



- 首先下载对应的词文件，移动到 `spark/data` 中
- 执行如下命令，使用预设的 `wordCount` 类

- `./spark-submit --name "JavaWordCount" --executor-memory 4g --class org.apache.spark.examples.JavaWordCount ../examples/jars/spark-examples_2.12-3.2.4.jar ../data/sample-2mb-text-file.txt`

- 得到如下结果

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
23/09/29 19:40:04 INFO SparkContext: Running Spark version 3.2.4
23/09/29 19:40:04 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
23/09/29 19:40:05 INFO ResourceUtils: =====
23/09/29 19:40:05 INFO ResourceUtils: No custom resources configured for spark.driver.
23/09/29 19:40:05 INFO ResourceUtils: =====
23/09/29 19:40:05 INFO SparkContext: Submitted application: JavaWordCount
23/09/29 19:40:05 INFO ResourceProfile: Default ResourceProfile created, executor resources: Map(cores -> name: cores, amount: 1, script: ,
23/09/29 19:40:05 INFO ResourceProfile: Limiting resource is cpu
23/09/29 19:40:05 INFO ResourceProfileManager: Added ResourceProfile id: 0
23/09/29 19:40:05 INFO SecurityManager: Changing view acls to: xralph
23/09/29 19:40:05 INFO SecurityManager: Changing modify acls to: xralph
23/09/29 19:40:05 INFO SecurityManager: Changing view acls groups to:
23/09/29 19:40:05 INFO SecurityManager: Changing modify acls groups to:
23/09/29 19:40:05 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users with view permissions: Set(xralph)
23/09/29 19:40:05 INFO Utils: Successfully started service 'sparkDriver' on port 58184.
23/09/29 19:40:05 INFO SparkEnv: Registering MapOutputTracker
23/09/29 19:40:05 INFO SparkEnv: Registering BlockManagerMaster
23/09/29 19:40:05 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology information
23/09/29 19:40:05 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
23/09/29 19:40:05 INFO SparkEnv: Registering BlockManagerMasterHeartbeat
23/09/29 19:40:05 INFO DiskBlockManager: Created local directory at /private/var/folders/fq/c01hbj8n5wxg5gkn216hfwvh0000gn/T/blockmgr-198e2
23/09/29 19:40:05 INFO MemoryStore: MemoryStore started with capacity 366.3 MiB
23/09/29 19:40:05 INFO SparkEnv: Registering OutputCommitCoordinator
23/09/29 19:40:05 INFO Utils: Successfully started service 'SparkUI' on port 4040.
23/09/29 19:40:05 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://10.162.86.187:4040
23/09/29 19:40:05 INFO SparkContext: Added JAR file:/Users/xralph/Desktop/未命名文件夹/spark/spark-3.2.4-bin-hadoop3.2/examples/jars/spark-ex-
23/09/29 19:40:05 INFO Executor: Starting executor ID driver on host 10.162.86.187
23/09/29 19:40:05 INFO Executor: Fetching spark://10.162.86.187:58184/jars/spark-examples_2.12-3.2.4.jar with timestamp 1695987604900
23/09/29 19:40:06 INFO TransportClientFactory: Successfully created connection to /10.162.86.187:58184 after 30 ms (0 ms spent in bootstrap)
23/09/29 19:40:06 INFO Utils: Fetching spark://10.162.86.187:58184/jars/spark-examples_2.12-3.2.4.jar to /private/var/folders/fq/c01hbj8n5w
23/09/29 19:40:06 INFO Executor: Adding file:/private/var/folders/fq/c01hbj8n5wxg5gkn216hfwvh0000gn/T/spark-52d711f1-712b-45fb-8b02-7d76152
23/09/29 19:40:06 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 58188.
23/09/29 19:40:06 INFO NettyBlockTransferService: Server created on 10.162.86.187:58188
23/09/29 19:40:06 INFO BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block replication policy
23/09/29 19:40:06 INFO BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 10.162.86.187, 58188, None)
