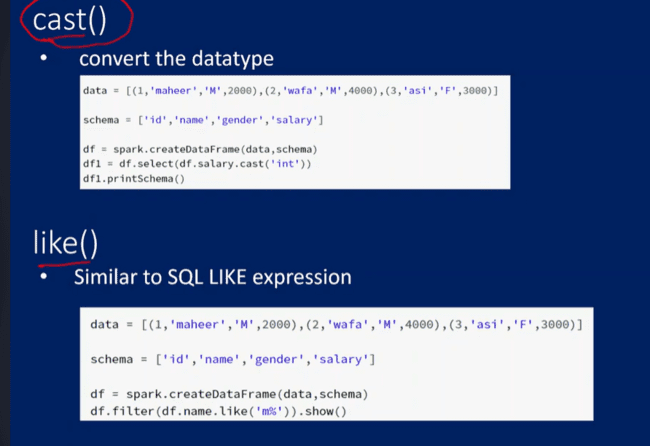


20.



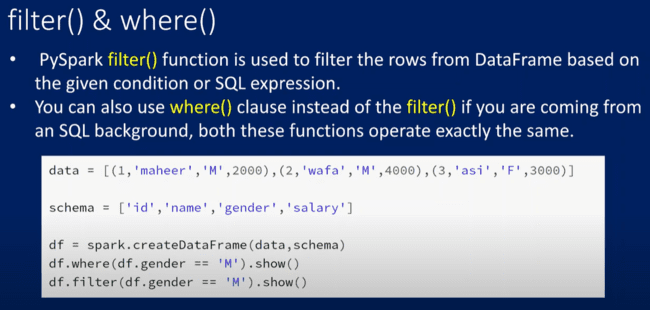








21





22.



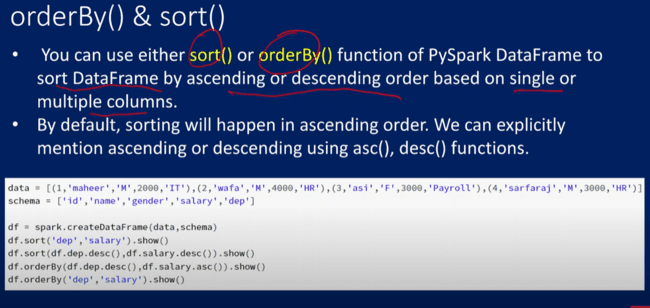
df.distinct().count()

In **PySpark**, df.distinct() is **not used to get distinct values of a specific column** — it returns **distinct (unique) rows across all columns**.

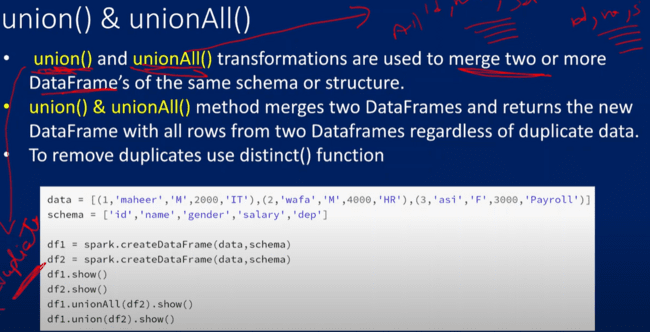
df.distinct("gender") ❌

df.distinct(["gender"]) ❌ 🡪 correct 🡪 df.select("gender").distinct()

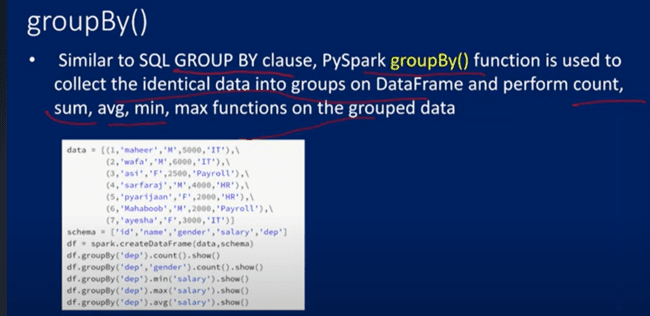
23

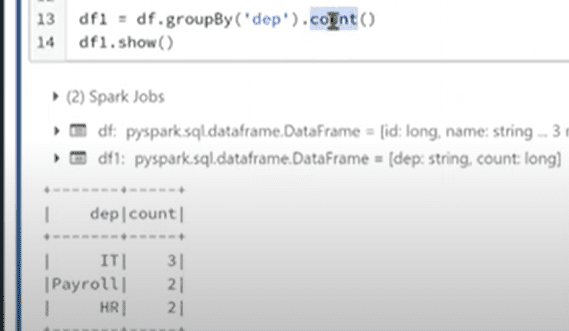


24.

