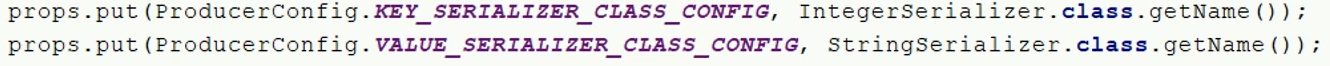
1. Text

   Description automatically generated  
   
2. **Agenda**:
   1. Kafka Serializer
3. 
4. The KafkaProducer is supposed to transmit the ProducerRecord over the network.  
   However, it doesn’t immediately transmit the record.
5. A screenshot of a computer

   Description automatically generated
6. Every record goes through
   1. Serialization:
   2. Partition.
   3. Then it is kept in Buffer.
7. **Serialization**:
   1. The Serialization is necessary to send the data over the network. Without it, Kafka doesn’t know how to transmit the data to the remote location.
   2. That is why specifying a key and value serializer is mandatory.  
        
      That is what we pass as part of KafkaProducer configuration.
   3. Above we have defined integer serializer for key and String Serializer for the value, but these are the most elementary serializer and don’t cover most of the use cases.
   4. In real life scenario, events (msgs) are represented by a complex java object not like the above as string.
   5. These objects must be serialized before the Kafka Producer can transmit to the Broker.
   6. Kafka gives you an option to use a **generic serialization** library like **Avro or Thrift**.  
      However, you have choice to create **custom serializer**.
   7. In the upcoming lecture, we will talk about JSON serializer and will show the process of creating and using a custom serializer.