800+ Q&As | Logout | Contact

Java-Success.com

Prepare to fast-track, choose & go places with 800+ Java & Big Data Q&As with lots of code & diagrams.

search here … Go

Home Why? ▾ 300+ Java FAQs ▾ 300+ Big Data FAQs ▾ Courses ▾

Membership ▾ Your Career ▾

Home › bigdata-success.com › Tutorials - Big Data › TUT - File Formats › 07: Avro IDL (e.g. avdl) to Java objects & Avro Schemas (i.e. avsc) tutorial

07: Avro IDL (e.g. avdl) to Java objects & Avro Schemas (i.e. avsc) tutorial

Posted on June 27, 2017

Avro IDL (i.e Interface Definition Language) schema can be specified with two type of files “avpr” (i.e. AVro PRotocol file) & “avdl” (i.e. AVro iDL).

Step 1: Create a maven based Java project from a command-line

```
$ mvn archetype:generate -B -DgroupId=com.mytutori
```

300+ Java Interview FAQs

300+ Java FAQs 🔥

16+ Java Key Areas Q&As

150+ Java Architect FAQs

80+ Java Code Quality Q&As

150+ Java Coding Q&As

300+ Big Data Interview FAQs

300+ Big Data FAQs 🔥

Tutorials - Big Data

TUT - ⏯ Starting Big Data

TUT - Starting Spark & Scala

Step 2: Import it into eclipse as a maven project.

Step 3: Create new sources folder “**src/main/resources**”

Step 4: Create a new folder “**avro**” inside “**src/main/resources**“.

Step 5: Create a new file “**simple.avdl**” inside the folder “avro”.

```
@namespace("com.mytutorial")
protocol Simple {

    enum CountryCodeType {
        AU, US, IN
    }

    record Role {
        string roleId;
        string roleName;
    }

    record User {
        string id;
        string firstName;
        string lastName;
        union { null, string } middleName;

        array<Role> roles;
        CountryCodeType country;

    }
}

```

Step 6: The “ConvertAvdlToAvsc.java” to convert “**.avdl**” to “**.avsc**“.

```
package com.mytutorial;

import java.io.File;
import java.util.ArrayList;
import java.util.List;
```

TUT - Starting with Python
TUT - Kafka
TUT - Pig
TUT - Apache Storm
TUT - Spark Scala on Zeppelin
TUT - Cloudera
TUT - Cloudera on Docker
TUT - File Formats
TUT - Spark on Docker
TUT - Flume
TUT - Hadoop (HDFS)
TUT - HBase (NoSQL)
TUT - Hive (SQL)
TUT - Hadoop & Spark
TUT - MapReduce
TUT - Spark and Scala
TUT - Spark & Java
TUT - PySpark on Databricks
TUT - Zookeeper

800+ Java Interview Q&As
300+ Core Java Q&As
300+ Enterprise Java Q&As
150+ Java Frameworks Q&As
120+ Companion Tech Q&As
Tutorials - Enterprise Java

```java

import org.apache.avro.tool.IdlToSchemataTool;

/**
 * Utility to convert avdl to avsc
 * (suitable for use in Hive tables).
 */
public class ConvertAvdlToAvsc {
    public static void main(String [] args) throws
        IdlToSchemataTool tool = new IdlToSchemat

        File inDir = new File(args[0]);
        File outDir = new File(args[1]);

        for (File inFile: inDir.listFiles()) {
            List<String> toolArgs = new ArrayList
            toolArgs.add(inFile.getAbsolutePath()
            toolArgs.add(outDir.getAbsolutePath()

            tool.run(System.in, System.out, Syste
        }
    }
}

```

Step 7: Finally, the pom.xml file with relevant dependency jars & plugins to generate:

1) Avro representation as POJOs to perform serialization to a file.