23/09/29 19:40:06 INFO BlockManagerMasterEndpoint: Registering block manager 10.162.86.187:58188 with 366.3 MiB RAM, BlockManagerId(driver,
23/09/29 19:40:06 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 10.162.86.187, 58188, None)
23/09/29 19:40:06 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 10.162.86.187, 58188, None)
23/09/29 19:40:06 INFO SharedState: Setting hive.metastore.warehouse.dir ('null') to the value of spark.sql.warehouse.dir.
23/09/29 19:40:06 INFO SharedState: Warehouse path is 'file:/Users/xralph/Desktop/未命名文件夹/spark/spark-3.2.4-bin-hadoop3.2/bin/spark-warel
23/09/29 19:40:07 INFO InMemoryFileIndex: It took 30 ms to list leaf files for 1 paths.
23/09/29 19:40:09 INFO FileSourceStrategy: Pushed Filters:
23/09/29 19:40:09 INFO FileSourceStrategy: Post-Scan Filters:
23/09/29 19:40:09 INFO FileSourceStrategy: Output Data Schema: struct<value: string>
23/09/29 19:40:09 INFO MemoryStore: Block broadcast_0 stored as values in memory (estimated size 338.3 KiB, free 366.0 MiB)
23/09/29 19:40:11 INFO MemoryStore: Block broadcast_0_piece0 stored as bytes in memory (estimated size 32.6 KiB, free 365.9 MiB)
23/09/29 19:40:11 INFO BlockManagerInfo: Added broadcast_0_piece0 in memory on 10.162.86.187:58188 (size: 32.6 KiB, free: 366.3 MiB)
23/09/29 19:40:11 INFO SparkContext: Created broadcast 0 from javaRDD at JavaWordCount.java:45
23/09/29 19:40:11 INFO FileSourceScanExec: Planning scan with bin packing, max size: 4194304 bytes, open cost is considered as scanning 419
23/09/29 19:40:11 INFO SparkContext: Starting job: collect at JavaWordCount.java:53
23/09/29 19:40:11 INFO DAGScheduler: Registering RDD 6 (mapToPair at JavaWordCount.java:49) as input to shuffle 0
23/09/29 19:40:11 INFO DAGScheduler: Got job 0 (collect at JavaWordCount.java:53) with 1 output partitions
23/09/29 19:40:11 INFO DAGScheduler: Final stage: ResultStage 1 (collect at JavaWordCount.java:53)
23/09/29 19:40:11 INFO DAGScheduler: Parents of final stage: List(ShuffleMapStage 0)
23/09/29 19:40:11 INFO DAGScheduler: Missing parents: List(ShuffleMapStage 0)
23/09/29 19:40:11 INFO DAGScheduler: Submitting ShuffleMapStage 0 (MapPartitionsRDD[6] at mapToPair at JavaWordCount.java:49), which has no
23/09/29 19:40:11 INFO MemoryStore: Block broadcast_1 stored as values in memory (estimated size 15.3 KiB, free 365.9 MiB)
23/09/29 19:40:11 INFO MemoryStore: Block broadcast_1_piece0 stored as bytes in memory (estimated size 7.8 KiB, free 365.9 MiB)
23/09/29 19:40:11 INFO BlockManagerInfo: Added broadcast_1_piece0 in memory on 10.162.86.187:58188 (size: 7.8 KiB, free: 366.3 MiB)
23/09/29 19:40:11 INFO SparkContext: Created broadcast 1 from broadcast at DAGScheduler.scala:1474
23/09/29 19:40:11 INFO DAGScheduler: Submitting 1 missing tasks from ShuffleMapStage 0 (MapPartitionsRDD[6] at mapToPair at JavaWordCount.j
23/09/29 19:40:11 INFO TaskSchedulerImpl: Adding task set 0.0 with 1 tasks resource profile 0
23/09/29 19:40:11 INFO TaskSetManager: Starting task 0.0 in stage 0.0 (TID 0) (10.162.86.187, executor driver, partition 0, PROCESS_LOCAL, .
23/09/29 19:40:11 INFO Executor: Running task 0.0 in stage 0.0 (TID 0)
23/09/29 19:40:12 INFO CodeGenerator: Code generated in 163.803485 ms
23/09/29 19:40:12 INFO FileScanRDD: Reading File path: file:///Users/xralph/Desktop/未命名文件夹/spark/spark-3.2.4-bin-hadoop3.2/data/sample-
```