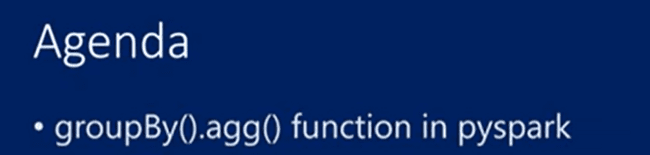


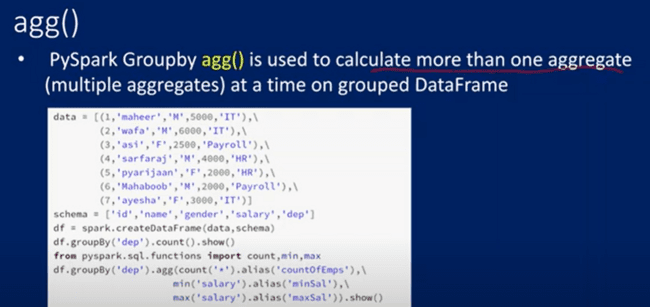
25.





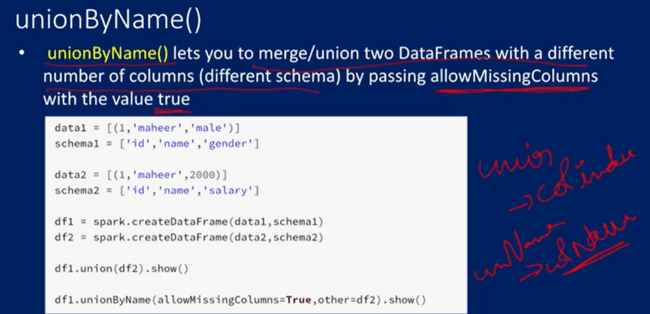
26.





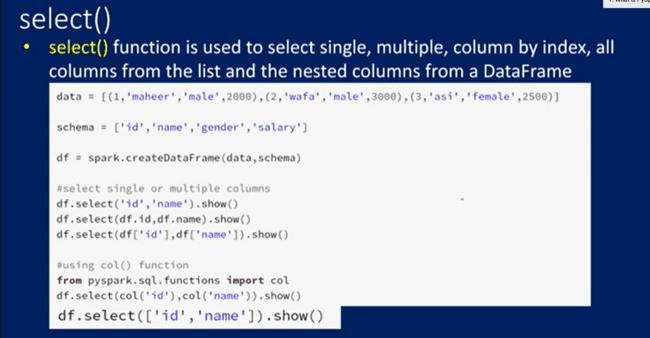


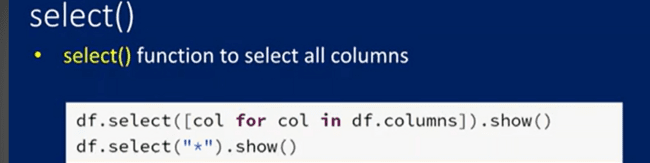
27.





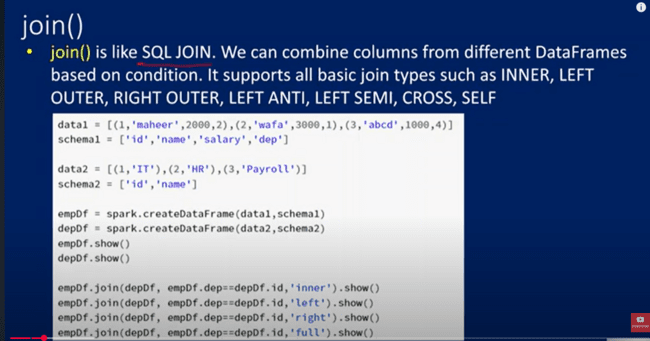
**28.**

****

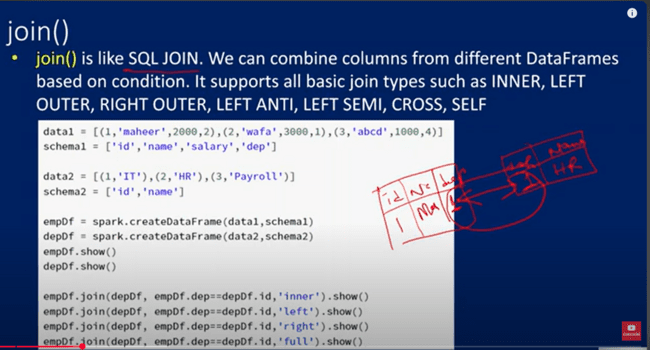
****



**29.**

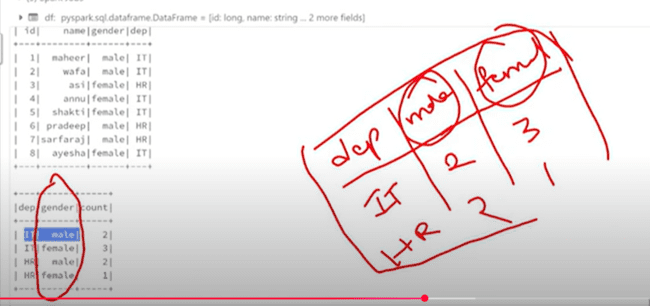
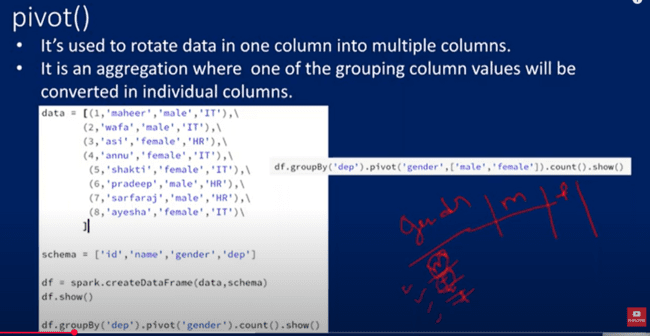
****

