

### Question :

We have a file windowdata.csv and the field names are country, weeknum, numinvoices, totalquantity, invoicevalue

Step 1: create spark session

Step 2: set the logging level to error

Step 3: Using the standard dataframe reader API load the file and create a dataframe.

Step 4: Use the standard dataframe writer api to save it in parquet format. While saving make sure data is stored where we should have a folder for each country, weeknum (combination)

Step 5: Also use the dataframe write api to save the data in Avro format. While saving make sure data is stored where we should have a folder for each country.

Step 6: Apply header

Step 7: Convert dataframe to dataset(Specific type)

### Code :

```
import org.apache.log4j.{Level, Logger}
import org.apache.spark.sql.{SaveMode, SparkSession}

object WindowDataReadWrite extends App{

  case class
  Customer(country:String,weeknum:Int,numinvoices:Int,totalquantity:Int,invoice
  value:Double)

  Logger.getLogger("org").setLevel(Level.ERROR)

  val spark = SparkSession
    .builder.appName("Window Data")
    .master("local[*]")
    .getOrCreate()

  // Reading the file without headers
  val df = spark.read
    .format("csv")
    .option("inferSchema","true")

  .option("path","C://Users//deept//Desktop//Airisdata//datasets//windowdata.csv")
    .load()
```

```

    // Adding column names
    val dfWithColumns = df.toDF("country", "weeknum", "numinvoices",
    "totalquantity", "invoicevalue")

    // Displaying the data
    dfWithColumns.show(10,false);
    dfWithColumns.printSchema()

    // Reading the data with headers
    val dfHeader = spark
        .read
        .format("csv")
        .option("header","true")
        .option("inferSchema","true")

    .option("path","C://Users//deept//Desktop//Airisdata//datasets//windowdata_header.csv")
    .load()

    import spark.implicits._
    val dsHeader = dfHeader.as[Customer]

    // Displaying the data with header
    dfHeader.show(10,false);

    // Each folder with country, week num combination
    dfWithColumns
        .write
        .mode("overwrite")
        .format("parquet")
        .partitionBy("country","weeknum")

    .option("path","C://Users//deept//Desktop//Airisdata//datasets//windowDataWriteOutput")
    .save()

    dfWithColumns
        .write
        .mode("overwrite")
        .format("avro")
        .partitionBy("country")

    .option("path","C://Users//deept//Desktop//Airisdata//datasets//windowDataWriteOutput_avro")
    .save()

    spark.stop();
}



























```

## Output :




```
+-----+-----+-----+-----+-----+
|country|weeknum|numinvoices|totalquantity|invoicevalue|
+-----+-----+-----+-----+-----+
|Spain  |49      |1          |67            |174.72      |
|Germany|48      |11         |1795          |3309.75     |
|Lithuania|48     |3          |622           |1598.06     |
|Germany|49      |12         |1852          |4521.39     |
|Bahrain|51      |1          |54            |205.74      |
|Iceland|49      |1          |319           |711.79      |
|India  |51      |5          |95            |276.84      |
|Australia|50     |2          |133           |387.95      |
|Italy  |49      |1          |-2            |-17.0       |
|India  |49      |5          |1280          |3284.1      |
+-----+-----+-----+-----+-----+
only showing top 10 rows
```

```
root
|-- country: string (nullable = true)
|-- weeknum: integer (nullable = true)
|-- numinvoices: integer (nullable = true)
|-- totalquantity: integer (nullable = true)
|-- invoicevalue: double (nullable = true)
```



