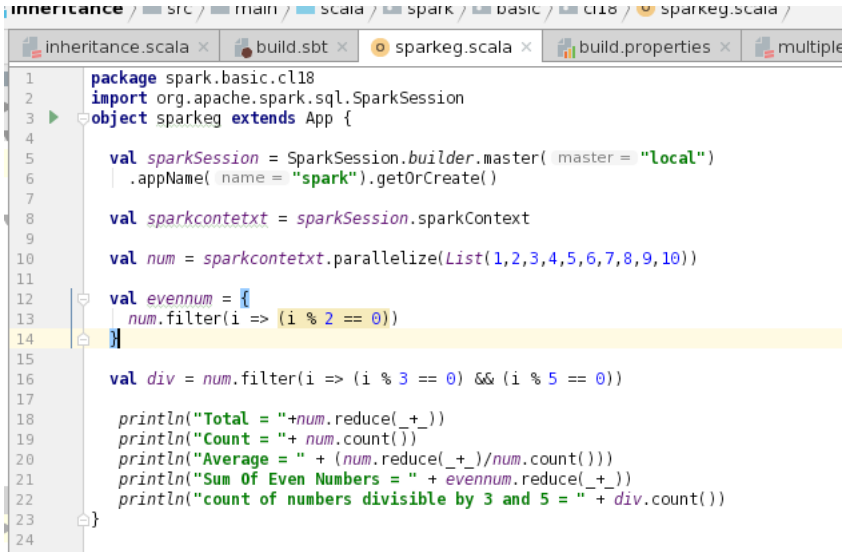
Spark

Task 1

Given a list of numbers - List[Int] (1, 2, 3, 4, 5, 6, 7, 8, 9, 10)

Code:

```
package spark.basic.cl18
import org.apache.spark.sql.SparkSession
object sparkeg extends App {

  val sparkSession = SparkSession.builder.master("local")
    .appName("spark").getOrCreate()

  val sparkcontetxt = sparkSession.sparkContext

  val num = sparkcontetxt.parallelize(List(1,2,3,4,5,6,7,8,9,10))

  val evennum = {
    num.filter(i => (i % 2 == 0))
  }

  val div = num.filter(i => (i % 3 == 0) && (i % 5 == 0))

   println("Total = "+num.reduce(_+_))
   println("Count = "+ num.count())
   println("Average = " + (num.reduce(_+_)/num.count()))
   println("Sum Of Even Numbers = " + evennum.reduce(_+_))
   println("count of numbers divisible by 3 and 5 = " + div.count())
}
```



- find the sum of all numbers

```
18/07/18 16:43:41 INFO Executor: Running task 0.0 in stage 0.0 (T
18/07/18 16:43:41 INFO Executor: Finished task 0.0 in stage 0.0 (
18/07/18 16:43:41 INFO TaskSetManager: Finished task 0.0 in stage
18/07/18 16:43:41 INFO TaskSchedulerImpl: Removed TaskSet 0.0, wh
18/07/18 16:43:41 INFO DAGScheduler: ResultStage 0 (reduce at spa
18/07/18 16:43:41 INFO DAGScheduler: Job 0 finished: reduce at sp
Total = 55
18/07/18 16:43:41 INFO SparkContext: Starting job: count at spark
18/07/18 16:43:41 INFO DAGScheduler: Got job 1 (count at sparkeg.
18/07/18 16:43:41 INFO DAGScheduler: Final stage: ResultStage 1 (
18/07/18 16:43:41 INFO DAGScheduler: Parents of final stage: List
18/07/18 16:43:41 INFO DAGScheduler: Missing parents: List()
18/07/18 16:43:41 INFO DAGScheduler: Submitting ResultStage 1 (Pa
```

- find the total elements in the list

```
18/07/18 16:43:41 INFO TaskSetManager: Starting task 0.0 in stage 1.0 (TID 1
18/07/18 16:43:41 INFO Executor: Running task 0.0 in stage 1.0 (TID 1)
18/07/18 16:43:41 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 789
18/07/18 16:43:41 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1
18/07/18 16:43:41 INFO DAGScheduler: ResultStage 1 (count at sparkeg.scala:1
18/07/18 16:43:41 INFO DAGScheduler: Job 1 finished: count at sparkeg.scala:
Count = 10
18/07/18 16:43:41 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks h
18/07/18 16:43:41 INFO SparkContext: Starting job: reduce at sparkeg.scala:2
18/07/18 16:43:41 INFO DAGScheduler: Got job 2 (reduce at sparkeg.scala:20)
18/07/18 16:43:41 INFO DAGScheduler: Final stage: ResultStage 2 (reduce at s
18/07/18 16:43:41 INFO DAGScheduler: Parents of final stage: List()
18/07/18 16:43:41 INFO DAGScheduler: Missing parents: List()
18/07/18 16:43:41 INFO DAGScheduler: Submitting ResultStage 2 (ParallelColle
18/07/18 16:43:41 INFO MemoryStore: Block broadcast_2 stored as values in me
18/07/18 16:43:41 INFO MemoryStore: Block broadcast_2_piece0 stored as bytes
18/07/18 16:43:41 INFO BlockManagerInfo: Added broadcast_2_piece0 in memory
18/07/18 16:43:41 INFO SparkContext: Created broadcast 2 from broadcast at D
```

