Complete PySpark RDD Methods Reference Guide

🔄 TRANSFORMATIONS (Lazy - Return new RDD)

Basic Transformations

Method	Returns	Description	Example
`map(func)`	RDD	Apply function to each element	`rdd.map(lambda x: x*2)`
`filter(func)`	RDD	Filter elements by condition	`rdd.filter(lambda x: x > 5)`
`flatMap(func)`	RDD	Apply function and flatten result	`rdd.flatMap(lambda x: x.split())`
`distinct()`	RDD	Remove duplicate elements	`rdd.distinct()`
`sample(withReplacement, fraction, seed)`	RDD	Random sample of elements	`rdd.sample(False, 0.1)`

Set Operations

Method	Returns	Description	Example
`union(other)`	RDD	Union with another RDD	`rdd1.union(rdd2)`
`intersection(other)`	RDD	Common elements between RDDs	`rdd1.intersection(rdd2)`
`subtract(other)`	RDD	Elements in this RDD but not other	`rdd1.subtract(rdd2)`
`cartesian(other)`	RDD	Cartesian product with another RDD	`rdd1.cartesian(rdd2)`

Key-Value Transformations (for Pair RDDs)

Method	Returns	Description	Example
`mapValues(func)`	RDD	Apply function only to values	`pair_rdd.mapValues(lambda x: x*2)`
`keys()`	RDD	Extract keys only	`pair_rdd.keys()`
`values()`	RDD	Extract values only	`pair_rdd.values()`
`reduceByKey(func)`	RDD	Reduce values by key	`pair_rdd.reduceByKey(lambda a,b: a+b)`

Method	Returns	Description	Example
groupByKey()	RDD	Group values by key	pair_rdd.groupByKey()
sortByKey(ascending)	RDD	Sort by key	pair_rdd.sortByKey()
join(other)	RDD	Inner join on keys	rdd1.join(rdd2)
leftOuterJoin(other)	RDD	Left outer join	rdd1.leftOuterJoin(rdd2)
rightOuterJoin(other)	RDD	Right outer join	rdd1.rightOuterJoin(rdd2)
fullOuterJoin(other)	RDD	Full outer join	rdd1.fullOuterJoin(rdd2)
cogroup(other)	RDD	Group together values from both RDDs	rdd1.cogroup(rdd2)
subtractByKey(other)	RDD	Remove elements with keys in other	rdd1.subtractByKey(rdd2)

Partitioning & Ordering

Method	Returns	Description	Example
partitionBy(numPartitions, partitioner)	RDD	Custom partitioning	rdd.partitionBy(4)
repartition(numPartitions)	RDD	Repartition with shuffle	rdd.repartition(8)
coalesce(numPartitions)	RDD	Reduce partitions without shuffle	rdd.coalesce(2)
sortBy(func, ascending)	RDD	Sort by function result	rdd.sortBy(lambda x: x[1])
randomSplit(weights, seed)	List[RDD]	Split RDD randomly	rdd.randomSplit([0.7, 0.3])

Advanced Transformations

Method	Returns	Description	Example
mapPartitions(func)	RDD	Apply function to each partition	rdd.mapPartitions(process_partition)
mapPartitionsWithIndex(func)	RDD	Map partitions with index	rdd.mapPartitionsWithIndex(func)
glom()	RDD	Convert each partition to	rdd.glom()

Method	Returns	Description array	Example
`pipe(command)`	RDD	Pipe through external command	`rdd.pipe("grep pattern")`
`keyBy(func)`	RDD	Create key-value pairs	`rdd.keyBy(lambda x: x[0])`
`zipWithIndex()`	RDD	Zip with element indices	`rdd.zipWithIndex()`
`zipWithUniqueId()`	RDD	Zip with unique IDs	`rdd.zipWithUniqueId()`
`zip(other)`	RDD	Zip with another RDD	`rdd1.zip(rdd2)`

⚡ ACTIONS (Eager - Trigger computation, return values to driver)

Collection Actions

Method	Returns	Description	Example
`collect()`	List	Return all elements as list	`rdd.collect()`
`take(n)`	List	Return first n elements	`rdd.take(10)`
`takeOrdered(n, key)`	List	Return n smallest elements	`rdd.takeOrdered(5, key=lambda x: x)`
`top(n, key)`	List	Return n largest elements	`rdd.top(5)`
`takeSample(withReplacement, n, seed)`	List	Return random sample	`rdd.takeSample(False, 10)`
`first()`	Element	Return first element	`rdd.first()`

Aggregation Actions

Method	Returns	Description	Example
`reduce(func)`	Element	Reduce elements using function	`rdd.reduce(lambda a,b: a+b)`
`fold(zeroValue, func)`	Element	Fold with initial value	`rdd.fold(0, lambda a,b: a+b)`
`aggregate(zeroValue, seqOp, combOp)`	Element	General aggregation	`rdd.aggregate(0, add, add)`
`sum()`	Number	Sum of elements	`rdd.sum()`
`count()`	Int	Number of elements	`rdd.count()`
`countByValue()`	Dict	Count occurrences of each value	`rdd.countByValue()`
`countByKey()`	Dict	Count occurrences by key	`pair_rdd.countByKey()`
`max()`	Element	Maximum element	`rdd.max()`
`min()`	Element	Minimum element	`rdd.min()`
`mean()`	Float	Average of elements	`rdd.mean()`
`variance()`	Float	Variance of elements	`rdd.variance()`
`stdev()`	Float	Standard deviation	`rdd.stdev()`
`stats()`	StatCounter	Statistical summary	`rdd.stats()`
`histogram(buckets)`	Tuple	Histogram of values	`rdd.histogram(10)`

