1. Graphical user interface, text

   Description automatically generated
2. 
3. In the previous lecture, we learnt about JSON serialization.
4. **Agenda**:
   1. In this lecture, we will learn about AVRO Serialization.
5. 
6. Avro is very much a Standard and very much accepted for the community for Big Data Project.
7. If you prefer to use Avro, you can quickly generate POJO from an Avro Schema with the same kind of simplicity as we learnt from JSON.
8. 
9. We will use the same Event Model to generate Avro schema.  
   Table

   Description automatically generated
10. However, there is a small limitation which requires us to change remodel our schema.
11. In JSON Schema to POJO last lecture, we created four schema definitions.  
    A screenshot of a computer

    Description automatically generated  
      
    A screenshot of a computer

    Description automatically generated
12. Let’s create a solution.
13. We want to define these objects using simple schema definition language and automatically generate serializable classes.
14. Let’s create schema definition for those 3 classes using Avro Schema Syntax.
15. Graphical user interface, text, application

    Description automatically generated
16. Graphical user interface, application

    Description automatically generated
17. Graphical user interface, text, application

    Description automatically generated  
    We are done with Avro schemas for those 3 model classes.
18. Now need to create POJO classes from these Avro Schemas.  
    For this we can use Avro Maven Plugin.  
      
      
    The build will generate Avro source code and you need Avro package to compile the generated code.  
      
    Graphical user interface, text, application

    Description automatically generated  
      
    Now let’s jump to the maven lifecycle and compile the project.  
    Graphical user interface, text, application, email

    Description automatically generated  
    We’re done creating the POJO from Avro Schema.  
    But still we don’t know how to use the Avro produced POJOs in Kafka Producers.

Downloadable is attached with the lec and besides the note itself look for **06-avro-to-pojo.zip**