```

23/09/29 19:40:12 INFO CodeGenerator: Code generated in 11.071031 ms
23/09/29 19:40:12 INFO Executor: Finished task 0.0 in stage 0.0 (TID 0). 1794 bytes result sent to driver
23/09/29 19:40:12 INFO TaskSetManager: Finished task 0.0 in stage 0.0 (TID 0) in 909 ms on 10.162.86.187 (executor driver) (1/1)
23/09/29 19:40:12 INFO TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
23/09/29 19:40:12 INFO DAGScheduler: ShuffleMapStage 0 (mapToPair at JavaWordCount.java:49) finished in 1.077 s
23/09/29 19:40:12 INFO DAGScheduler: looking for newly runnable stages
23/09/29 19:40:12 INFO DAGScheduler: running: Set()
23/09/29 19:40:12 INFO DAGScheduler: waiting: Set(ResultStage 1)
23/09/29 19:40:12 INFO DAGScheduler: failed: Set()
23/09/29 19:40:12 INFO DAGScheduler: Submitting ResultStage 1 (ShuffledRDD[7] at reduceByKey at JavaWordCount.java:51), which has no missing partitions
23/09/29 19:40:12 INFO MemoryStore: Block broadcast_2 stored as values in memory (estimated size 5.2 KiB, free 365.9 MiB)
23/09/29 19:40:12 INFO MemoryStore: Block broadcast_2_piece0 stored as bytes in memory (estimated size 2.9 KiB, free 365.9 MiB)
23/09/29 19:40:12 INFO BlockManagerInfo: Added broadcast_2_piece0 in memory on 10.162.86.187:58188 (size: 2.9 KiB, free: 366.3 MiB)
23/09/29 19:40:12 INFO SparkContext: Created broadcast 2 from broadcast at DAGScheduler.scala:1474
23/09/29 19:40:12 INFO DAGScheduler: Submitting 1 missing tasks from ResultStage 1 (ShuffledRDD[7] at reduceByKey at JavaWordCount.java:51)
23/09/29 19:40:12 INFO TaskSchedulerImpl: Adding task set 1.0 with 1 tasks resource profile 0
23/09/29 19:40:12 INFO TaskSetManager: Starting task 0.0 in stage 1.0 (TID 1) (10.162.86.187, executor driver, partition 0, NODE_LOCAL, 427 bytes)
23/09/29 19:40:12 INFO Executor: Running task 0.0 in stage 1.0 (TID 1)
23/09/29 19:40:12 INFO ShuffleBlockFetcherIterator: Getting 1 (5.7 KiB) non-empty blocks including 1 (5.7 KiB) local and 0 (0.0 B) host-local
23/09/29 19:40:12 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 9 ms
23/09/29 19:40:12 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 14633 bytes result sent to driver
23/09/29 19:40:12 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 136 ms on 10.162.86.187 (executor driver) (1/1)
23/09/29 19:40:12 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
23/09/29 19:40:12 INFO DAGScheduler: ResultStage 1 (collect at JavaWordCount.java:53) finished in 0.146 s
23/09/29 19:40:12 INFO DAGScheduler: Job 0 is finished. Cancelling potential speculative or zombie tasks for this job
23/09/29 19:40:12 INFO TaskSchedulerImpl: Killing all running tasks in stage 1: Stage finished
23/09/29 19:40:12 INFO DAGScheduler: Job 0 finished: collect at JavaWordCount.java:53, took 1.285486 s