**30**

****

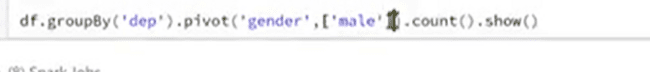


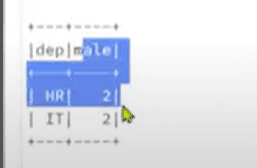
**31.**

****

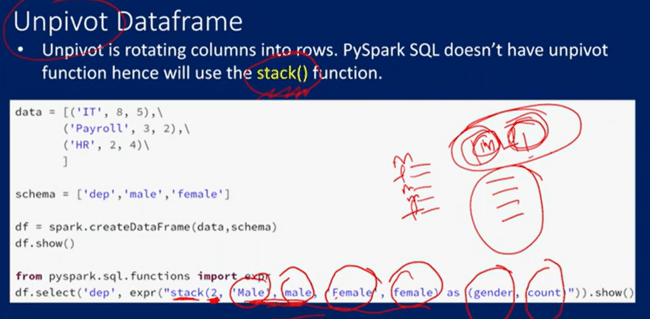


****

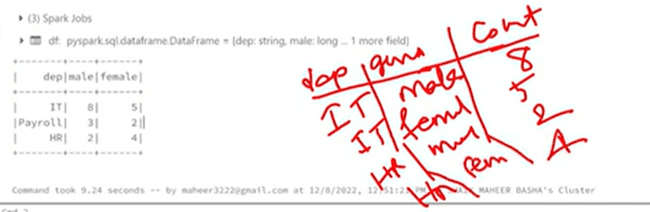
****

****

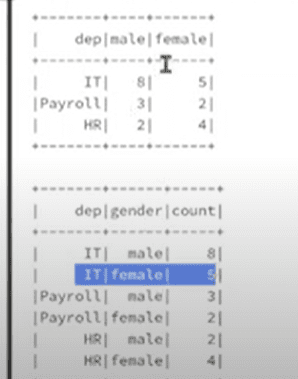
**32.**

****

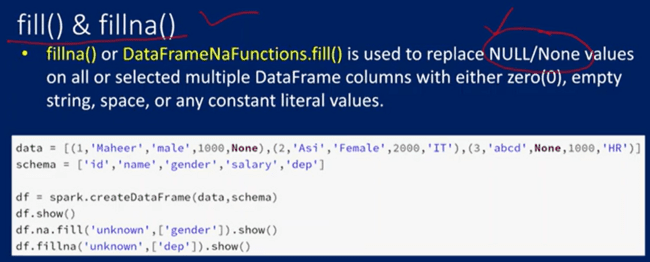


****



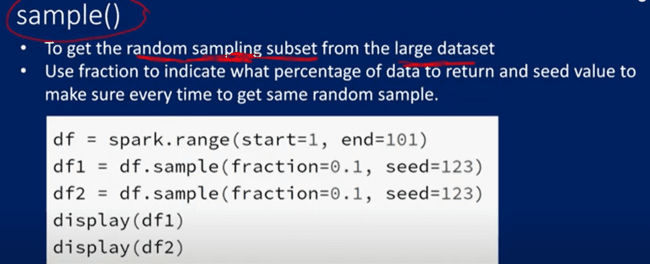
****

**33.**

****

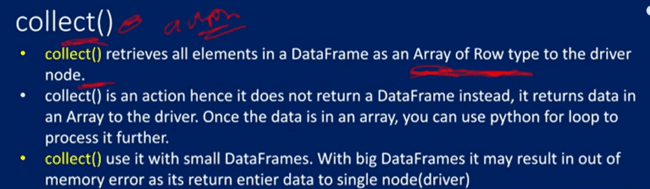
**🡪 entire dataframe**

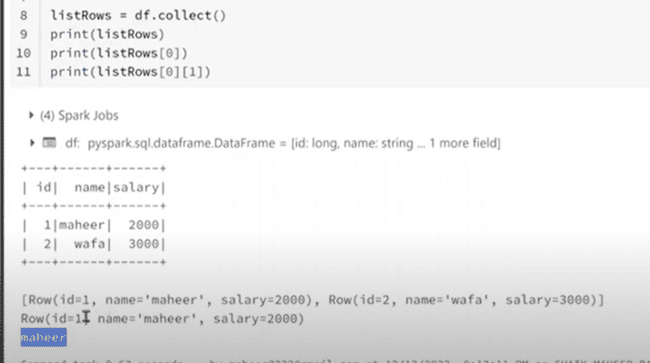
34.



