

Group A

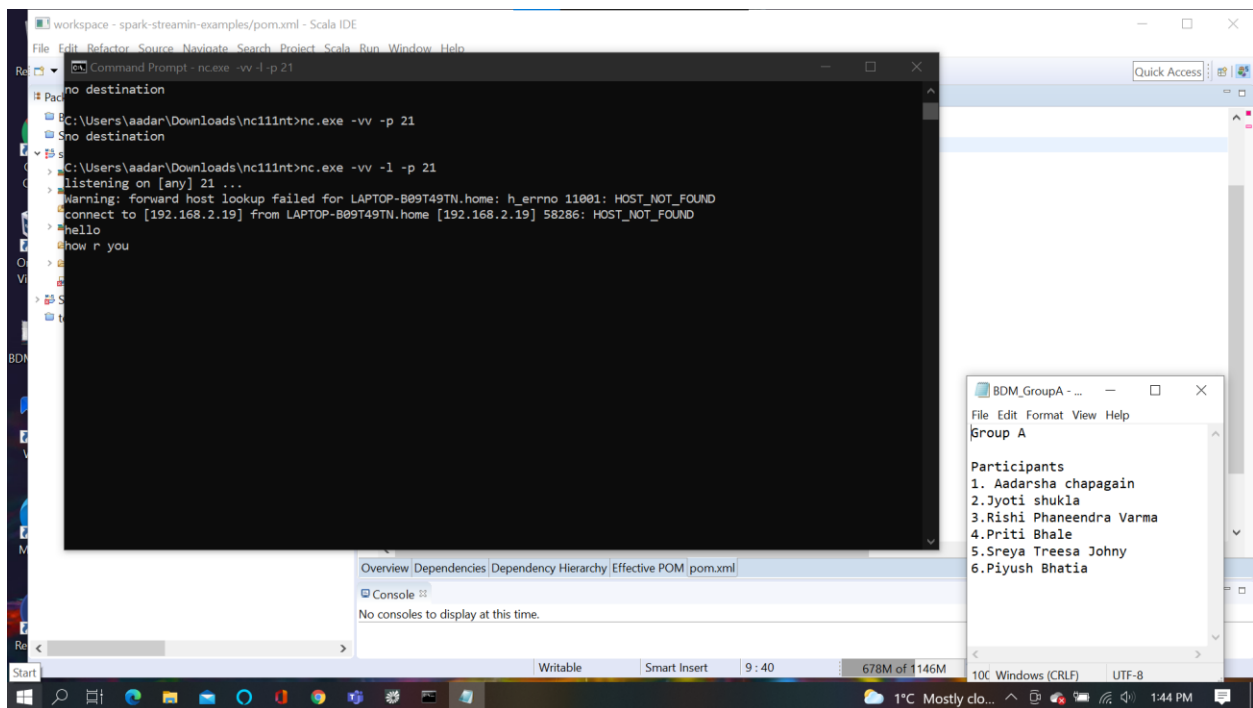
Lab 4a

Spark Streaming

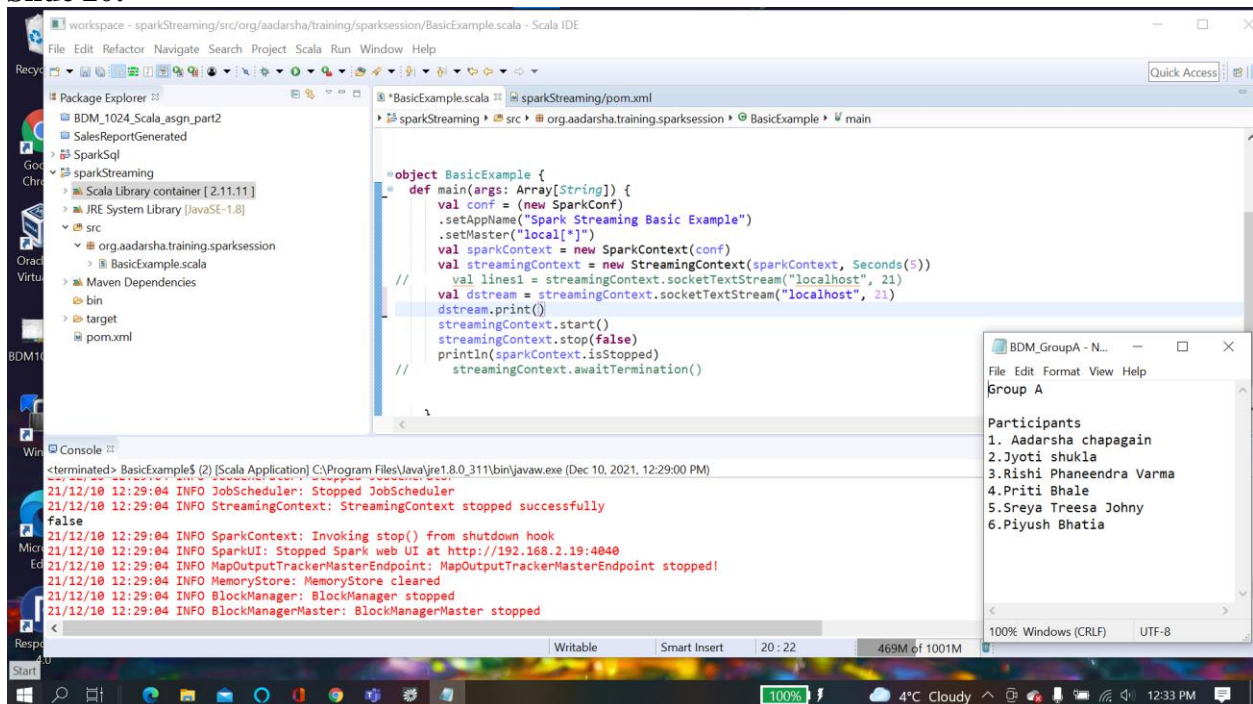
Participants:

1. Aadarsha chapagain
2. Jyoti shukla
3. Rishi Phaneendra Varma
4. Priti Bhale
5. Sreya Treasa Johny
6. Piyush Bhatia

Slide 9:



Slide 20:



Slide 42:

package org.aadarsha.training.sparksession

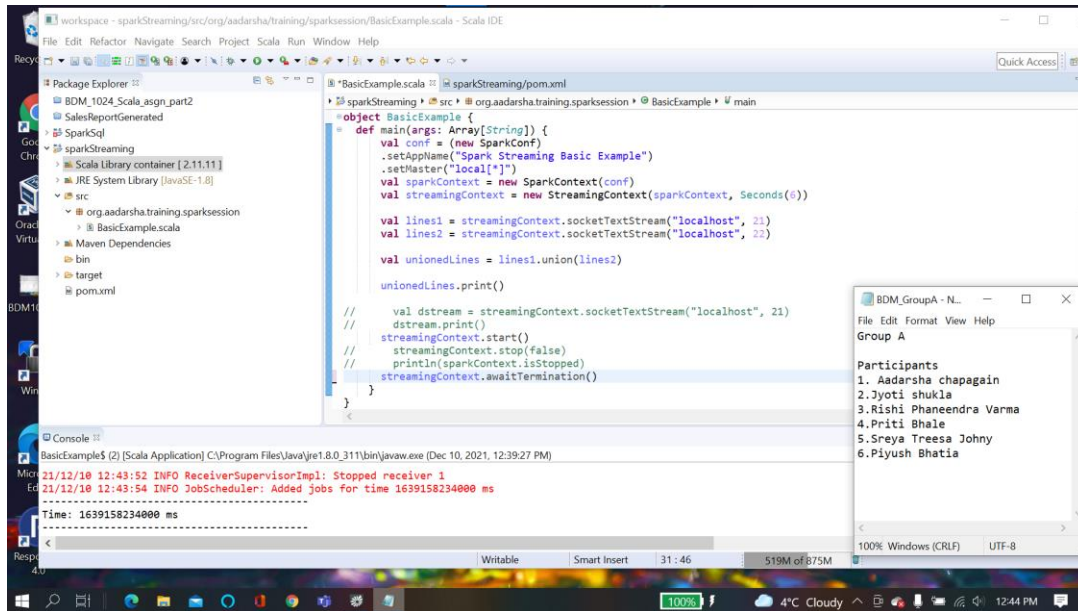
```
import org.apache.spark.SparkConf
import org.apache.spark.streaming.Seconds
import org.apache.spark.streaming.StreamingContext
import org.apache.spark.SparkContext
//import org.apache.spark.sql.SparkSession
```

```
object BasicExample {
  def main(args: Array[String]) {
    val conf = (new SparkConf)
      .setAppName("Spark Streaming Basic Example")
      .setMaster("local[*]")
    val sparkContext = new SparkContext(conf)
    val streamingContext = new StreamingContext(sparkContext, Seconds(6))

    val lines1 = streamingContext.socketTextStream("localhost", 21)
    val lines2 = streamingContext.socketTextStream("localhost", 22)

    val unionedLines = lines1.union(lines2)
    unionedLines.print()
    streamingContext.start()
    streamingContext.awaitTermination()
  }
}
```