interdum: 1024
ultrices.: 328
mi.: 290
erat: 773
Ultrices: 226
Pharetra: 289
Dictumst: 69
laoreet: 859
feugiat: 1613
cum.: 43
congue.: 144
urna: 1995
Velit: 263
Quam: 301
incididunt: 5
Fames: 107
lectus.: 339
Magna: 197
fames: 607
Justo: 163
Lacus: 292
augue: 1226
sapient.: 167
Placerat: 141
rhoncus.: 193
Eros: 81
cursus: 1750
Cum: 38
Augue: 241
nulla.: 427
mollis.: 82
ultrices: 1188
elementum: 1814
Sollicitudin: 144
Platea: 73
elit.: 5
ridiculus: 191
dolore: 5
mauris: 2375
phasellus: 610
est: 1581
facilisis: 1210
Ornare: 248
vulputate: 1367
tempor.: 182
nec.: 288
Quisque: 152
urna.: 354
Molestie: 151
donec: 1421
metus: 578

```

volutpat.: 342  
volutpat: 1923  
ac.: 506  
gravida: 1574  
Metus: 90  
Elit: 308  
bibendum: 1254  
sociis: 200  
Facilisis: 198  
sem.: 181  
Nullam: 148  
sodales.: 87  
Nulla: 418  
enim.: 557  
pellentesque: 2694  
id.: 699  
non.: 433  
viverra.: 460  
ipsum: 1430  
potenti.: 37  
hac.: 75  
Rhoncus: 152  
Vestibulum: 242  
Pellentesque: 500  
pulvinar.: 286  
parturient: 206  
maecenas: 974  
augue.: 212  
Purus: 326  
Sem: 199  
erat.: 118  
montes: 212  
Suspendisse: 224  
Mus: 32  
Tincidunt: 418  
varius: 1038  
duis: 1178  
proin.: 217  
at: 3206  
accumsan.: 141  
Donec: 250  
Praesent: 118  
ut.: 715  
Habitant: 124  
nunc.: 653  
Nibh: 355  
nisl: 1895  
Phasellus: 108  
luctus.: 63  
fringilla: 815  
eros: 410  
Rutrum: 67  
ligula: 193  
dolor: 1268  
hac: 370  
Massa: 442  
Scelerisque: 326  
porttitor.: 177  
mi: 1644  
sagittis.: 260  
accumsan: 777  
phasellus.: 103  
Gravida: 268  
potenti: 211  
massa.: 465  
Nunc: 604  
orci: 1594  
etiam: 942  
dictum: 1002  
dictumst.: 48  
curabitur.: 69  
ipsum.: 233  
Mi: 308  
varius.: 193  
consequat.: 218  
auctor.: 176  
justo.: 136  
tempor: 928

posuere.: 172  
sem: 998  
nunc: 3391  
Duis: 231  
labore: 5  
massa: 2377  
suscipit: 415  
neque.: 398  
Dis: 29  
est.: 301  
Adipiscing: 363  
Vehicula: 30  
commodo.: 225  
platea.: 71  
morbi: 2293  
aenean.: 189  
Tempor: 178  
senectus: 629  
Ligula: 34  
Aliquet: 313  
praesent: 594  
leo: 1323  
fringilla.: 128  
Facilisi: 145  
Venenatis: 188  
Sagittis: 245  
eget.: 678  
vestibulum: 1216  
Magnis: 33  
Ullamcorper: 274  
egestas: 2783  
vitae.: 582  
pharetra: 1632  
fermentum: 1205  
Varius: 198  
pretium.: 284  
lobortis.: 127  
amet.: 809  
eleifend.: 108  
Orci: 312  
diam: 2638  
mattis.: 352  
Habitasse: 66  
hendrerit.: 103  
Non: 469  
Mauris: 437  
suspendisse.: 230  
tincidunt: 2277  
Nisi: 200  
tristique: 1627  
risus: 2238  
porttitor: 962  
dui: 1449  