Key-Value Actions (for Pair RDDs)

Method	Returns	Description	Example
`collectAsMap()`	Dict	Collect as dictionary	`pair_rdd.collectAsMap()`
`lookup(key)`	List	Values for specific key	`pair_rdd.lookup("key1")`
`countByKey()`	Dict	Count by key	`pair_rdd.countByKey()`

Output Actions

Method	Returns	Description	Example
`saveAsTextFile(path)`	None	Save as text files	`rdd.saveAsTextFile("output/")`

Method	Returns	Description	Example
saveAsPickleFile(path)	None	Save as pickle files	rdd.saveAsPickleFile("output.pkl")
saveAsSequenceFile(path)	None	Save as sequence files	rdd.saveAsSequenceFile("output/")
foreach(func)	None	Apply function to each element	rdd.foreach(print)
foreachPartition(func)	None	Apply function to each partition	rdd.foreachPartition(process)

Boolean Actions

Method	Returns	Description	Example
isEmpty()	Boolean	Check if RDD is empty	rdd.isEmpty()

🔧 UTILITY METHODS (Metadata & Control)

RDD Information

Method	Returns	Description	Example
getNumPartitions()	Int	Number of partitions	rdd.getNumPartitions()
partitions()	List	List of partitions	rdd.partitions()
partitioner()	Partitioner	Partitioner object	rdd.partitioner()
glom()	RDD[List]	Show partition contents	rdd.glom().collect()
toDebugString()	String	Debug information	rdd.toDebugString()

Caching & Persistence

Method	Returns	Description	Example
cache()	RDD	Cache in memory	rdd.cache()
persist(storageLevel)	RDD	Persist with storage level	rdd.persist(MEMORY_AND_DISK)
unpersist(blocking)	RDD	Remove from cache	rdd.unpersist()
checkpoint()	None	Checkpoint RDD	rdd.checkpoint()
localCheckpoint()	RDD	Local checkpoint	rdd.localCheckpoint()
isCheckpointed()	Boolean	Check if checkpointed	rdd.isCheckpointed()

Method	Returns	Description	Example
`getCheckpointFile()`	String	Get checkpoint file	`rdd.getCheckpointFile()`

Dependencies & Lineage

Method	Returns	Description	Example
`dependencies()`	List	RDD dependencies	`rdd.dependencies()`
`context()`	SparkContext	Get SparkContext	`rdd.context()`
`name()`	String	RDD name	`rdd.name()`
`setName(name)`	RDD	Set RDD name	`rdd.setName("my_rdd")`
`id()`	Int	Unique RDD ID	`rdd.id()`

📊 STORAGE LEVELS (for persist())

Storage Level	Memory	Disk	Serialized	Replicated
`MEMORY_ONLY`	✓	✗	✗	✗
`MEMORY_AND_DISK`	✓	✓	✗	✗
`MEMORY_ONLY_SER`	✓	✗	✓	✗
`MEMORY_AND_DISK_SER`	✓	✓	✓	✗
`DISK_ONLY`	✗	✓	✗	✗
`MEMORY_ONLY_2`	✓	✗	✗	✓
`MEMORY_AND_DISK_2`	✓	✓	✗	✓

🎯 METHOD CATEGORIES SUMMARY

🔄 Transformations (Lazy)

- **Element-wise**: `map`, `filter`, `flatMap`, `distinct`
- **Set operations**: `union`, `intersection`, `subtract`
- **Key-Value**: `reduceByKey`, `groupByKey`, `join`, `mapValues`
- **Repartitioning**: `repartition`, `coalesce`, `partitionBy`
- **Sorting**: `sortBy`, `sortByKey`

⚡ Actions (Eager)

- **Collection**: `collect`, `take`, `first`, `takeOrdered`
- **Aggregation**: `reduce`, `count`, `sum`, `mean`, `max`, `min`
- **Counting**: `countByValue`, `countByKey`
- **Output**: `saveAsTextFile`, `foreach`
- **Lookup**: `lookup` (for pair RDDs)

🔧 Utilities

- **Info**: `count`, `getNumPartitions`, `toDebugString`
- **Caching**: `cache`, `persist`, `unpersist`
- **Checkpointing**: `checkpoint`, `isCheckpointed`

💡 Performance Tips

1. **Use transformations over actions** when possible (lazy evaluation)
2. **Cache frequently used RDDs** with `cache()` or `persist()`
3. **Avoid** `collect()` on large datasets
4. **Use** `reduceByKey` **over** `groupByKey` for aggregations
5. **Partition appropriately** - too few = underutilization, too many = overhead
6. **Use** `coalesce` **over** `repartition` when reducing partitions