EFFICIENTLY JOIN TABLES IN SPARK: 7 WAYS OF JOINING TABLES

Gather

Insights

Together

CHALLENGE AT HAND

- INSIGHTS NEEDS TO BE CREATED FROM DATA STORED IN MULTIPLE TABLES
- FINDING RELATIONSHIPS BETWEEN ENTITIES, AND HOW THEY ARE RELATED
- REDUCING THE TIME TAKEN FOR CREATING DASHBOARDS
- INTRODUCE FLEXIBILITY IN COMBINING DATA USING INNER, AND OUTER JOIN
- ESTABLISH INTEGRITY OF THE DATA BY USING ONE TO MANY AND MANY TO MANY MAPS.
- THINKING ABOUT MULTIPLE WAYS OF AGGREGATING AND THEN COMBINING THE DATA

7 JOIN & PROBLEM IT SOLVES

INNER JOIN

- CREATES A ONE TO ONE MAP

LEFT JOIN

- ONE TO ONE MAP WITH DATA ON THE LEFT RELATION IS MAINTAINED

RIGHT JOIN

- ONE TO ONE MAP WITH DATA ON THE RIGHT RELATION IS MAINTAINED

FULL JOIN

- ONE TO ONE MAP WITH DATA ON THE BOTH RELATION IS MAINTAINED

SEMI JOIN

- ONE TO ONE MAP WITH DATA LEFT RETURNED WHERE IT MATCHES WITH RIGHT

ANTI JOIN

- RETURNS THE LEFT RELATION WHERE THERE IS NO MATCH WITH THE RIGHT

7 JOIN CLAUSES

INNER JOIN

- THE INNER JOIN IS THE DEFAULT JOIN IN SPARK SQL. IT SELECTS ROWS THAT HAVE MATCHING VALUES IN BOTH RELATIONS.
- RELATION [INNER] JOIN RELATION [JOIN_CRITERIA]

LEFT JOIN

- A LEFT JOIN RETURNS ALL VALUES FROM THE LEFT RELATION AND THE MATCHED VALUES FROM THE RIGHT RELATION, OR APPENDS NULL IF THERE IS NO MATCH. IT IS ALSO REFERRED TO AS A LEFT OUTER JOIN.
- RELATION LEFT [OUTER] JOIN RELATION [JOIN_CRITERIA]

RIGHT JOIN

- A RIGHT JOIN RETURNS ALL VALUES FROM THE RIGHT RELATION AND THE MATCHED VALUES FROM THE LEFT RELATION, OR APPENDS NULL IF THERE IS NO MATCH. IT IS ALSO REFERRED TO AS A RIGHT OUTER JOIN.
- RELATION RIGHT [OUTER] JOIN RELATION [JOIN_CRITERIA]

FULL JOIN

- A FULL JOIN RETURNS ALL VALUES FROM BOTH RELATIONS, APPENDING NULL VALUES ON THE SIDE THAT DOES NOT HAVE A MATCH. IT IS ALSO REFERRED TO AS A FULL OUTER JOIN.
- RELATION FULL [OUTER] JOIN RELATION [JOIN_CRITERIA]

CROSS JOIN

- A CROSS JOIN RETURNS THE CARTESIAN PRODUCT OF TWO RELATIONS.
- RELATION CROSS JOIN RELATION [JOIN_CRITERIA]