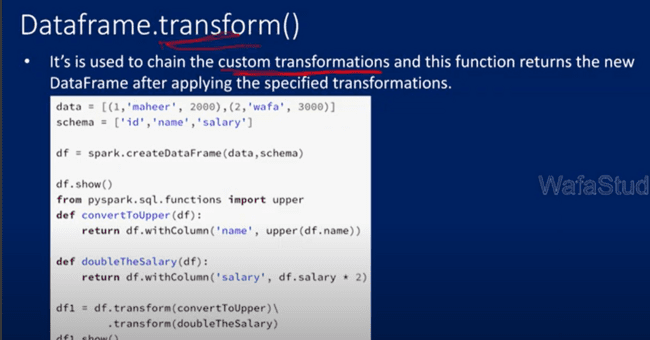


35.



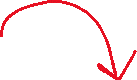


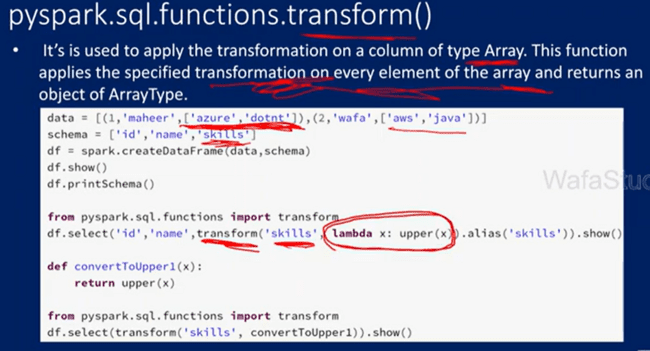
36.





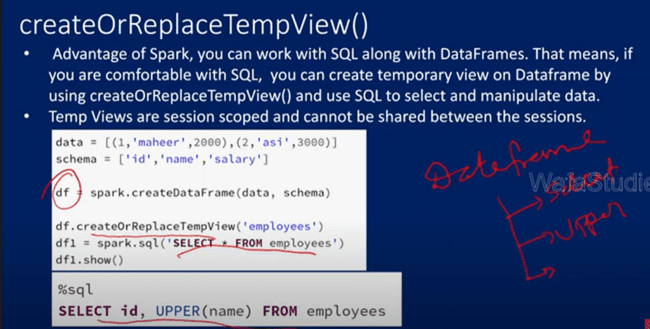
37.





