```
======
JSON
======

=> JSON stands for Java Script Object Notation

=> JSON represents data in key-value format

Ex:

        {
                "id" : 101,
                "name" : "Ashok",
                "phno" : 12345,
                "gender" : "Male"
        }

=> JSON is very light weight

=> JSON is platform independent & language independent

=> JSON is used to transfer data over a network

=> Distributed applications will use JSON data for request & response

=========================================
Working with JSON in Java Applications
=========================================

=> Java Applications can use JSON data

=> To work with JSON in Java app, we have third party libraries

                a) Jackson
                b) Gson

=> By using above libraries we can convert JSON data to Java Object and vice versa


                                        jackson/gson
                    java obj <--------------------> json data



=> The process of converting java obj to json is called as Serialization.

=> The process of converting json data to java obj is called as de-serialization.

==============
Jackson API
==============

1) Add below dependency in pom.xml file

                <dependency>
                        <groupId>com.fasterxml.jackson.core</groupId>
                        <artifactId>jackson-databind</artifactId>
                        <version>2.17.0</version>
                </dependency>

2) Create binding class to represent data

public class Customer {

        private Integer id;
        private String name;
        private Long phno;
```

```
     }

  3) Develop java class convert java to json and json to java

                  ObjectMapper mapper = new ObjectMapper();

                  mapper.writeValue(File f, Object obj);

                  mapper.readValue(File f, Class clz);


  ========================================================================================
  public class App {

          public static void main(String[] args) throws Exception {
                  App a = new App();
                  //a.convertJavaToJson();
                  a.convertJsonToJava();
          }

          public void convertJsonToJava() throws Exception{
                  ObjectMapper mapper = new ObjectMapper();
                  Customer c = mapper.readValue(new File("customer.json"), Customer.class);
                  System.out.println(c);
          }

          public void convertJavaToJson() throws Exception {
                  Customer c = new Customer(101, "John", 99853966771);
                  ObjectMapper mapper = new ObjectMapper();
                  mapper.writeValue(new File("customer.json"), c);
                  System.out.println("Serialization completed...");
          }
  }


  =========================================================================================

  ===========
  GSON API
  ===========

  ### Git Hub Rep : https://github.com/ashokitschool/GSON-JSON-APP.git

                  ---------------------------------------------
                  <dependency>
                          <groupId>com.google.code.gson</groupId>
                          <artifactId>gson</artifactId>
                          <version>2.3.1</version>
                  </dependency>
                  ---------------------------------------------

                  Gson gson  = new Gson ( );

                  gson.toJson(Object obj);

                  gson.fromJson(File file);
```