## join Transformation When called on datasets of type (K, V) and (K, W), returns a dataset of (K, (V, W)) pairs with all pairs of elements for each key. Outer joins are supported through leftOuterJoin, rightOuterJoin, and fullOuterJoin. Below, these are explored. syntax of join for DataFrame - join(dataset,joinExprs,joinType) In [1]: from pyspark.sql import SparkSession spark = SparkSession \ .builder \ .master("local[2]") \ .appName("join Transformation") \ .enableHiveSupport() \ .getOrCreate() #key-value pairs from dataset 1 kvPair = [(1,"v"), (1,"i"), (2, "s"), (3,"h"), (4,"a")]kvPairRDD = spark.sparkContext.parallelize(kvPair) print(kvPairRDD.collect()) #key-value pairs from dataset 2 otherKvPair = [(1,"y"), (1,"l"), (2, "d"), (3,"m"), (8,"z2")]otherKvPairRDD = spark.sparkContext.parallelize(otherKvPair) print(otherKvPairRDD.collect()) #join() transformation to join both datasets to return (K, (V,W)) joined = kvPairRDD.join(otherKvPairRDD) print("joined.count(): ", joined.count()) print("joined.collect(): ", joined.collect()) 22/10/14 00:33:10 WARN Utils: Your hostname, Vaishalis-MacBook-Pro.local resolves to a loopback address: 127.0. 0.1; using 192.168.0.105 instead (on interface en0) 22/10/14 00:33:10 WARN Utils: Set SPARK\_LOCAL\_IP if you need to bind to another address Setting default log level to "WARN". To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel). 22/10/14 00:33:11 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builti n-java classes where applicable [(1, 'v'), (1, 'i'), (2, 's'), (3, 'h'), (4, 'a')] [(1, 'y'), (1, 'l'), (2, 'd'), (3, 'm'), (8, 'z2')]joined.count(): 6 joined.collect(): [(1, ('v', 'y')), (1, ('v', 'l')), (1, ('i', 'y')), (1, ('i', 'l')), (2, ('s', 'd')), (3, ('h', 'm'))] In [2]: #Let us look at another example to understand various joinType data1 = [(1,"John",-1,"2019","10","M"), \ (2,"Alex",1,"2015","20","M"), \ (3,"Williams",1,"2016","10","M"), \ (4,"Arjun",2,"2010","10","F"), \ (5,"Brown",2,"2012","40",""), \ (6,"Brown",2,"2012","50","") \ ] df1 = spark.createDataFrame(data= data1, schema = data1\_columns) df1.printSchema() df1.show(truncate=False) root -- employee\_id: long (nullable = true) |-- name: string (nullable = true) |-- superior\_emp\_id: long (nullable = true) |-- year\_joined: string (nullable = true) |-- emp\_dept\_id: string (nullable = true) |-- gender: string (nullable = true) |employee\_id|name |superior\_emp\_id|year\_joined|emp\_dept\_id|gender| |1 |John |2019 |2 |Alex 11 12015 120 |M |M |3 |Williams|1 |2016 |10 |4 |2010 |F |Arjun |2 |10 |5 Brown |2 |2012 |40 |6 Brown |2 |2012 |50 In [3]: data2 = [("Finance",10), \ ("Marketing",20), \ ("Sales",30), \ ("IT",40)] data2\_columns = ["dept\_name","dept\_id"] df2 = spark.createDataFrame(data = data2, schema = data2\_columns) df2.printSchema() df2.show(truncate=False) root |-- dept\_name: string (nullable = true) |-- dept\_id: long (nullable = true) |dept\_name|dept\_id| |Finance | 10 |Marketing|20 |Sales |30 |40 |IT In [4]: df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"inner") \ .show(truncate=False) |employee\_id|name |superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id| |1 |John |2019 |10 |Finance | 10 |3 |Williams|1 |2016 |10 M |Finance |10 |4 |Arjun |2 |2010 |10 |F |Finance | 10 |2 |Alex |1 |2015 |20 |M |Marketing|20 |5 Brown |2 |2012 |40 |40 In [5]: df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"cross") \ .show(truncate=False) |employee\_id|name |superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id| |1 |John |-1 |2019 |10 |M |Finance | 10 |3 |Williams|1 |2016 |10 |M |Finance | 10 |2 |2010 |F |Finance | 10 |4 |Arjun |10 |2 |Alex |1 |2015 |20 ΙM |Marketing|20 |5 Brown |2 |2012 |40 |IT |40 df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"outer") \ .show(truncate=False) |employee\_id|name |superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id| |1 John |2019 |10 |Finance 13 |Williams|1 12016 110 ΙM **IFinance** 110