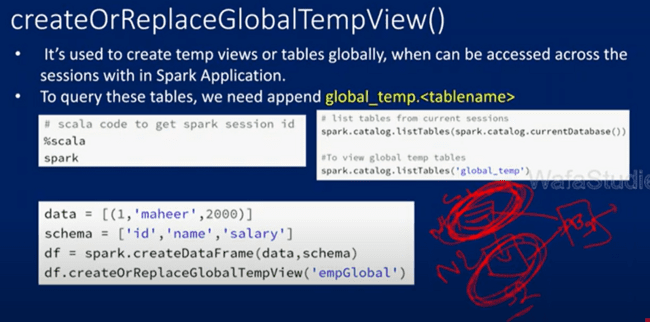


38. SqL 🡨- , these are at seession level only , for global view 🡪 39.



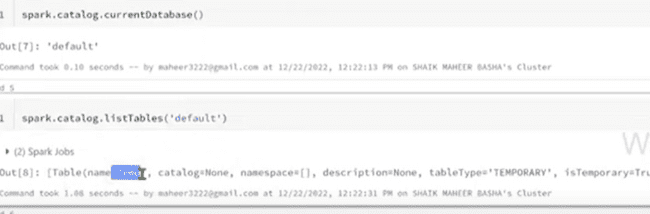


39. with global 🡪 we can use it another notebook

****



****

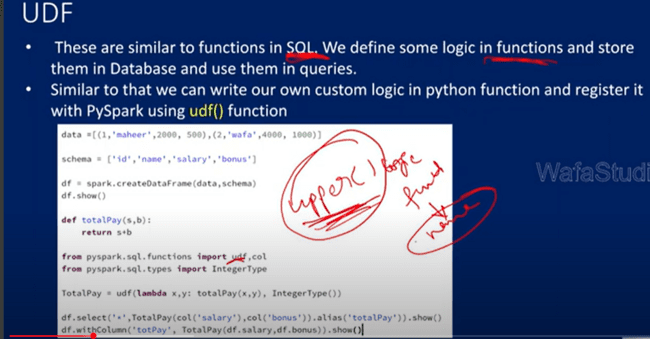
****

****

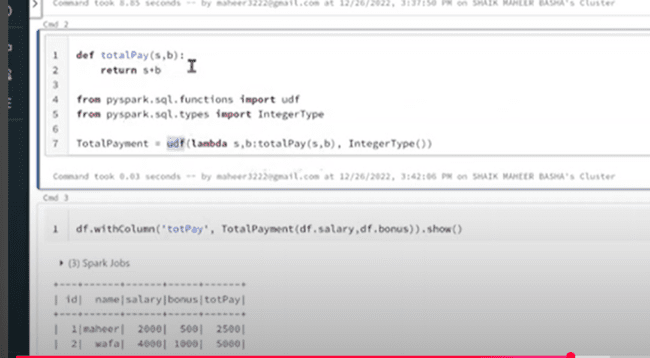
**Dropping**

****

**40.**

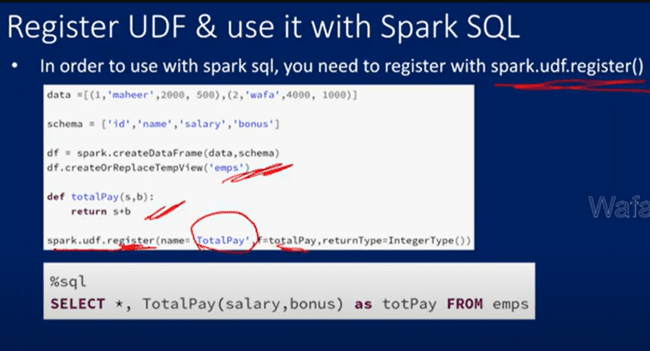
****



****

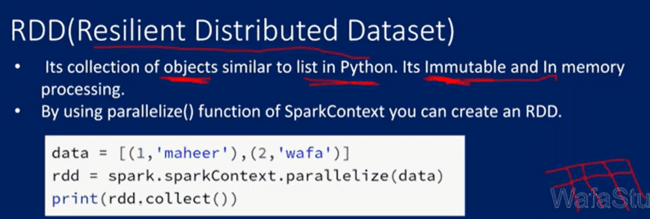
****



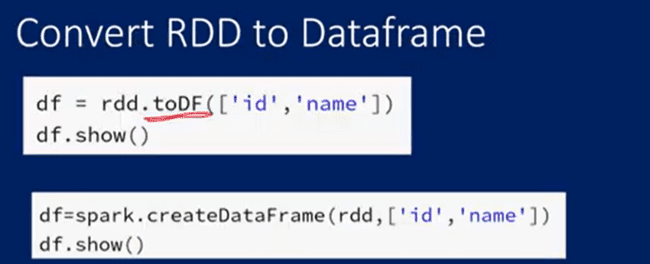
****



**41.**

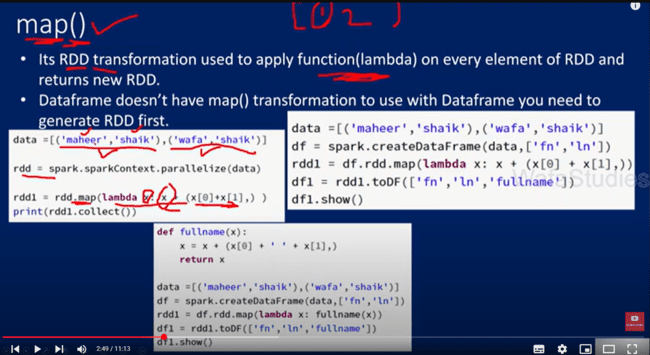
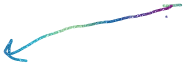
****



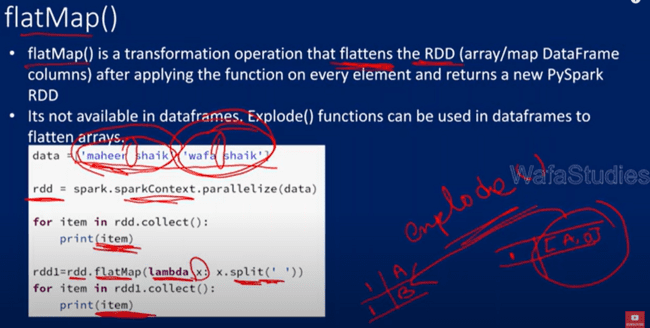
****

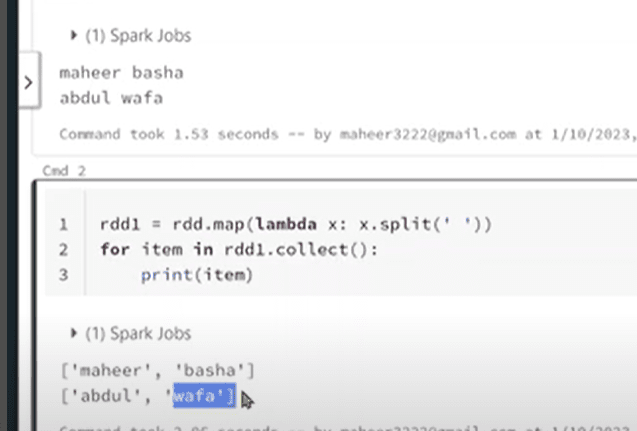


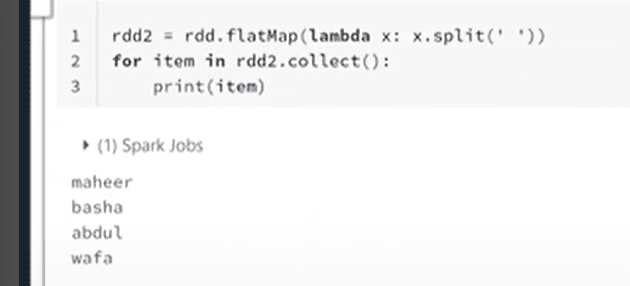
**42. rdd.map( lambda x:)**



43 .

