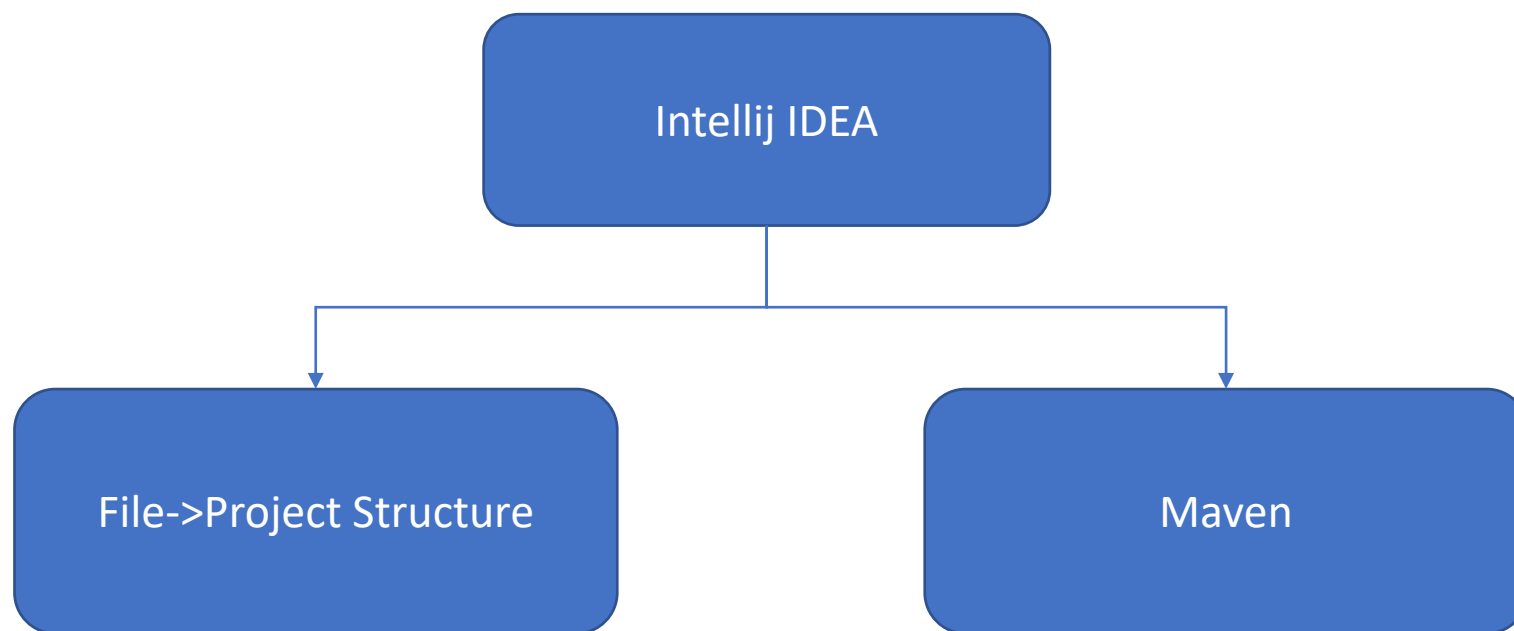


# Dasturga JDK dan boshqa kutubxonalarni ulash

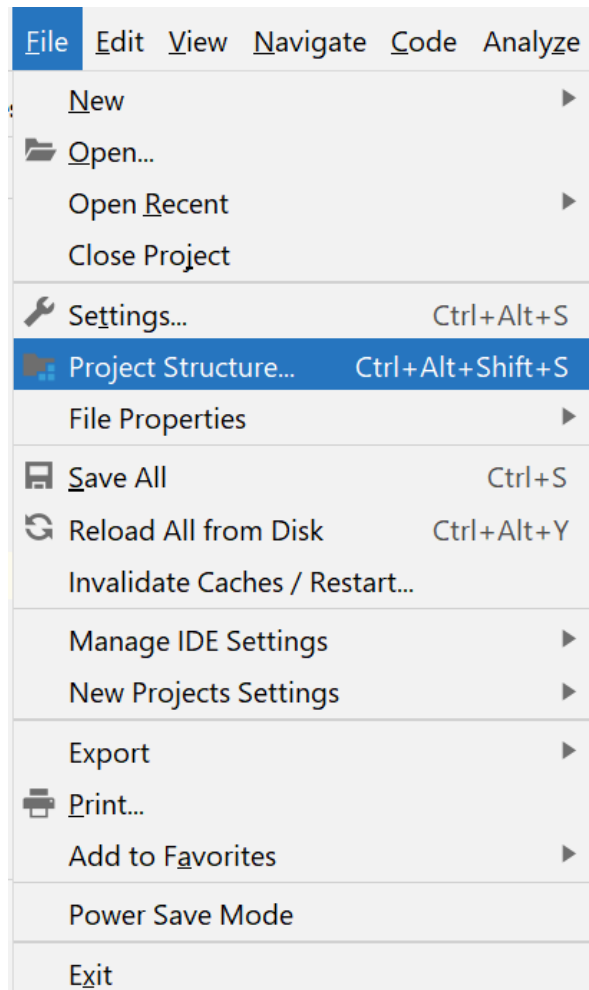
