1. **Agenda**:
   1. This section will cover **several components** which make up JMS 1.X API.
2. **Components**: Easy to understand the components once we understand the relations among these classes (**Components**)

|  |  |
| --- | --- |
| **ConnectionFactory** |  |
|  | 1. It is provided by the JMS Provider here like ActiveMQ. 2. JMS Provider (ActiveMQ) will put it into **JNDI Registry**. |
| **Destination** |  |
|  | 1. It can be **a Topic (Pub-Sub System) or a Queue (P2P)**. 2. We ask JMS Provider (ActiveMQ) to create it. 3. Then ActiveMQ will put it under **JNDI registry**. 4. We access it from JNDI Registry. |
| This is very 1st step in any JMS Client Code/App to create ConnectionFactory and Destination (Topic/Queue) and put in **JDNI Registry**. | |
| **Connection** | Using **ConnectionFactory**, we create **Connection**. Consider it physical connection with JMS Provider (Here ActiveMQ). |
| **Session** | From **Connection**, we get **Session**. We can create any # of sessions using single connection. |
| **Message** | It can be created independently. |
| **MessageProducer** | MessageProducer can be created using a session. If it is queue, then we will get **QueueProducer** or if it is a topic, then we will get **TopicProducer**. RED: It is read as not sure if it is QueueProducer class or not mentor said “Queue Producer” |
| **MessageConsumer** | MessageConsumer can be created using a session. If it is queue, then we will get **QueueConsumer** or if it is a topic, then we will get **TopicConsumer**. |