Handling JSON Files

- JSON
  - JSON stands for Java Script Object Notation
  - Simular to XML, JSON is used to transport the data between two parties
    - JSON is popular as it is the latest, light weight, advanced and faster than XML
  - Create an Example JSON file holding Employee data - View here
- Demonstrate reading the locators from the JSON files
  - Create a new Maven project
  - Create 'OR' folder and create a JSON file under it say 'LoginObjects.json'
  - Create locators under Home, Login and Account page in the LoginObjects.json file - View here
  - Add the following dependencies to the pom.xml file
    - JSON Smart - https://mvnrepository.com/artifact/net.minidev/json-smart
    - SLF4J API - https://mvnrepository.com/artifact/org.slf4j/slf4j-api
    - JSON Path - https://mvnrepository.com/artifact/com.jayway.jsonpath/json-path
    - GSON - https://mvnrepository.com/artifact/com.google.code.gson/gson
  - Create a new Java class say Demo and write the program which parses the JSON file, gets the locators in the object repository and prints - Demonstrate here
  - Demonstrate using these JSON objects in Selenium to verify the Login scenario - Demonstrate here
    - Add Selenium dependencies to the pom.xml file - https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java
    - Create a reusable method for converting the JSON objects in above program to WebElements
    - Call the reusable method and perform operations