# Reja:

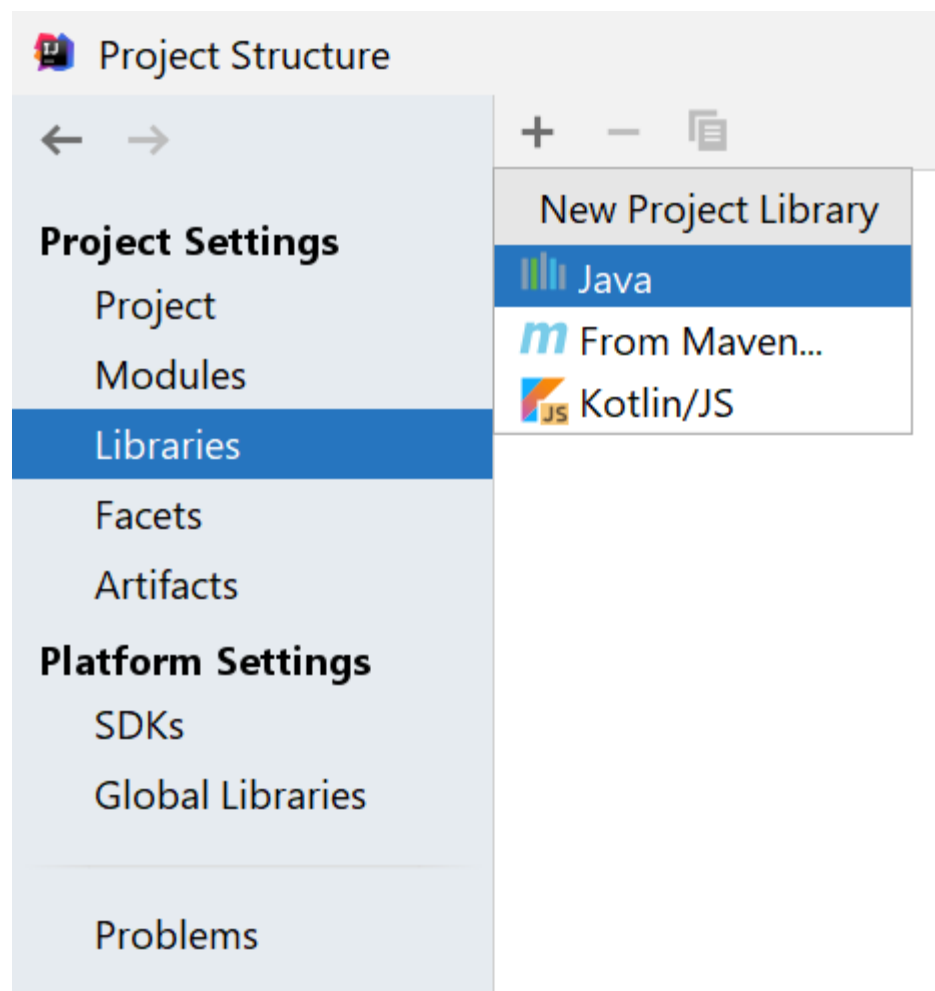
1. Dasturga tashqi kutubxonalarni ulash
2. Maven texnologiyasi
3. Json
4. Gson kutubxonasi
5. Project Lombok