}



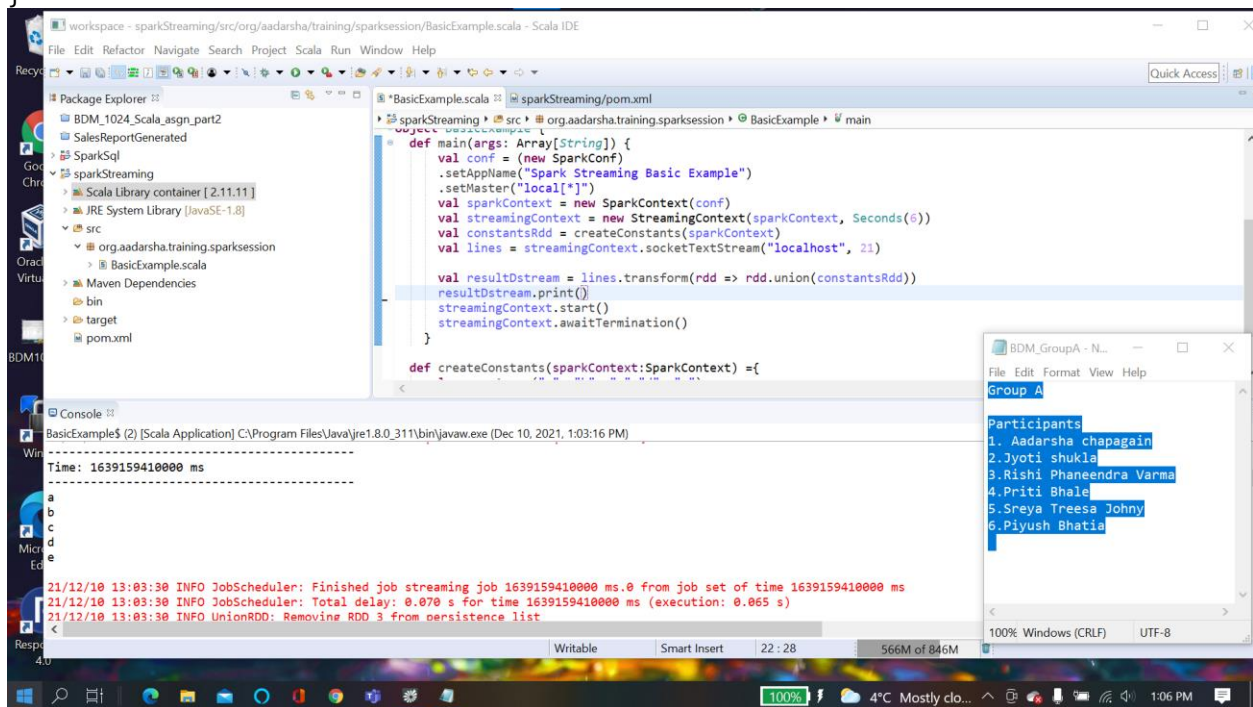
Slide 52:

package org.aadarsha.training.sparksession

```
import org.apache.spark.SparkConf  
import org.apache.spark.streaming.Seconds  
import org.apache.spark.streaming.StreamingContext  
import org.apache.spark.SparkContext
```

```
object BasicExample {  
  def main(args: Array[String]) {  
    val conf = (new SparkConf)  
      .setAppName("Spark Streaming Basic Example")  
      .setMaster("local[*]")  
    val sparkContext = new SparkContext(conf)  
    val streamingContext = new StreamingContext(sparkContext, Seconds(6))  
    val constantsRdd = createConstants(sparkContext)  
    val lines = streamingContext.socketTextStream("localhost", 21)  
  
    val resultDstream = lines.transform(rdd => rdd.union(constantsRdd))  
    resultDstream.print()  
  
    streamingContext.start()  
    streamingContext.awaitTermination()  
  }  
}
```

```
def createConstants(sparkContext:SparkContext) = {  
  val arr = Array("a", "b", "c", "d", "e")  
  sparkContext.parallelize(arr)
```



Slide 57:

package org.aadarsha.training.sparksession

import org.apache.spark.SparkConf

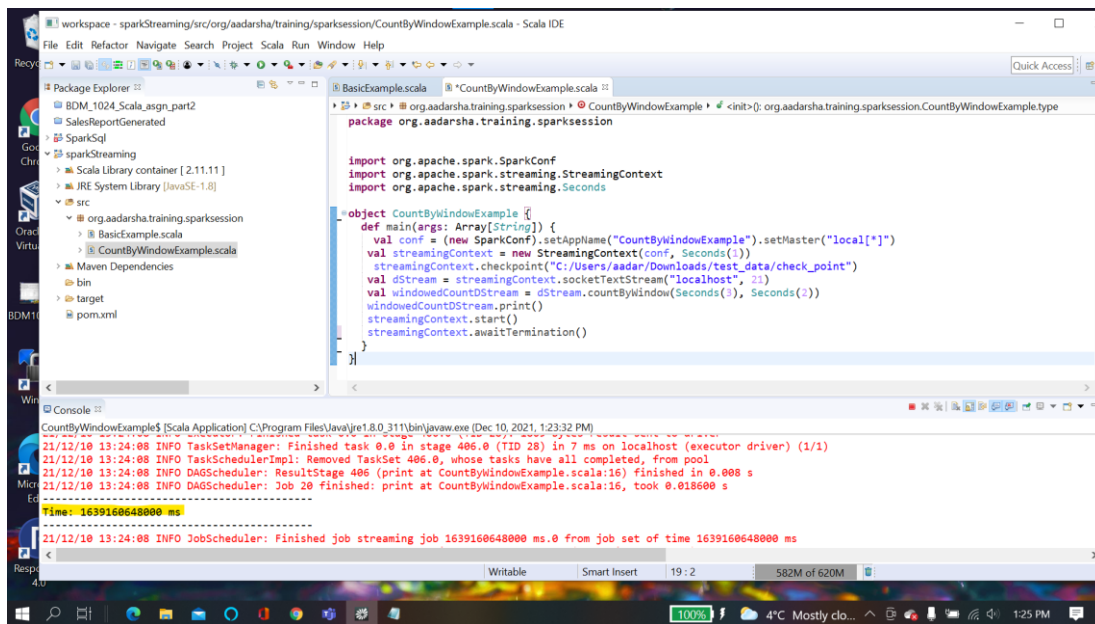
import org.apache.spark.streaming.StreamingContext

import org.apache.spark.streaming.Seconds

```

object CountByWindowExample {
  def main(args: Array[String]) {
    val conf = (new
SparkConf).setAppName("CountByWindowExample").setMaster("local[*]")
    val streamingContext = new StreamingContext(conf, Seconds(1))
    streamingContext.checkpoint("C:/Users/aadar/Downloads/test_data/check_point")
    val dStream = streamingContext.socketTextStream("localhost", 21)
    val windowedCountDStream = dStream.countByWindow(Seconds(3), Seconds(2))
    windowedCountDStream.print()
    streamingContext.start()
    streamingContext.awaitTermination()
  }
}

