{

"type": "record",

"name": "Person",

"fields": [

{"name": "name", "type": "string"},

{"name": "age", "type": "int"},

{"name": "city", "type": "string"}

]

}

import org.apache.avro.Schema;

import org.apache.avro.file.DataFileWriter;

import org.apache.avro.generic.GenericData;

import org.apache.avro.generic.GenericRecord;

import org.apache.avro.io.DatumWriter;

import org.apache.avro.io.EncoderFactory;

import org.apache.avro.io.JsonEncoder;

import org.apache.avro.specific.SpecificDatumWriter;

import java.io.File;

import java.io.IOException;

public class AvroExample {

public static void main(String[] args) throws IOException {

// Define Avro schema

Schema schema = new Schema.Parser().parse("{\"type\":\"record\",\"name\":\"Person\",\"fields\":[{\"name\":\"name\",\"type\":\"string\"},{\"name\":\"age\",\"type\":\"int\"},{\"name\":\"city\",\"type\":\"string\"}]}");

// Create a GenericRecord representing a Person

GenericRecord person = new GenericData.Record(schema);

person.put("name", "John Doe");

person.put("age", 25);

person.put("city", "ExampleCity");

// Serialize the GenericRecord to a file using Avro

serializeAvroRecord(person, schema, "person.avro");

System.out.println("Avro serialization completed.");

}

private static void serializeAvroRecord(GenericRecord record, Schema schema, String filePath) throws IOException {

// Create a DatumWriter for the Avro serialization

DatumWriter<GenericRecord> datumWriter = new SpecificDatumWriter<>(schema);

// Create a DataFileWriter to write Avro data to a file

try (DataFileWriter<GenericRecord> dataFileWriter = new DataFileWriter<>(datumWriter)) {

dataFileWriter.create(schema, new File(filePath));

// Write the Avro record to the file

dataFileWriter.append(record);

}

}

}

Sequence file

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.fs.Path;

import org.apache.hadoop.io.SequenceFile;

import org.apache.hadoop.io.Text;

public class SequenceFileExample {

public static void main(String[] args) throws Exception {

Configuration conf = new Configuration();

// Create a SequenceFile Writer

SequenceFile.Writer writer = SequenceFile.createWriter(conf,

SequenceFile.Writer.file(new Path("int\_pairs.seq")),

SequenceFile.Writer.keyClass(Text.class),

SequenceFile.Writer.valueClass(IntPairWritable.class));

// Write some data to the SequenceFile

writer.append(new Text("pair1"), new IntPairWritable(10, 20));

writer.append(new Text("pair2"), new IntPairWritable(30, 40));

// Close the writer

writer.close();

// You can now read this SequenceFile using a SequenceFile Reader

}

}