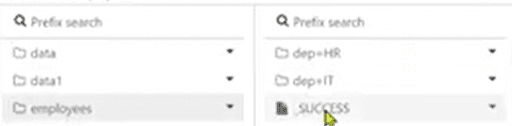


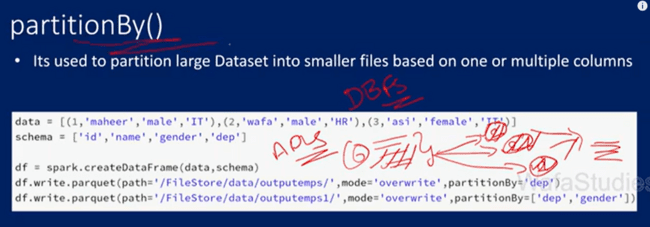




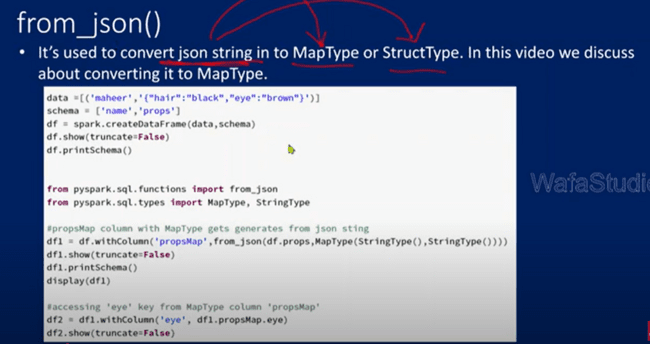
44. df.write() 🡪 will create a folder

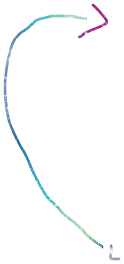






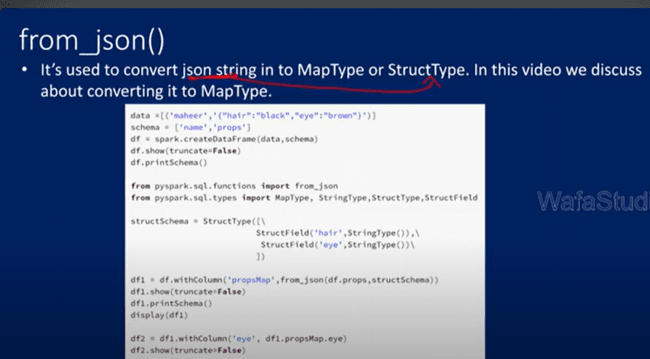
45. Now with Struct Type







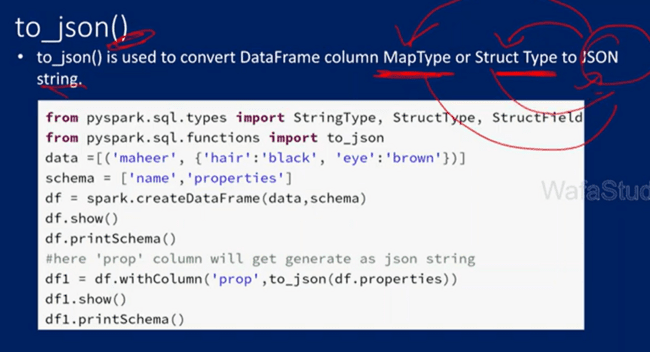
46.

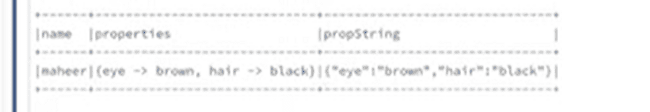






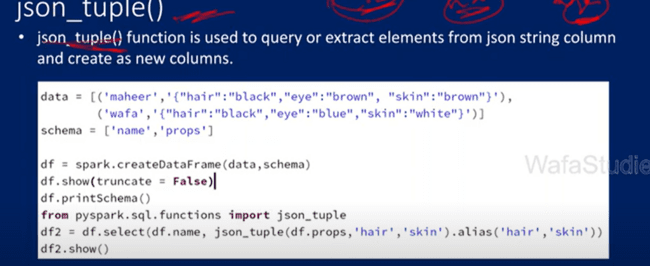
47.



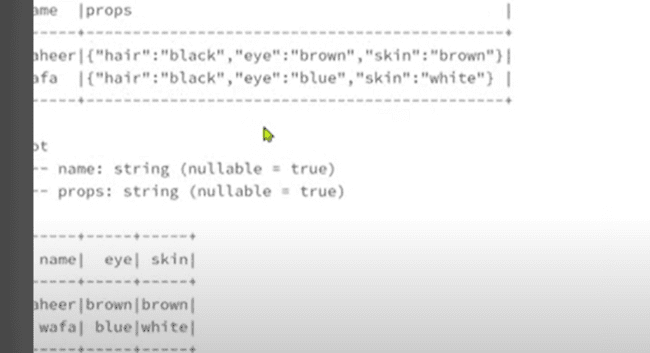




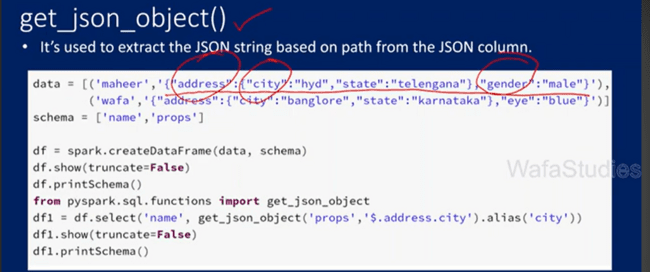


48. 