- calculate the average of the numbers in the list

```
Run: sparkeg
18/07/18 16:43:41 INFO BlockManagerInfo: Added broadcast_3_piece
18/07/18 16:43:41 INFO SparkContext: Created broadcast 3 from br
18/07/18 16:43:41 INFO DAGScheduler: Submitting 1 missing tasks
18/07/18 16:43:41 INFO TaskSchedulerImpl: Adding task set 3.0 wi
18/07/18 16:43:41 INFO TaskSetManager: Starting task 0.0 in stag
18/07/18 16:43:41 INFO Executor: Running task 0.0 in stage 3.0 (
18/07/18 16:43:41 INFO BlockManagerInfo: Removed broadcast_1_pie
18/07/18 16:43:41 INFO Executor: Finished task 0.0 in stage 3.0
18/07/18 16:43:41 INFO TaskSetManager: Finished task 0.0 in stag
Average = 5
18/07/18 16:43:41 INFO DAGScheduler: ResultStage 3 (count at spa
18/07/18 16:43:41 INFO DAGScheduler: Job 3 finished: count at sp
18/07/18 16:43:41 INFO TaskSchedulerImpl: Removed TaskSet 3.0, w
18/07/18 16:43:41 INFO BlockManagerInfo: Removed broadcast_2_pie
18/07/18 16:43:41 INFO SparkContext: Starting job: reduce at spa
18/07/18 16:43:41 INFO DAGScheduler: Got job 4 (reduce at sparke
18/07/18 16:43:41 INFO DAGScheduler: Final stage: ResultStage 4
18/07/18 16:43:41 INFO DAGScheduler: Parents of final stage: Lis
18/07/18 16:43:41 INFO DAGScheduler: Missing parents: List()
18/07/18 16:43:41 INFO DAGScheduler: Submitting ResultStage 4 (M
18/07/18 16:43:41 INFO BlockManagerInfo: Removed broadcast_3_pie
18/07/18 16:43:41 INFO MemoryStore: Block broadcast_4 stored as
18/07/18 16:43:41 INFO MemoryStore: Block broadcast_4_piece0 sto
```

- find the sum of all the even numbers in the list

```
18/07/18 16:43:41 INFO TaskSchedulerImpl: Removed TaskSe
18/07/18 16:43:41 INFO DAGScheduler: ResultStage 4 (redu
18/07/18 16:43:41 INFO DAGScheduler: Job 4 finished: rec
Sum Of Even Numbers = 30
18/07/18 16:43:41 INFO SparkContext: Starting job: count
18/07/18 16:43:41 INFO DAGScheduler: Got job 5 (count at
18/07/18 16:43:41 INFO DAGScheduler: Final stage: Result
18/07/18 16:43:41 INFO DAGScheduler: Parents of final st
18/07/18 16:43:41 INFO DAGScheduler: Missing parents: Li
18/07/18 16:43:41 INFO DAGScheduler: Submitting ResultSt
18/07/18 16:43:41 INFO MemoryStore: Block broadcast_5 st
```

- find the total number of elements in the list divisible by both 5 and 3

```
18/07/18 16:43:41 INFO TaskSetManager: Starting task 0.0 in stage 5.
18/07/18 16:43:41 INFO Executor: Running task 0.0 in stage 5.0 (TID
18/07/18 16:43:41 INFO Executor: Finished task 0.0 in stage 5.0 (TID
18/07/18 16:43:41 INFO TaskSetManager: Finished task 0.0 in stage 5.
18/07/18 16:43:41 INFO TaskSchedulerImpl: Removed TaskSet 5.0, whose
18/07/18 16:43:41 INFO DAGScheduler: ResultStage 5 count at sparkeg
count of numbers divisible by 3 and 5 = 0
18/07/18 16:43:41 INFO DAGScheduler: Job 5 finished: count at sparke
18/07/18 16:43:41 INFO SparkContext: Invoking stop() from shutdown h
18/07/18 16:43:41 INFO SparkUI: Stopped Spark web UI at http://192.1
18/07/18 16:43:41 INFO MapOutputTrackerMasterEndpoint: MapOutputTrac
18/07/18 16:43:41 INFO MemoryStore: MemoryStore cleared
Compilation completed successfully in 7 s 372 ms (2 minutes ago)
```

Task 2

1) Pen down the limitations of MapReduce.

1. Interactive Processing
2. Real-time (stream) Processing
3. Iterative (delta) Processing
4. In-memory Processing
5. Graph Processing

2) What is RDD? Explain few features of RDD?

Resilient Distributed Datasets is a fundamental data structure of Spark. It is an immutable distributed collection of objects. Each dataset in RDD is divided into logical partitions, which may be computed on different nodes of the cluster.

Features:

- **Resilient** = fault-tolerant with the help of RDD lineage graph and so able to recompute missing or damaged partitions due to node failures.
- **Distributed** with data residing on multiple nodes in a cluster.
  - **In-Memory** = data inside RDD is stored in memory as much (size) and long (time) as possible.
  - **Immutable** or **Read-Only** = it does not change once created and can only be transformed using transformations to new RDDs.
  - **Lazy evaluated** = the data inside RDD is not available or transformed until an action is executed that triggers the execution.
  - **Cacheable** = hold all the data in a persistent "storage" like memory (default and the most preferred) or disk (the least preferred due to access speed).
  - **Parallel** = process data in parallel.

3) List down few Spark RDD operations and explain each of them.

Count:
count is used to return the number of elements in the RDD. Below is the sample demonstration of the above scenario.

CountByValue:
countByValue is used to count the number of occurrences of the elements in the RDD. Below is the sample demonstration of the above scenario.

Collect:
collect is used to return all the elements in the RDD.

Filter:
filter returns an RDD which meets the filter condition.

Map:
Map will take each row as input and return an RDD for the row.