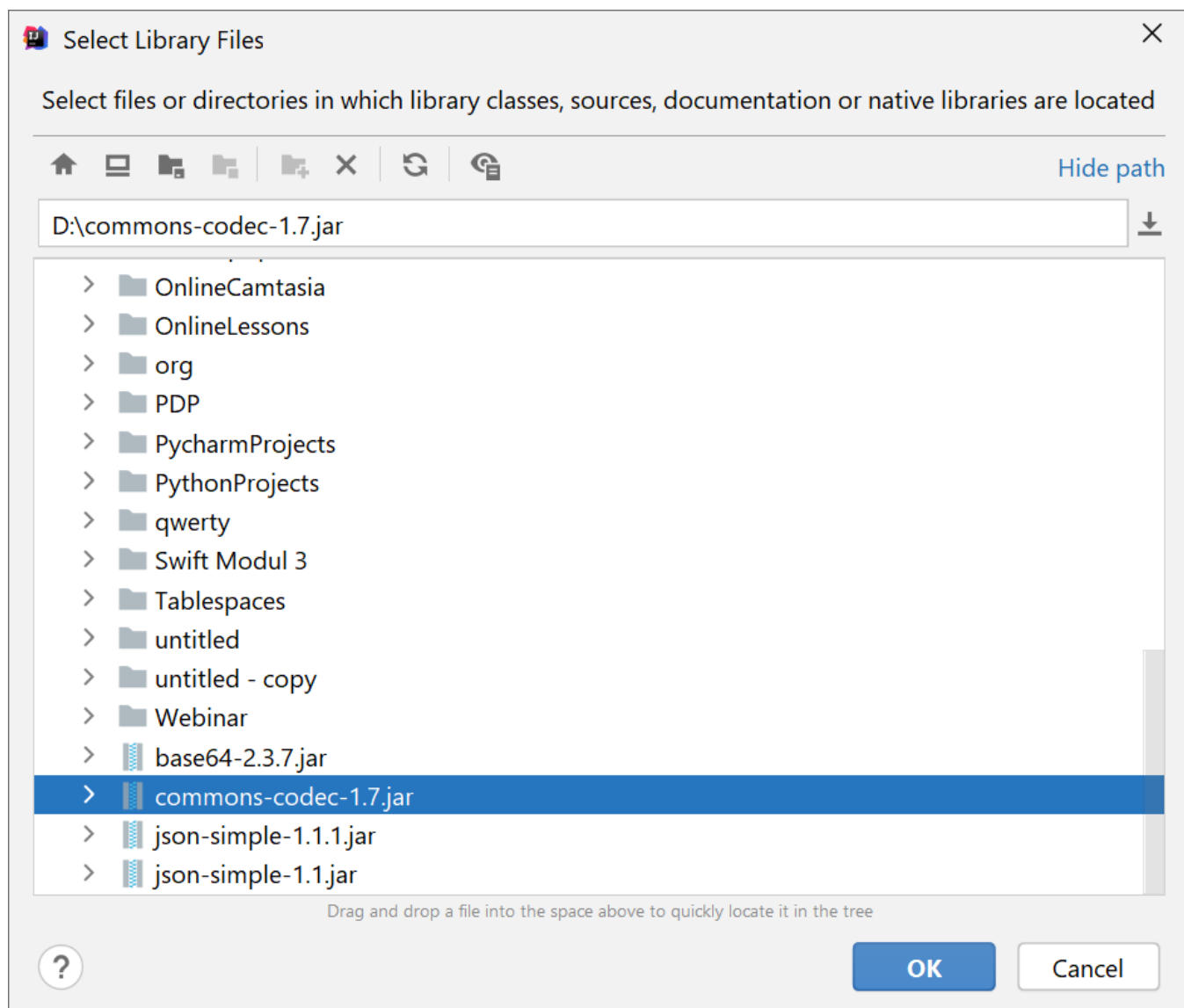
# Dasturga tashqi kutubxonalarni ulash



# File->Project Structure







# Maven texnologiyasi

- Maven haqida
