```
                                  ^

scala> val cards = sc.textFile("/user/mydir/deckofcards.txt")
cards: org.apache.spark.rdd.RDD[String] = /user/mydir/deckofcards.txt MapPartitionsRDD[1] at textFile at
 <console>:27

scala> cards.collect()
res0: Array[String] = Array(BLACK|SPADE|2, BLACK|SPADE|3, BLACK|SPADE|4, BLACK|SPADE|5, BLACK|SPADE|6, B
LACK|SPADE|7, BLACK|SPADE|8, BLACK|SPADE|9, BLACK|SPADE|10, BLACK|SPADE|J, BLACK|SPADE|Q, BLACK|SPADE|K,
 BLACK|SPADE|A, BLACK|CLUB|2, BLACK|CLUB|3, BLACK|CLUB|4, BLACK|CLUB|5, BLACK|CLUB|6, BLACK|CLUB|7, BLAC
K|CLUB|8, BLACK|CLUB|9, BLACK|CLUB|10, BLACK|CLUB|J, BLACK|CLUB|Q, BLACK|CLUB|K, BLACK|CLUB|A, RED|DIAMO
ND|2, RED|DIAMOND|3, RED|DIAMOND|4, RED|DIAMOND|5, RED|DIAMOND|6, RED|DIAMOND|7, RED|DIAMOND|8, RED|DIAM
OND|9, RED|DIAMOND|10, RED|DIAMOND|J, RED|DIAMOND|Q, RED|DIAMOND|K, RED|DIAMOND|A, RED|HEART|2, RED|HEAR
T|3, RED|HEART|4, RED|HEART|5, RED|HEART|6, RED|HEART|7, RED|HEART|8, RED|HEART|9, RED|HEART|10, RED|HEA
RT|J, RED|HEART|Q, RED|HEART|K, RED|HEART|A)

scala> █
```

```
scala> cards.collect()
res0: Array[String] = Array(BLACK|SPADE|2, BLACK|SPADE|3, BLACK|SPADE|4, BLACK|SPADE|5, BLACK|SPADE|6, B
LACK|SPADE|7, BLACK|SPADE|8, BLACK|SPADE|9, BLACK|SPADE|10, BLACK|SPADE|J, BLACK|SPADE|Q, BLACK|SPADE|K,
 BLACK|SPADE|A, BLACK|CLUB|2, BLACK|CLUB|3, BLACK|CLUB|4, BLACK|CLUB|5, BLACK|CLUB|6, BLACK|CLUB|7, BLAC
K|CLUB|8, BLACK|CLUB|9, BLACK|CLUB|10, BLACK|CLUB|J, BLACK|CLUB|Q, BLACK|CLUB|K, BLACK|CLUB|A, RED|DIAMO
ND|2, RED|DIAMOND|3, RED|DIAMOND|4, RED|DIAMOND|5, RED|DIAMOND|6, RED|DIAMOND|7, RED|DIAMOND|8, RED|DIAM
OND|9, RED|DIAMOND|10, RED|DIAMOND|J, RED|DIAMOND|Q, RED|DIAMOND|K, RED|DIAMOND|A, RED|HEART|2, RED|HEAR
T|3, RED|HEART|4, RED|HEART|5, RED|HEART|6, RED|HEART|7, RED|HEART|8, RED|HEART|9, RED|HEART|10, RED|HEA
RT|J, RED|HEART|Q, RED|HEART|K, RED|HEART|A)

scala> cards.count()
res1: Long = 52

scala> val cards = sc.textFile("/user/mydir/largedeck.txt")
cards: org.apache.spark.rdd.RDD[String] = /user/mydir/largedeck.txt MapPartitionsRDD[3] at textFile at <cons
ole>:27

scala> cards.count()
res2: Long = 54525952

scala> val blackcards = cards.filter( line => line.contains("BLACK"))
blackcards: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[4] at filter
at <console>:29
```

```
scala> val cards = sc.textFile("/user/mydir/largedeck.txt")
cards: org.apache.spark.rdd.RDD[String] = /user/mydir/largedeck.txt MapPartitionsRDD[1] at textFile at <cons
ole>:27

scala> val blackcards = cards.filter( line => line.contains("BLACK"))
blackcards: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[2] at filter at <console>:29

scala> blackcards.count()
res0: Long = 27262976

scala> █
```