tincidunt.: 407  
semper: 1193  
do: 5  
Posuere: 192  
sollicitudin: 788  
risus.: 420  
Interdum: 179  
Amet: 816  
placerat: 778  
mus.: 38  
Felis: 153  
tortor.: 363  
molestie.: 134  
nec: 1506  
dui.: 260  
Mattis: 271  
donec.: 251  
dignissim: 1198  
quisque: 795  
Luctus: 73  
arcu: 2499  
enim: 2978  
egestas.: 473  
Hac: 77  
lacus.: 296

facilisi: 803  
aliqua.: 5  
condimentum.: 138  
magna.: 215  
Tristique: 305  
ultricies.: 218  
Tempus: 183  
eiusmod: 5  
integer: 1140  
velit.: 286  
Sodales: 100  
Id: 695  
in: 4748  
Vivamus: 69  
Sociis: 44  
tempus.: 183  
pulvinar: 1400  
Lectus: 339  
iaculis: 756  
Erat: 138  
purus: 2007  
habitant.: 104  
ornare: 1468  
ullamcorper: 1460  
scelerisque.: 335  
diam.: 463  
integer.: 203  
imperdiet: 1011  
Neque: 333  
Pretium: 255  
vulputate.: 270  
Ultrices: 334  
Arcu: 453  
aliquet.: 339  
molestie: 813  
adipiscing.: 313  
netus.: 130  
fermentum.: 215  
cras: 1487  
ornare.: 280  
: 2847  
Aliquam: 436  
laoreet.: 136  
nisl.: 295  
elit: 1538  
lacinia: 418  
blandit.: 218  
faucibus.: 408  
fusce: 385  
proin: 1121  
Egestas: 524  
sit.: 765  
libero: 981  
Enim: 540  
Ipsum: 286  
lobortis: 840  
senectus.: 106  
Penatibus: 40  
nisi.: 222  
Fringilla: 144  
Dictum: 176  
Quis: 471  
Blandit: 213  
Pulvinar: 238  
ac: 2573  
justo: 768  
turpis.: 368  
malesuada: 1419  
magnis: 219  
nibh: 2025  
nulla: 2412  
magnis.: 43  
vehicula: 180  
Fusce: 86  
Ante: 76  
Nisl: 304  
dignissim.: 214  
sed.: 1014

penatibus.: 24  
eros.: 78  
Dolor: 248  
Commodo: 280  
Cras: 245  
Convallis: 200  
porta: 634  
congue: 771  
Ridiculus: 36  
Malesuada: 251  
vivamus.: 68  
nascetur: 208  
Ut: 731  
dapibus.: 26  
habitasse: 383  
aliquet: 1820  
At: 624  
Morbi: 419  
Laoreet: 143  
iaculis.: 180  
Tortor: 379  
morbi.: 365  
condimentum: 826  
Mollis: 68  
Leo: 259  
Tellus: 430  
ut: 3812  
nam.: 104  
suspendisse: 1176  
et.: 643  
sed: 5650  
magna: 1225  
semper.: 239  
amet: 4594  
Suscipit: 55  
Sapient: 146  
natoque.: 34  
Congue: 148  
tellus: 2407  
Sed: 986  
neque: 2070  
faucibus: 2180  
viverra: 2583  
consectetur: 1312  
odio.: 329  
Eleifend: 119  
consectetur.: 256  
et: 3484  
Proin: 187  
sociis.: 38  
Risus: 410  
scelerisque: 1762  
felis.: 158  
lacus: 1483  
nullam.: 141  
quam.: 320  
suscipit.: 77  
Accumsan: 176  
mauris.: 411  
Vulputate: 238  
Cursus: 314  
facilisi.: 135  
consequat: 1203  
Natoque: 44  
pellentesque.: 512  
Curabitur: 77  
nullam: 804  
aenean: 1041  
felis: 994  
lorem: 1147  
cum: 200  
orci.: 290  
duis.: 222  
rhoncus: 999  
Et: 575  
hendrerit: 470  
porta.: 98  
Nascetur: 29

vestibulum.: 231  
nascetur.: 40  
Euismod: 174  
Lobortis: 146  
dis.: 39  
quis: 2588  
Urna: 357  
sollicitudin.: 152  
Semper: 195  
purus.: 363  
cras.: 253  
netus: 630  
Consequat: 227  
Lacinia: 70  
Maecenas: 173  
Iaculis: 142  
ullamcorper.: 268  
Vitae: 622  
dictum.: 193  
montes.: 30  
imperdiet.: 181  
turpis: 1952  
quisque.: 158  
Viverra: 512  
quis.: 473  
Libero: 170  
curabitur: 418  
penatibus: 230  
tellus.: 445  
etiam.: 221  
Porta: 114  
Lorem: 281  
gravida.: 310  
auctor: 968  
ligula.: 36  
a.: 387  
dolor.: 223  
in.: 868  
A: 372  
Porttitor: 194  
nam: 562  
Eu: 531  
dapibus: 210  
amet,: 5  
ante.: 68  
tempus: 988  
facilisis.: 218  
Vel: 393  
vivamus: 398  
dis: 213  
Potenti: 38  
arcu.: 440  
Dui: 262  
nisi: 1183  
placerat.: 162  
Netus: 126  
tristique.: 291  
nibh.: 377  
Sit: 867  
pretium: 1347  
malesuada.: 256  
ante: 395  
venenatis.: 173  
eu: 2766  
praesent.: 95  
Integer: 215  
non: 2472  
Dignissim: 238  
Dapibus: 36  
sagittis: 1460  
Condimentum: 149  
tortor: 2035  
blandit: 1161  
quam: 1804  
velit: 1678  
pharetra.: 299  
Feugiat: 251  
Aenean: 177



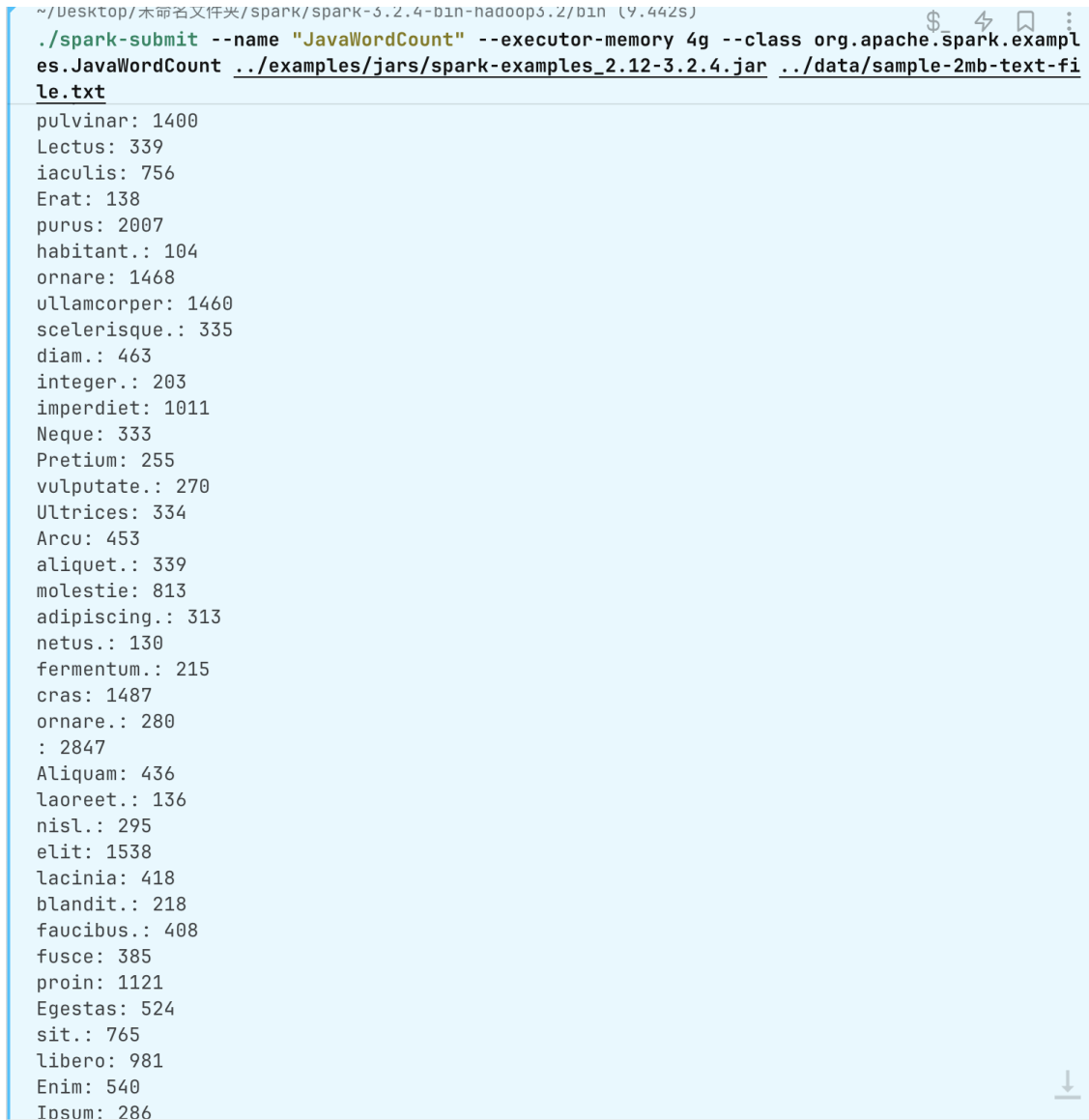
Imperdiet: 180  
mattis: 1649  
vehicula.: 39  
sodales: 655  
convallis.: 203  
Senectus: 106  
lorem.: 221  
sit: 4541  
elit.: 282  
elementum.: 344  
ridiculus.: 46  
Est: 271  
cursus.: 331  
maecenas.: 189  
Hendrerit: 79  
venenatis: 1024  
Consectetur: 235  
posuere: 1004  
odio: 1815  
sapien: 1039  
ultrices: 1918  
bibendum.: 234  
aliquam.: 387  
euismod.: 203  
parturient.: 36  
eget: 3700  
Odio: 331  
vel.: 367  
libero.: 185  
euismod: 1030  
platea: 378  
habitant: 607  
In: 896  
natoque: 210  
Faucibus: 392  
Montes: 36  
Etiam: 169  
fames.: 112  
metus.: 108  
convallis: 1065  
leo.: 244  
dictumst: 403  
Fermentum: 225  
Ac: 431  
at.: 583  
eu.: 546  
Eget: 690  
vitae: 3361  
rutrum.: 65  
vel: 1979  
feugiat.: 306  
Diam: 514  
habitasse.: 64  
a: 2029  
Auctor: 191  
lacinia.: 63  
Turpis: 373  
Volutpat: 354  
commodo: 1336  
interdum.: 200  
luctus: 376  
aliquam: 2358  
id: 3853  
Parturient: 36  
lectus: 1890  
Nam: 124  
adipiscing: 1956  
rutrum: 411  
eleifend: 598  
mus: 189  
Elementum: 328  
fusce.: 73  
mollis: 391  
Nec: 289  
Bibendum: 223  
23/09/29 19:40:12 INFO SparkUI: Stopped Spark web UI at http://10.162.86.187:4040  
23/09/29 19:40:12 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!  
23/09/29 19:40:12 INFO MemoryStore: MemoryStore cleared

```

23/09/29 19:40:12 INFO BlockManager: BlockManager stopped
23/09/29 19:40:12 INFO BlockManagerMaster: BlockManagerMaster stopped
23/09/29 19:40:12 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
23/09/29 19:40:12 INFO SparkContext: Successfully stopped SparkContext
23/09/29 19:40:12 INFO ShutdownHookManager: Shutdown hook called
23/09/29 19:40:12 INFO ShutdownHookManager: Deleting directory /private/var/folders/fq/c01hbj8n5wxg5gkn216hfwvh0000gn/T/spark-52d711f1-712b
23/09/29 19:40:12 INFO ShutdownHookManager: Deleting directory /private/var/folders/fq/c01hbj8n5wxg5gkn216hfwvh0000gn/T/spark-e33a6f32-6ce2

```

- 运行截图



```

~/Desktop/未命名文件夹/spark/spark-3.2.4-bin-hadoop3.2/bin (9.442s)
./spark-submit --name "JavaWordCount" --executor-memory 4g --class org.apache.spark.examples.JavaWordCount ./examples/jars/spark-examples_2.12-3.2.4.jar ../data/sample-2mb-text-file.txt
pulvinar: 1400
Lectus: 339
iaculis: 756
Erat: 138
purus: 2007
habitant.: 104
ornare: 1468
ullamcorper: 1460
scelerisque.: 335
diam.: 463
integer.: 203
imperdiet: 1011
Neque: 333
Pretium: 255
vulputate.: 270
Ultrices: 334
Arcu: 453
aliquet.: 339
molestie: 813
adipiscing.: 313
netus.: 130
fermentum.: 215
cras: 1487
ornare.: 280
: 2847
Aliquam: 436
laoreet.: 136
nisl.: 295
elit: 1538
lacinia: 418
blandit.: 218
faucibus.: 408
fusce: 385
proin: 1121
Egestas: 524
sit.: 765
libero: 981
Enim: 540
Insum: 286

```

### 三、编写自定义并行处理任务

#### 3.1 辅助类 `FrechetDistanceHelper`

- 这个类的主要作用就是计算 `discretFrechet` 距离，代码如下，其实现参考了<https://zhuanlan.zhihu.com/p/74561481>

```

package org.example;

import java.util.Arrays;
import java.util.List;

```

```

import org.apache.hadoop.shaded.org.eclipse.jetty.util.ajax.JSON;
import scala.Tuple2;

public class FrechetDistanceHelper {

    public static double euclideanDistance(double[] x, double[] y) {
        double sum = 0.0;
        for (int i = 0; i < x.length; i++) {
            double diff = x[i] - y[i];
            sum += diff * diff;
        }
        return Math.sqrt(sum);
    }

    private static double _c(double[][] ca, int i, int j, double[][] P, double[][] Q) {
        if (ca[i][j] > -1) {
            return ca[i][j];
        } else if (i == 0 && j == 0) {
            ca[i][j] = euclideanDistance(P[0], Q[0]);
        } else if (i > 0 && j == 0) {
            ca[i][j] = Math.max(_c(ca, i - 1, 0, P, Q), euclideanDistance(P[i], Q[0]));
        } else if (i == 0 && j > 0) {
            ca[i][j] = Math.max(_c(ca, 0, j - 1, P, Q), euclideanDistance(P[0], Q[j]));
        } else if (i > 0 && j > 0) {
            ca[i][j] =
                Math.max(Math.min(Math.min(_c(ca, i - 1, j, P, Q), _c(ca, i - 1, j - 1, P, Q)), _c(ca, i, j - 1, P, Q)),
                    euclideanDistance(P[i], Q[j]));
        } else {
            ca[i][j] = Double.POSITIVE_INFINITY;
        }
        return ca[i][j];
    }

