

```
In [1]: import findspark
        from pyspark.streaming import StreamingContext
        from pyspark.sql import SparkSession
        import math

        findspark.init()
```

```
In [82]: try: ssc.stop(True, True)
        except: pass
        try: spark.stop()
        except: pass
```

```
printreduceByKeyAndWindow []
-----
Time: 2021-09-19 00:55:40
-----

printreduceByKeyAndWindow []
-----
Time: 2021-09-19 00:55:41
-----

printreduceByKeyAndWindow []
-----
Time: 2021-09-19 00:55:42
-----
abc

printreduceByKeyAndWindow [('abc', 1)]
-----
Time: 2021-09-19 00:55:43
-----
abc
def

printrddwindow [('abc', 1), ('abc', 1), ('def', 1)]
printreddcountByWindow [3]
printreduceByKeyAndWindow [('abc', 2), ('def', 1)]
printreddcountByValueAndWindow [([('abc', 1), 2), ('def', 1), 1]]
-----
Time: 2021-09-19 00:55:44
-----

printreduceByKeyAndWindow [('abc', 2), ('def', 1)]
-----
Time: 2021-09-19 00:55:45
-----
def

printreduceByKeyAndWindow [('abc', 2), ('def', 2)]
-----
Time: 2021-09-19 00:55:46
-----
ghi

printreduceByKeyAndWindow [('abc', 2), ('def', 2), ('ghi', 1)]
-----
Time: 2021-09-19 00:55:47
-----

-----
Time: 2021-09-19 00:55:48
```

frg

printrddwindow [('abc', 1), ('abc', 1), ('def', 1), ('def', 1), ('ghi', 1), ('frg', 1)]
printrddcountByWindow [6]
printrddcountByValueAndWindow [(('abc', 1), 2), (('def', 1), 2), (('ghi', 1), 1), (('frg', 1), 1)]

Time: 2021-09-19 00:55:49

Time: 2021-09-19 00:55:50

ghi

Time: 2021-09-19 00:55:51

Time: 2021-09-19 00:55:52

abc

Time: 2021-09-19 00:55:53

printrddwindow [('def', 1), ('ghi', 1), ('frg', 1), ('ghi', 1), ('abc', 1)]
printrddcountByWindow [5]
printrddcountByValueAndWindow [(('def', 1), 1), (('ghi', 1), 2), (('frg', 1), 1), (('abc', 1), 1)]

Time: 2021-09-19 00:55:54

def

Time: 2021-09-19 00:55:55

Time: 2021-09-19 00:55:56

frg

Time: 2021-09-19 00:55:57

Time: 2021-09-19 00:55:58

printrddwindow [('ghi', 1), ('abc', 1), ('def', 1), ('frg', 1)]
printrddcountByWindow [4]
printrddcountByValueAndWindow [(('ghi', 1), 1), (('abc', 1), 1), (('def', 1), 1), (('frg', 1), 1)]

Time: 2021-09-19 00:55:59

Time: 2021-09-19 00:56:00

```

pmj
-----
Time: 2021-09-19 00:56:01
-----

-----
Time: 2021-09-19 00:56:02
-----
abc

-----
Time: 2021-09-19 00:56:03
-----
def

-----
printrddwindow [('def', 1), ('frg', 1), ('pmj', 1), ('abc', 1), ('def', 1)]
printrddcountByWindow [5]
printrddcountByValueAndWindow [ (('def', 1), 2), (('frg', 1), 1), (('pmj', 1), 1), (('abc', 1), 1)]
-----
Time: 2021-09-19 00:56:04
-----
ghi

-----
Time: 2021-09-19 00:56:05
-----

-----
Time: 2021-09-19 00:56:06
-----

-----
Time: 2021-09-19 00:56:07
-----

-----
Time: 2021-09-19 00:56:08
-----

-----
printrddwindow [('pmj', 1), ('abc', 1), ('def', 1), ('ghi', 1)]
printrddcountByWindow [4]
printrddcountByValueAndWindow [ (('pmj', 1), 1), (('abc', 1), 1), (('def', 1), 1), (('ghi', 1), 1)]

```

```

In [77]: spark=SparkSession.builder.appName("SparkStreaming-04").master('local[1]').getOrCreate()
         sc=spark.sparkContext
         # .config("spark.driver.allowMultipleContexts","true")
         spark

```

Out[77]: **SparkSession - in-memory**

SparkContext

[Spark UI](#)

Version	v2.4.8
Master	local[1]
AppName	SparkStreaming-04

```

In [78]: ssc=StreamingContext(sc, 1)

```

```
ssc.checkpoint('checkpoint-04')
ssc # 1=1 second
```

Out[78]: <pyspark.streaming.context.StreamingContext at 0x24023ce5f48>

```
In [79]: lines=ssc.socketTextStream("localhost", 8000)
lines.pprint()
```

```
In [80]: lines=lines.flatMap(lambda x:x.split(" ")).map(lambda x:(x,1))
# lines.window(30,10).pprint()

def printrddwindow(rdd):
    print("printrddwindow", rdd.collect())

def printrddcountByWindow(rdd):
    print("printrddcountByWindow", rdd.collect())

def printreduceByKeyAndWindow(rdd):
    print("printreduceByKeyAndWindow", rdd.collect())

def printrddcountByValueAndWindow(rdd):
    print("printrddcountByValueAndWindow", rdd.collect())

# slideDuration print after seconds
# windowDuration discard old elements registered before this time frame
lines.window(10,5).foreachRDD(printrddwindow)
lines.countByWindow(10,5).foreachRDD(printrddcountByWindow)
lines.reduceByKeyAndWindow(lambda x,y: x+y, 10, 5).foreachRDD(printreduceByKeyAndWindow)
lines.countByValueAndWindow(10,5).foreachRDD(printrddcountByValueAndWindow)
```

```
In [81]: ssc.start()
```

```
-----
Time: 2021-09-19 00:55:39
-----
```

In []:

In []: