**POM.xml**

<project xmlns=*"http://maven.apache.org/POM/4.0.0"* xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<groupId>com.training.section4</groupId>

<artifactId>section4</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>org.apache.beam</groupId>

<artifactId>beam-sdks-java-core</artifactId>

<version>2.20.0</version>

</dependency>

<dependency>

<groupId>org.apache.beam</groupId>

<artifactId>beam-runners-direct-java</artifactId>

<version>2.20.0</version>

</dependency>

<dependency>

<groupId>org.apache.beam</groupId>

<artifactId>beam-sdks-java-extensions-join-library</artifactId>

<version>2.20.0</version>

</dependency>

</dependencies>

</project>

DistinctExample.java

package section4;

import java.io.Serializable;

import java.util.ArrayList;

import java.util.List;

import org.apache.avro.generic.GenericRecord;

import org.apache.beam.sdk.Pipeline;

import org.apache.beam.sdk.coders.AvroCoder;

import org.apache.beam.sdk.coders.DefaultCoder;

import org.apache.beam.sdk.coders.SerializableCoder;

import org.apache.beam.sdk.coders.StringUtf8Coder;

import org.apache.beam.sdk.io.TextIO;

import org.apache.beam.sdk.schemas.annotations.DefaultSchema;

import org.apache.beam.sdk.transforms.Create;

import org.apache.beam.sdk.transforms.Distinct;

import org.apache.beam.sdk.transforms.DoFn;

import org.apache.beam.sdk.transforms.ParDo;

import org.apache.beam.sdk.transforms.DoFn.ProcessContext;

import org.apache.beam.sdk.transforms.DoFn.ProcessElement;

import org.apache.beam.sdk.transforms.Keys;

import org.apache.beam.sdk.values.KV;

import org.apache.beam.sdk.values.PCollection;

public class DistinctExample {

public static void main(String[] args) {

// TODO Auto-generated method stub

Pipeline p = Pipeline.create();

PCollection<String> pCustList = p.apply(TextIO.read().from("C:\\Beam\\Distinct.csv"));

PCollection<String> uniqueCust=pCustList.apply(Distinct.<String>create());

uniqueCust.apply(TextIO.write().to("C:\\Beam\\distinct\_out.csv").withNumShards(1).withSuffix(".csv"));

p.run();

}

}

CountExample.java

**package** section4;

**import** org.apache.beam.sdk.Pipeline;

**import** org.apache.beam.sdk.io.TextIO;

**import** org.apache.beam.sdk.transforms.Count;

**import** org.apache.beam.sdk.transforms.DoFn;

**import** org.apache.beam.sdk.transforms.ParDo;

**import** org.apache.beam.sdk.values.PCollection;

**public** **class** CountExample {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

Pipeline p = Pipeline.*create*();

PCollection<String> pCount = p.apply(TextIO.*read*().from("C:\\Beam\\Count.csv"));

PCollection<Long> pLong = pCount.apply(Count.*globally*());

pLong.apply(ParDo.*of*(**new** DoFn<Long, Void>() {

@ProcessElement

**public** **void** processElement(ProcessContext c) {

System.***out***.println(c.element());

}

}));

p.run();

}

}

GroupByKeyExample.java

**package** section4;

**import** org.apache.beam.sdk.Pipeline;

**import** org.apache.beam.sdk.io.TextIO;

**import** org.apache.beam.sdk.transforms.DoFn;

**import** org.apache.beam.sdk.transforms.GroupByKey;

**import** org.apache.beam.sdk.transforms.ParDo;

**import** org.apache.beam.sdk.values.KV;

**import** org.apache.beam.sdk.values.PCollection;

// STEP 2

**class** StringToKV **extends** DoFn<String,KV<String, Integer>> {

@ProcessElement

**public** **void** processElement(ProcessContext c) {

String input=c.element();

String arr[] = input.split(",");

c.output(KV.*of*(arr[0], Integer.*valueOf*(arr[3])));

}

}

//STEP 4

**class** KVToString **extends** DoFn<KV<String, Iterable<Integer>>,String> {

@ProcessElement

**public** **void** processElement(ProcessContext c) {

String strKey=c.element().getKey();

Iterable<Integer> vals=c.element().getValue();

Integer sum=0;

**for** (Integer integer : vals) {

sum=sum+integer;

}

c.output(strKey+","+sum.toString());

}

}

**public** **class** GroupByKeyExample {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

Pipeline p = Pipeline.*create*();

// Step 1

PCollection<String> pCustOrderList = p.apply(TextIO.*read*().from("C:\\Beam\\GroupByKey\_data.csv"));

// Step 2 : Convert String to KV

PCollection<KV<String,Integer>> kvOrder = pCustOrderList.apply(ParDo.*of*(**new** StringToKV()));

// Step 3 : Apply groupbyKey and build KV<String, Iterable<Integer>

PCollection<KV<String,Iterable<Integer>>> kvOrder2 = kvOrder.apply(GroupByKey.<String, Integer>*create*());

// Step 4 : Convert KV<String, Iterable<Integer> to String and write

PCollection<String> output = kvOrder2.apply(ParDo.*of*(**new** KVToString()));

output.apply(TextIO.*write*().to("C:\\Beam\\group\_by\_output.csv").withHeader("Id,Amount").withNumShards(1).withSuffix(".csv"));

p.run();

}

}