## Parquet Format :

Name ^	Date modified	Type	Size
 country=Australia	7/8/2022 12:39 PM	File folder	
 country=Austria	7/8/2022 12:39 PM	File folder	
 country=Bahrain	7/8/2022 12:39 PM	File folder	
 country=Belgium	7/8/2022 12:39 PM	File folder	
 country=Channel%20Islands	7/8/2022 12:39 PM	File folder	
 country=Cyprus	7/8/2022 12:39 PM	File folder	
 country=Denmark	7/8/2022 12:39 PM	File folder	
 country=Finland	7/8/2022 12:39 PM	File folder	
 country=France	7/8/2022 12:39 PM	File folder	
 country=Germany	7/8/2022 12:39 PM	File folder	
 country=Iceland	7/8/2022 12:39 PM	File folder	
 country=India	7/8/2022 12:39 PM	File folder	
 country=Israel	7/8/2022 12:39 PM	File folder	
 country=Italy	7/8/2022 12:39 PM	File folder	
 country=Japan	7/8/2022 12:39 PM	File folder	
 country=Lithuania	7/8/2022 12:39 PM	File folder	
 country=Netherlands	7/8/2022 12:39 PM	File folder	
 country=Norway	7/8/2022 12:39 PM	File folder	
 country=Poland	7/8/2022 12:39 PM	File folder	
 country=Portugal	7/8/2022 12:39 PM	File folder	
 country=Spain	7/8/2022 12:39 PM	File folder	
 country=Sweden	7/8/2022 12:39 PM	File folder	
 country=Switzerland	7/8/2022 12:39 PM	File folder	
 country=United%20Kingdom	7/8/2022 12:39 PM	File folder	
 ._SUCCESS.crc	7/8/2022 12:39 PM	CRC File	1 KB
 ._SUCCESS	7/8/2022 12:39 PM	File	0 KB



























Inside each Country's folder:

Name	Date modified	Type	Size
 weeknum=48	7/8/2022 12:01 PM	File folder	
 weeknum=49	7/8/2022 12:01 PM	File folder	
 weeknum=50	7/8/2022 12:01 PM	File folder	



Inside each weeknum folder:

Name	Date modified	Type	Size
 .part-00000-5c1de599-a0bb-4d3c-98ed-...	7/8/2022 12:28 PM	CRC File	1 KB
 part-00000-5c1de599-a0bb-4d3c-98ed-...	7/8/2022 12:28 PM	PARQUET File	1 KB

## Avro Format :

Name	Date modified	Type	Size
 country=Australia	7/8/2022 12:28 PM	File folder	
 country=Austria	7/8/2022 12:28 PM	File folder	
 country=Bahrain	7/8/2022 12:28 PM	File folder	
 country=Belgium	7/8/2022 12:28 PM	File folder	
 country=Channel%20Islands	7/8/2022 12:28 PM	File folder	
 country=Cyprus	7/8/2022 12:28 PM	File folder	
 country=Denmark	7/8/2022 12:28 PM	File folder	
 country=Finland	7/8/2022 12:28 PM	File folder	
 country=France	7/8/2022 12:28 PM	File folder	
 country=Germany	7/8/2022 12:28 PM	File folder	
 country=Iceland	7/8/2022 12:28 PM	File folder	
 country=India	7/8/2022 12:28 PM	File folder	
 country=Israel	7/8/2022 12:28 PM	File folder	
 country=Italy	7/8/2022 12:28 PM	File folder	
 country=Japan	7/8/2022 12:28 PM	File folder	
 country=Lithuania	7/8/2022 12:28 PM	File folder	
 country=Netherlands	7/8/2022 12:28 PM	File folder	
 country=Norway	7/8/2022 12:28 PM	File folder	
 country=Poland	7/8/2022 12:28 PM	File folder	
 country=Portugal	7/8/2022 12:39 PM	File folder	
 country=Spain	7/8/2022 12:39 PM	File folder	
 country=Sweden	7/8/2022 12:39 PM	File folder	
 country=Switzerland	7/8/2022 12:39 PM	File folder	
 country=United%20Kingdom	7/8/2022 12:39 PM	File folder	
 _SUCCESS.crc	7/8/2022 12:39 PM	CRC File	1 KB
 _SUCCESS	7/8/2022 12:39 PM	File	0 KB

Inside country folder :

Name	Date modified	Type	Size
 .part-00000-db9b0b30-86d1-4b4f-ad73-...	7/8/2022 12:28 PM	CRC File	1 KB
 part-00000-db9b0b30-86d1-4b4f-ad73-...	7/8/2022 12:28 PM	AVRO File	1 KB