```



Slide 75:

```
package org.aadarsha.training.sparksession
import org.apache.spark.SparkConf
import org.apache.spark.streaming.StreamingContext
import org.apache.spark.streaming.Seconds
import org.apache.spark.streaming.dstream.DStream.toPairDStreamFunctions
import java.time.Instant

object ReduceByKeyAndWindowExample {
  def main(args: Array[String]) {
    val conf = (new
SparkConf).setAppName("CountByWindowExample").setMaster("local[*]")
    val streamingContext = new StreamingContext(conf, Seconds(2))
    streamingContext.checkpoint("C:/Users/aadar/Downloads/test_data/check_point")
    val dStream = streamingContext.socketTextStream("localhost", 21)
    val windowedCountDStream = dStream.window(Seconds(10), Seconds(8))
    windowedCountDStream.foreachRDD(rdd=>{
      println("Hello")
      rdd.saveAsTextFile("C:/Users/aadar/Downloads/test_data/output")
    })
    streamingContext.start()
    streamingContext.awaitTermination()
  }
}
```

workspace - sparkStreaming/src/org/aadarsha/training/sparksession/ReduceByKeyAndWindowExample.scala - Scala IDE

File Edit Refactor Navigate Search Project Scala Run Window Help

Package Explorer

- sparkStreaming
- src
- org.aadarsha.training.sparksession
- BasicExample.scala
- CountByWindowExample.scala
- ReduceByKeyAndWindowExample.scala
- bin
- target
- pom.xml

```
import org.apache.spark.streaming.StreamingContext
import org.apache.spark.streaming.Seconds
import org.apache.spark.streaming.dstream.DStream.toPairDStreamFunctions
import java.time.Instant

object ReduceByKeyAndWindowExample {
  def main(args: Array[String]) {
    val conf = (new SparkConf).setAppName("CountByWindowExample").setMaster("local[*]")
    val streamingContext = new StreamingContext(conf, Seconds(2))
    streamingContext.checkpoint("C:/Users/aadar/Downloads/test_data/check_point")
    val dStream = streamingContext.socketTextStream("localhost", 21)
    val windowedCountDStream = dStream.window(Seconds(10), Seconds(5))
    windowedCountDStream.foreachRDD(rdd=>{
      println("Hello")
      rdd.saveAsTextFile("C:/Users/aadar/Downloads/test_data/output")
    })
    streamingContext.start()
    streamingContext.awaitTermination()
  }
}
```

Console

ReduceByKeyAndWindowExample\$ [Scala Application] C:\Program Files\Java\jre1.8.0_311\bin\javaw.exe (Dec 10, 2021, 1:35:24 PM)

```
21/12/18 13:35:42 INFO DStreamGraph: Updating checkpoint data for time 1639161342000 ms
21/12/18 13:35:42 INFO DStreamGraph: Updated checkpoint data for time 1639161342000 ms
21/12/18 13:35:42 INFO JobScheduler: Starting job streaming job 1639161342000 ms.0 from job set of time 1639161342000 ms
Hello
21/12/18 13:35:42 INFO CheckpointWriter: Submitted checkpoint of time 1639161342000 ms to writer queue
21/12/18 13:35:42 INFO CheckpointWriter: Saving checkpoint for time 1639161342000 ms to file 'file:/C:/Users/aadar/Downloads/test_data/check_point'
21/12/18 13:35:42 WARN CheckpointWriter: Error in attempt 1 of writing checkpoint to 'file:/C:/Users/aadar/Downloads/test_data/check_point'
java.io.IOException: (null) entry in command string: null chmod 0644 C:/Users/aadar/Downloads/test_data/check_point/temp
```

BDM_GroupA - N...

File Edit Format View Help

Group A

Participants

1. Aadarsha chapagain
2. Jyoti shukla
3. Rishi Phaneendra Varma
4. Priti Bhale
5. Sreya Treasa Johny
6. Piyush Bhatia

100% Windows (CRLF) UTF-8

Writable Smart Insert 22:4 495M of 746M

100% 5°C Mostly clo... 1:38 PM