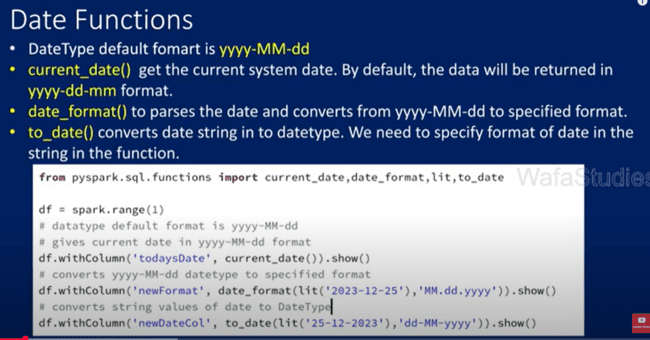


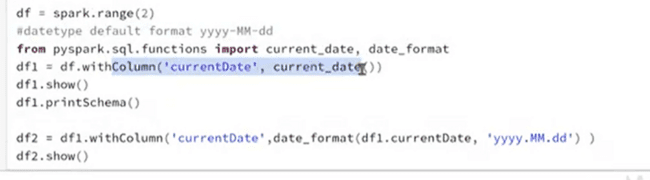
49.

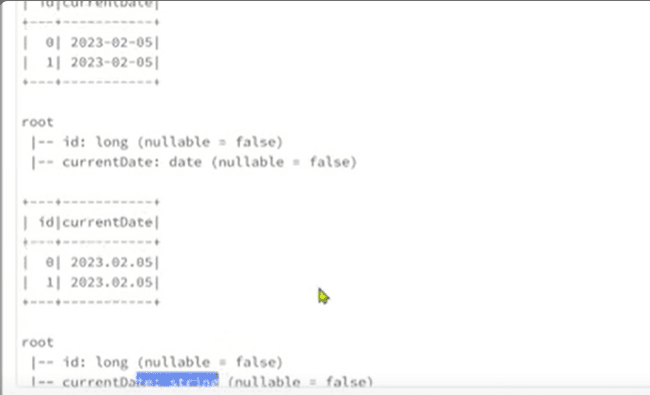




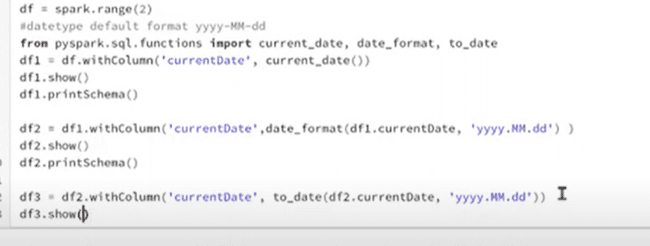
50.