    public static double discretFrechet(List<Tuple2<Double, Double>> P, List<Tuple2<Double, Double>> Q) {
        double[][] ca = new double[P.size()][Q.size()];
        for (double[] row : ca) {
            Arrays.fill(row, -1.0);
        }
        return _c(ca, P.size() - 1, Q.size() - 1,
            P.stream().map(tuple -> new double[] {tuple._1, tuple._2}).toArray(double[][]::new),
            Q.stream().map(tuple -> new double[] {tuple._1, tuple._2}).toArray(double[][]::new));
    }
}

```

### 3.2 主类 **FrechetDistanceJob**

- 实际执行运算工作的类，代码如下
- 在使用前，务必确保 `basePath`是输入的Base, `queryPath`是待查数据文件夹, `resultPath`是输出文件夹, 且非空

```

package org.example;

import java.io.Serializable;
import java.nio.file.Paths;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;
import java.util.stream.Collectors;
import org.apache.spark.SparkConf;
import org.apache.spark.api.java.JavaPairRDD;
import org.apache.spark.api.java.JavaSparkContext;
import scala.Tuple2;

public class FrechetDistanceJob implements Serializable {
    private static final String basePath = "../课程作业/BaseTraj";
    private static final String queryPath = "../课程作业/QueryTraj";
    private static final String resultPath = "../课程作业/Result";

    public static void main(String[] args) {
        SparkConf conf = new SparkConf().setAppName("FrechetDistanceJob").setMaster("local[*]");
        JavaSparkContext sc = new JavaSparkContext(conf);

        List<Tuple2<String, List<Tuple2<Double, Double>>>> samples = sc.wholeTextFiles(basePath).mapToPair(tuple -> {
            String fileName = Paths.get(tuple._1).getFileName().toString();
            String fileContent = tuple._2;

```

```

        List

```

- 运行上述程序，即可在 `resultPath` 中得到结果（请在结果后加入.txt后缀）

queryPath	Result	今天 00:09	--	文件夹
	_SUCCESS	今天 00:08	0 字节	文稿
	part-00000	今天 00:08	977 字节	文稿
	part-00001	今天 00:08	972 字节	文稿

- 以下是对结果格式的介绍
  - 每行的第一个文件名，表示查询的文件名
  - 每行的最后一个文件名，表示对应 `discretFrechet` 最小的 `Base` 文件名
  - 每行中间的值，表示两个文件名对应的轨迹的 `discretFrechet` 距离

part-00000.txt

使用“文本编辑器”

```
(6.txt,(0.005259922778802331,138.txt))
(24.txt,(0.009709580318146551,152.txt))
(40.txt,(0.015426318101722063,143.txt))
(4.txt,(0.018007449145665034,168.txt))
(35.txt,(0.0076258086468902146,49.txt))
(11.txt,(0.0058643096638424175,25.txt))
(37.txt,(0.03269117445489319,130.txt))
(39.txt,(0.008727465310157353,76.txt))
(19.txt,(0.00844945057540221,40.txt))
(28.txt,(0.04006611880491308,130.txt))
(2.txt,(0.006860028139881258,1.txt))
(17.txt,(0.044960649039466095,46.txt))
(20.txt,(0.009509082187575596,3.txt))
(46.txt,(0.011377377617886857,147.txt))
(26.txt,(0.004835320300663208,100.txt))
(31.txt,(0.007925124214167321,84.txt))
(22.txt,(0.009752974396049661,89.txt))
(44.txt,(0.007614749372763023,138.txt))
(15.txt,(0.007403397667075119,62.txt))
(13.txt,(0.016803000420247252,16.txt))
(8.txt,(0.010569302480872749,92.txt))
(42.txt,(0.015562293225506792,133.txt))
(33.txt,(0.03404880831776413,44.txt))
(0.txt,(0.008207835365062569,28.txt))
(48.txt,(0.012824661545057453,161.txt))
```

### 3.3 如何运行？

- 在idea中，导入附件中的项目文件
- 将basePath, queryPath和resultPath修改为对应的值，如果不修改，默认使用老师下发的示例文件路径
- 执行项目，可以在resultPath对应的目录下看到结果（记得添加.txt后缀，由于是并发执行，结果是无序的）
- 附件中，有我运行的结果，以及运行过程的视频