**Processing JSON using Jackson**

* Though there are three different ways to process JSON

1. Streaming API

(It’s is analogous to SAX parser for XML)

1. Tree Model

(It is analogous to DOM parser for XML)

1. Data Binding

(It converts JSON to and from Java Bean using property accessor or using annotations)

* “org.codehous.jackson.map.ObjectMapper” it’s a concrete class it helps us to Serialize/Marshall and De-serialize/Unmarshall regular Java Objects (POJOs or Java Beans)
* **There should be ONLY ONE object of “ObjectMapper” should exists per application. In other words it should be accessed in “Singleton” way.**

**NOTE: “ObjectMapper” is not a Singleton Class. (Same is applicable for JAXBContext)**

* The most commonly used methods of the ObjectMapper are

1. writeValue()
2. readValue()
3. writeValueAsString()

* **writeValue() Methods**
* There are 5 Overloaded Versions writeValue() Methods and all of them are Non-Static methods
* They helps us to convert Java Object to JSON
* return type of these methods is “void”
* All of these methods take Objects to be converted as input argument (as a 2nd input argument).
* **readValue() Methods**
* There are various Overloaded Versions readValue() Methods and all of them are Non-Static methods
* They helps us to convert JSON to Java Object
* return type of these methods is “Object of the Desired Class”
* All of these methods takes JSON (in various forms) as input argument.
* **writeValuesAsString() Method**
* There are NO Overloaded versions of writeValueAsString() exists
* It’s a Non-static method
* It convert Java Object to JSON & return as “String”