2) Avro schemas (i.e .avsc files).

```xml
<project xmlns="http://maven.apache.org/POM/4.0.0
    xsi:schemaLocation="http://maven.apache.org/P
    <modelVersion>4.0.0</modelVersion>
    <groupId>com.mytutorial</groupId>
    <artifactId>simple-avro</artifactId>
    <packaging>jar</packaging>
    <version>1.0-SNAPSHOT</version>
    <name>simple-avro</name>
    <url>http://maven.apache.org</url>

    <dependencies>
        <dependency>
            <groupId>junit</groupId>
            <artifactId>junit</artifactId>
            <version>4.12</version>
            <scope>test</scope>
        </dependency>
        <dependency>
            <groupId>org.apache.avro</groupId>
            <artifactId>avro</artifactId>
            <version>1.7.6-cdh5.8.3</version>
            <scope>provided</scope>
        </dependency>
        <dependency>
            <groupId>org.apache.avro</groupId>
            <artifactId>avro-tools</artifactId>
            <version>1.7.6-cdh5.8.3</version>
            <scope>provided</scope>
        </dependency>
    </dependencies>

    <build>
        <plugins>
            <!--Generates Java objects (i.e. *.ja
            <plugin>
                <groupId>org.apache.avro</groupId
                <artifactId>avro-maven-plugin</ar
                <version>1.7.6</version>
                <configuration>
                    <stringType>String</stringTyp
                </configuration>
                <executions>
                    <execution>
                        <phase>generate-sources</
                        <goals>
                            <goal>idl-protocol</g
                        </goals>
                        <configuration>
                            <includes>
                                <include>simple.a
                            </includes>
                            <sourceDirectory>${pr
                            <outputDirectory>${pr
                        </configuration>
                    </execution>
                </executions>
            </plugin>
            <!--Generates avro schema (i.e. .avsc
            <plugin>
                <groupId>org.codehaus.mojo</group
                <artifactId>exec-maven-plugin</ar
                <version>1.3</version>
                <executions>
                    <execution>
                        <phase>generate-sources</
                        <goals>
                            <goal>java</goal>
                        </goals>
                    </execution>
                </executions>
                <configuration>
```

```
                    <classpathScope>compile</clas
                    <includeProjectDependencies>tr
                    <includePluginDependencies>tru
                    <mainClass>com.mytutorial.Conv
                    <arguments>
                        <argument>${project.basedi
                        <argument>${project.basedi
                    </arguments>
                </configuration>
            </plugin>

        </plugins>
    </build>

    <repositories>
        <repository>
            <id>cloudera</id>
            <url>http://repository.cloudera.com/ar
        </repository>
    </repositories>
</project>


```

Step 8: Run as “mvn clean install” to generate the relevant AVRO artefacts.

Generated avro schemas

User.avsc

```

{
  "type" : "record",
  "name" : "User",
  "namespace" : "com.mytutorial",
  "fields" : [ {
    "name" : "id",
    "type" : "string"
  }, {
    "name" : "firstName",
    "type" : "string"
  }, {
    "name" : "lastName",
    "type" : "string"
  }, {
    "name" : "middleName",
    "type" : [ "null", "string" ]
  }, {
    "name" : "roles",
    "type" : {
      "type" : "array",
      "items" : {
        "type" : "record",
        "name" : "Role",
        "fields" : [ {
          "name" : "roleId",
          "type" : "string"
        }, {
          "name" : "roleName",
          "type" : "string"
        } ]
      }
    }
  }, {
    "name" : "country",
    "type" : {
      "type" : "enum",
      "name" : "CountryCodeType",
      "symbols" : [ "AU", "US", "IN" ]
    }
  } ]
}
```

Role.avsc

```
{
  "type" : "record",
  "name" : "Role",
  "namespace" : "com.mytutorial",
  "fields" : [ {
    "name" : "roleId",
    "type" : "string"
  }, {
    "name" : "roleName",
    "type" : "string"
  } ]
}
```

CountryCodeType.avsc

```
{
  "type" : "enum",
  "name" : "CountryCodeType",
  "namespace" : "com.mytutorial",
  "symbols" : [ "AU", "US", "IN" ]
}
```

‹ 11 Spring MVC Interview Q&As

03: Spring Boot Web & Actuator Beginner Tutorial Step by Step ›

Disclaimer

The contents in this Java-Success are copyrighted and from EmpoweringTech pty ltd. The EmpoweringTech pty ltd has the right to correct or enhance the current content without any prior notice. These are general advice only, and one needs to take his/her own circumstances into consideration. The EmpoweringTech pty ltd will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. Links to external sites do not imply endorsement of the linked-to sites. Privacy Policy