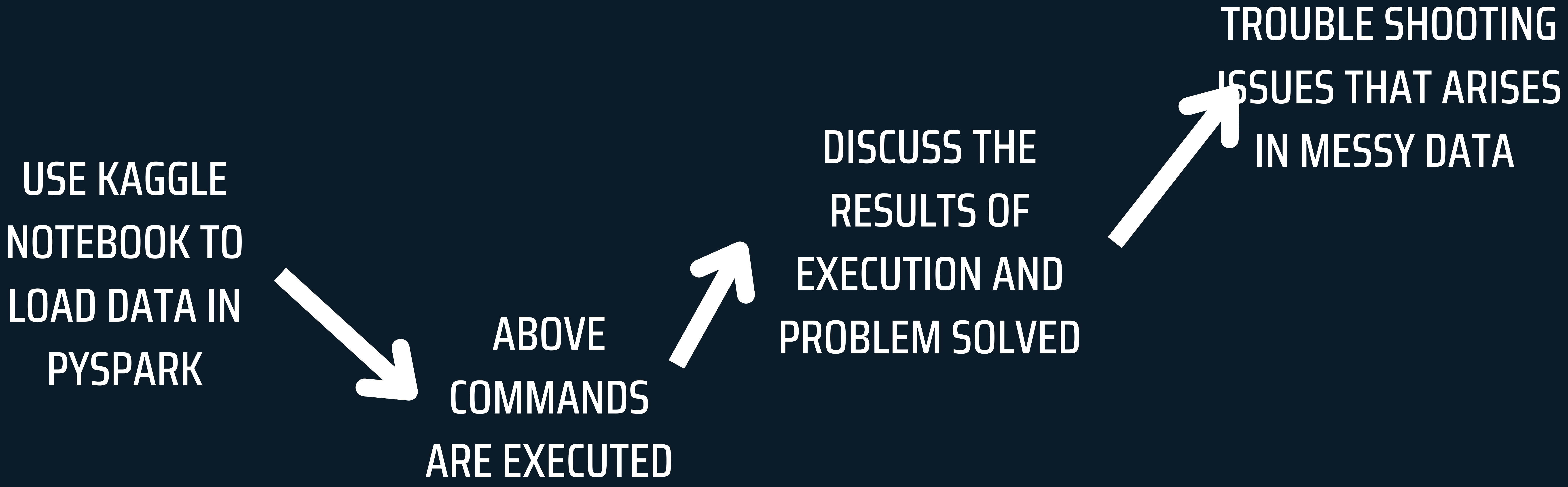
SEMI JOIN

- A SEMI JOIN RETURNS VALUES FROM THE LEFT SIDE OF THE RELATION THAT HAS A MATCH WITH THE RIGHT. IT IS ALSO REFERRED TO AS A LEFT SEMI JOIN.
- RELATION [LEFT] SEMI JOIN RELATION [JOIN_CRITERIA]

ANTI JOIN

- AN ANTI JOIN RETURNS VALUES FROM THE LEFT RELATION THAT HAS NO MATCH WITH THE RIGHT. IT IS ALSO REFERRED TO AS A LEFT ANTI JOIN.

HOW WE ARE DOING IT?



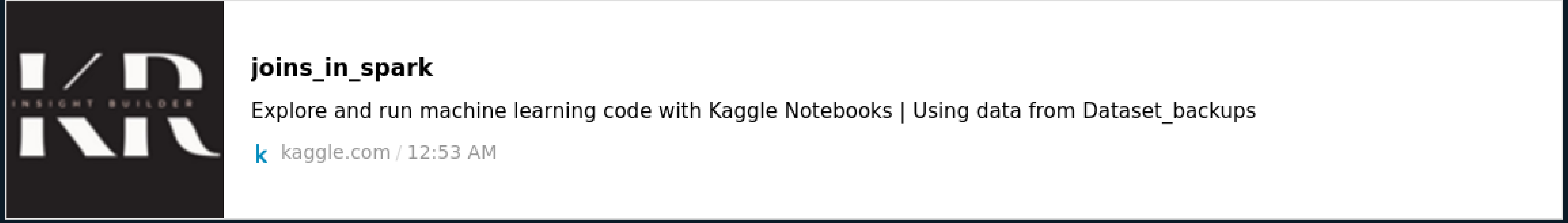
USE KAGGLE NOTEBOOK TO LOAD DATA IN PYSPARK

ABOVE COMMANDS ARE EXECUTED

DISCUSS THE RESULTS OF EXECUTION AND PROBLEM SOLVED

TROUBLE SHOOTING ISSUES THAT ARISES IN MESSY DATA

LETS GET OURSELF A PYSPARK NOTEBOOK AND DIG IN



joins_in_spark

Explore and run machine learning code with Kaggle Notebooks | Using data from Dataset_backups

k kaggle.com / 12:53 AM

REAL CLUSTER IS NOT NECESSARY FOR LEARNING THE DML

THANKS FOR WATCHING

PRACTICE

PRACTICE

♥ LIKE

SHARE

SUBSCRIBE

PRACTICE

PRACTICE

HTTPS://GITHUB.COM/INSIGHTBUILDER