data1\_columns = ["employee\_id", "name", "superior\_emp\_id", "year\_joined", "emp\_dept\_id", "gender"]

12010 |4 |Arjun |10 |F |Finance |10 |2 |Alex |Marketing|20 |2 |1 |2015 |20 |M |null |null Inull Inull |null |null |Sales |5 Brown |2 |2012 |40 |IT |40 |6 |2 |2012 |50 Brown |null |null In [7]: df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"fullouter") \

.show(truncate=False) |employee\_id|name |superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id| |1 |John |2019 |10 ΙM |Finance 110 |Williams|1 |3 |2016 |10 M |Finance |10 ۱F |4 |Arjun |2 |2010 |10 |Finance |10 |2 |Alex |1 |2015 |20 |Marketing|20 130 |null |null |null |null null |Sales |null |2012 140 |5 Brown 12 |40 |IT |6 Brown |2 |2012 |50 |null |null In [8]: df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"left") \

.show(truncate=False) |superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id| |employee\_id|name |1 |John |2019 |10 |Finance 13 |Williams|1 |2016 |10 |Finance |M |10 |2 |Alex 11 |2015 |20 |M |Marketing|20 |2 |50 |null

|6 Brown |2012 |null |4 |Arjun 12 |2010 |10 | F |Finance |10

|40

|superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id|

|superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id|

|10

|10

|10

|20

|40

|null

|superior\_emp\_id|year\_joined|emp\_dept\_id|gender|dept\_name|dept\_id|

|10

|10

|10

120

|40

|10

|10

|10

|20

|40

|50

|null

|M

ΙM

ΙM

|F

|F

M

M

|F

M

ΙM

|M

M

M

|F

|M

|null

null

|10

|10

120

|50

|10

|40

|IT

|Finance

|Finance

|Finance

|null

|IT

|Marketing|20

|Finance | 10

|Marketing|20

|10

130

|40

|10

|10

|10

|30

|40

|Finance

|Finance

|Finance

|Finance

|Finance

|Sales

IIT

|Marketing|20

|Sales

|IT

|null

|10

|40

|40

|2012

|2019

|2016

|2015

|2012

2010

2012

|2010

|2016

|2019

2015

null

2012

|2010

|2016

|2019

|2015

null

2012

|2019

|2016

|2010

|2015

|2012

|employee\_id|name |superior\_emp\_id|year\_joined|emp\_dept\_id|gender|

|2012

|superior\_emp\_id|year\_joined|emp\_dept\_id|gender|

|5

|1

|3

|2

|6

|4

15

|4

|3

|1

|2

|5

14

|3

|1

|2

15

|1

|3

|4

|2

|5

|6

|null

|null

Brown

.show(truncate=False)

John

|Alex

Brown

|Arjun

Brown

.show(truncate=False)

|Arjun

|John

|Alex

null

Brown

.show(truncate=False)

|Arjun

John

|Alex

|null

Brown

.show(truncate=False)

|John

|Arjun

|Alex

Brown

.show(truncate=False)

|Brown|2

|Williams|1

12

|1

|2

In [13]: df1.join(df2, df1.emp dept id == df2.dept id,"leftanti") \

|Williams|1

|Williams|1

|Williams|1

|1

|2

|2

12

In [10]: df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"right") \

|2

|-1

|1

|2

12

1 - 1

|1

12

|null

df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"leftsemi") \

|null

df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"rightouter") \

|employee\_id|name

|employee\_id|name

|employee\_id|name

|employee\_id|name

|2

df1.join(df2, df1.emp\_dept\_id == df2.dept\_id,"leftouter") \