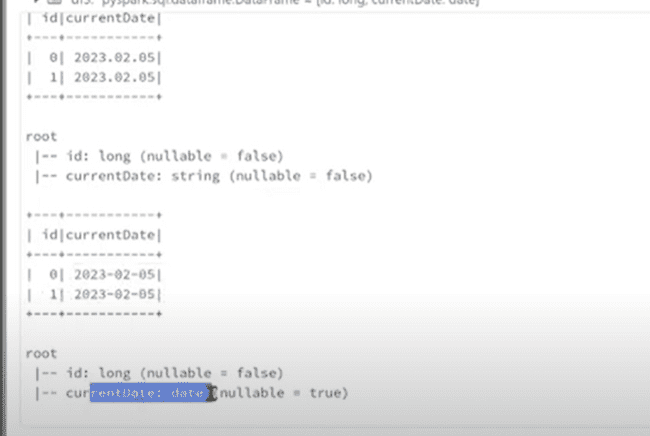




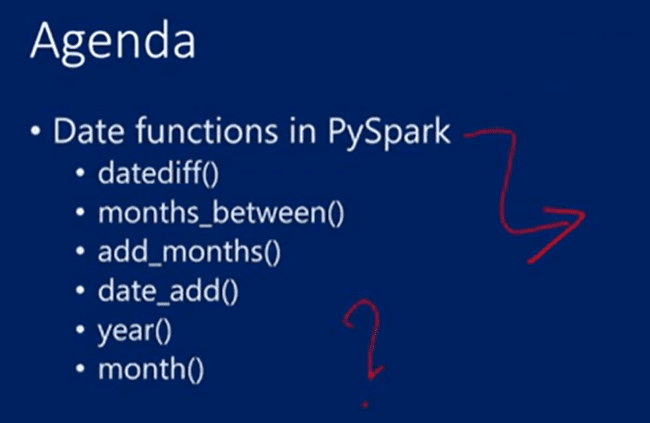




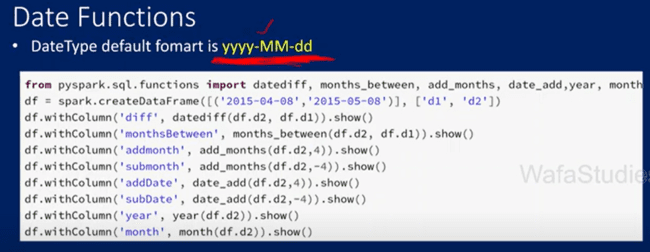




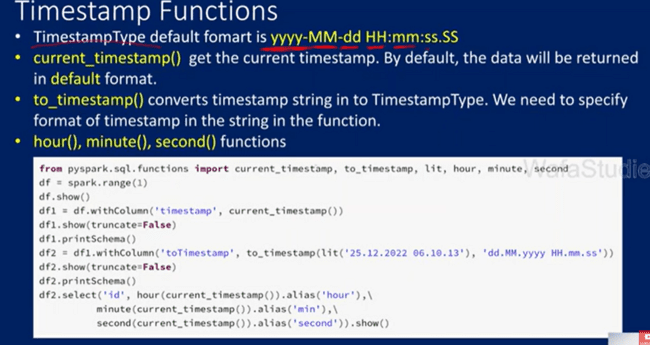
51



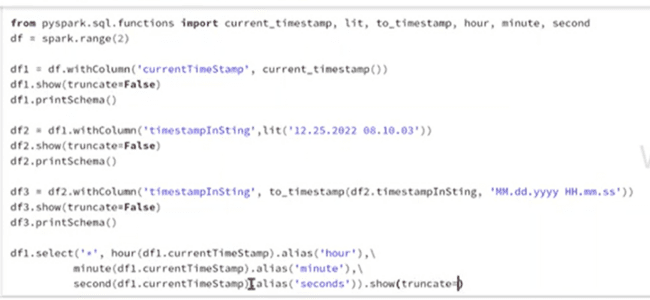




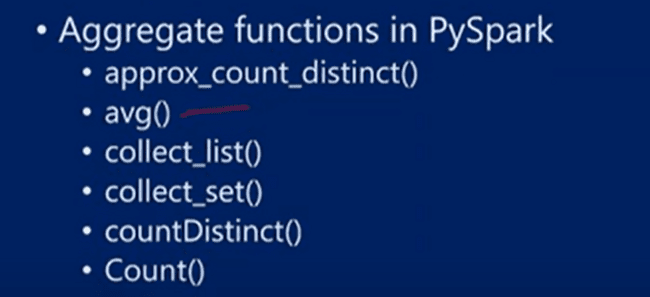
52.

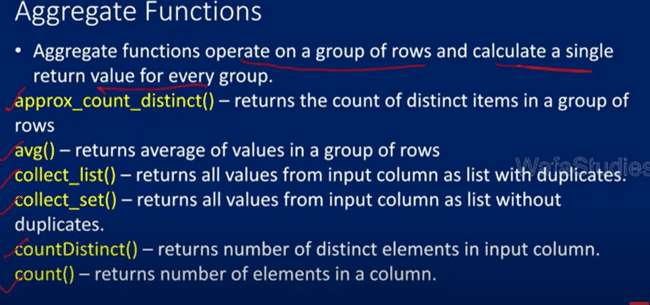


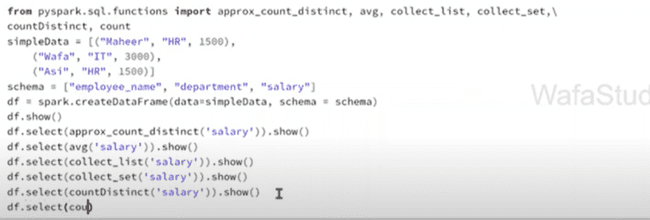




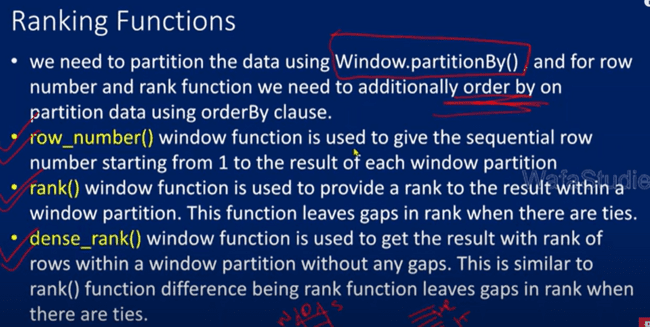
53.

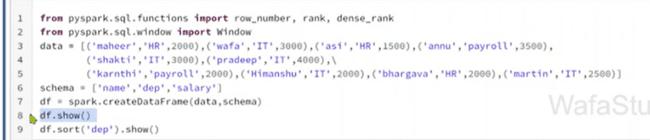




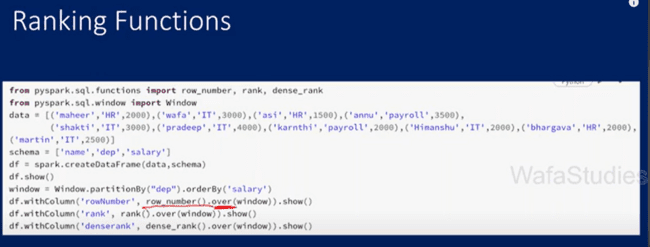


54.





First sort then rank



2nd output