- Maven yordamida project yaratish
- Pom.xml
- Dependency

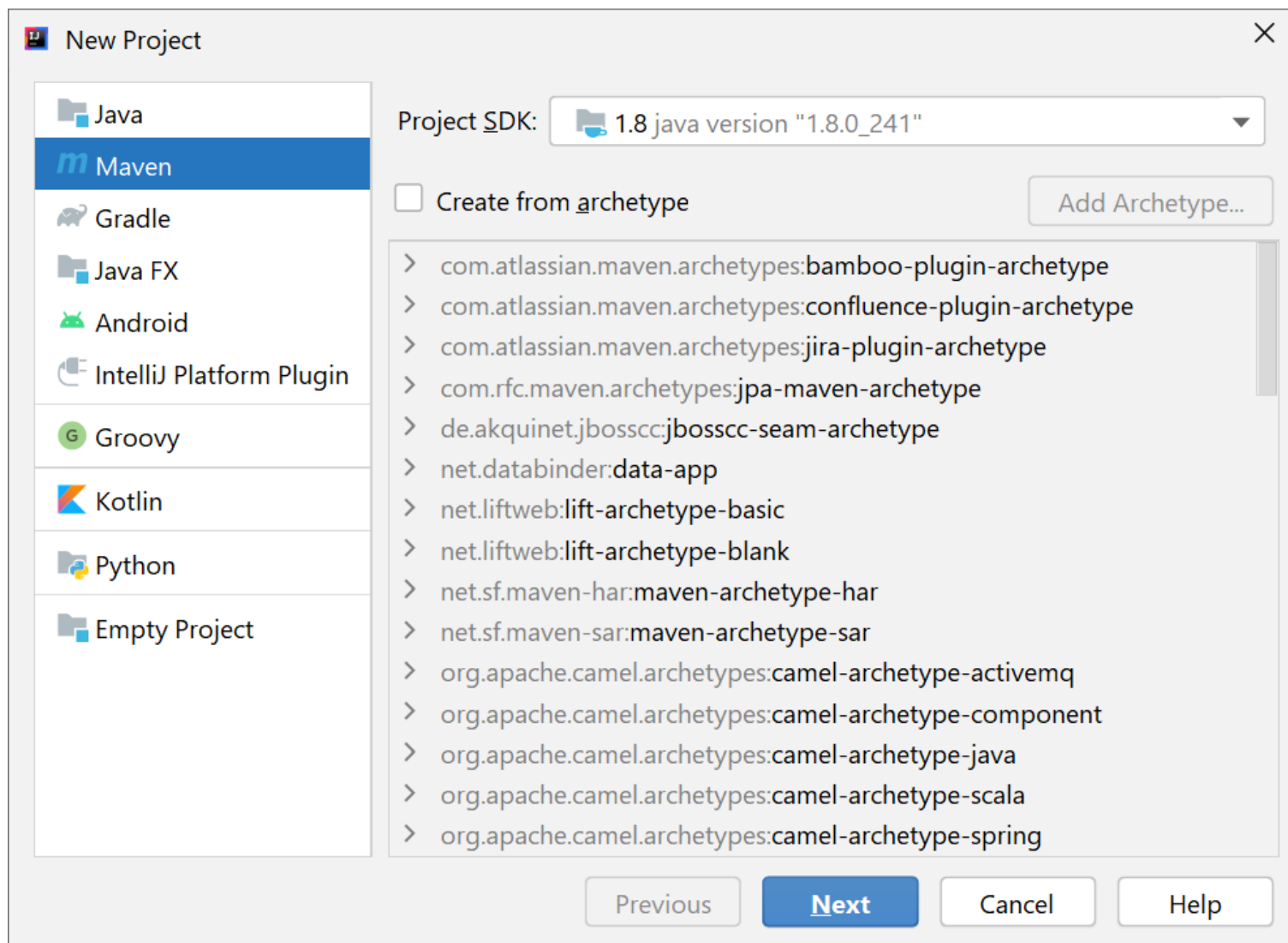


# Maven haqida

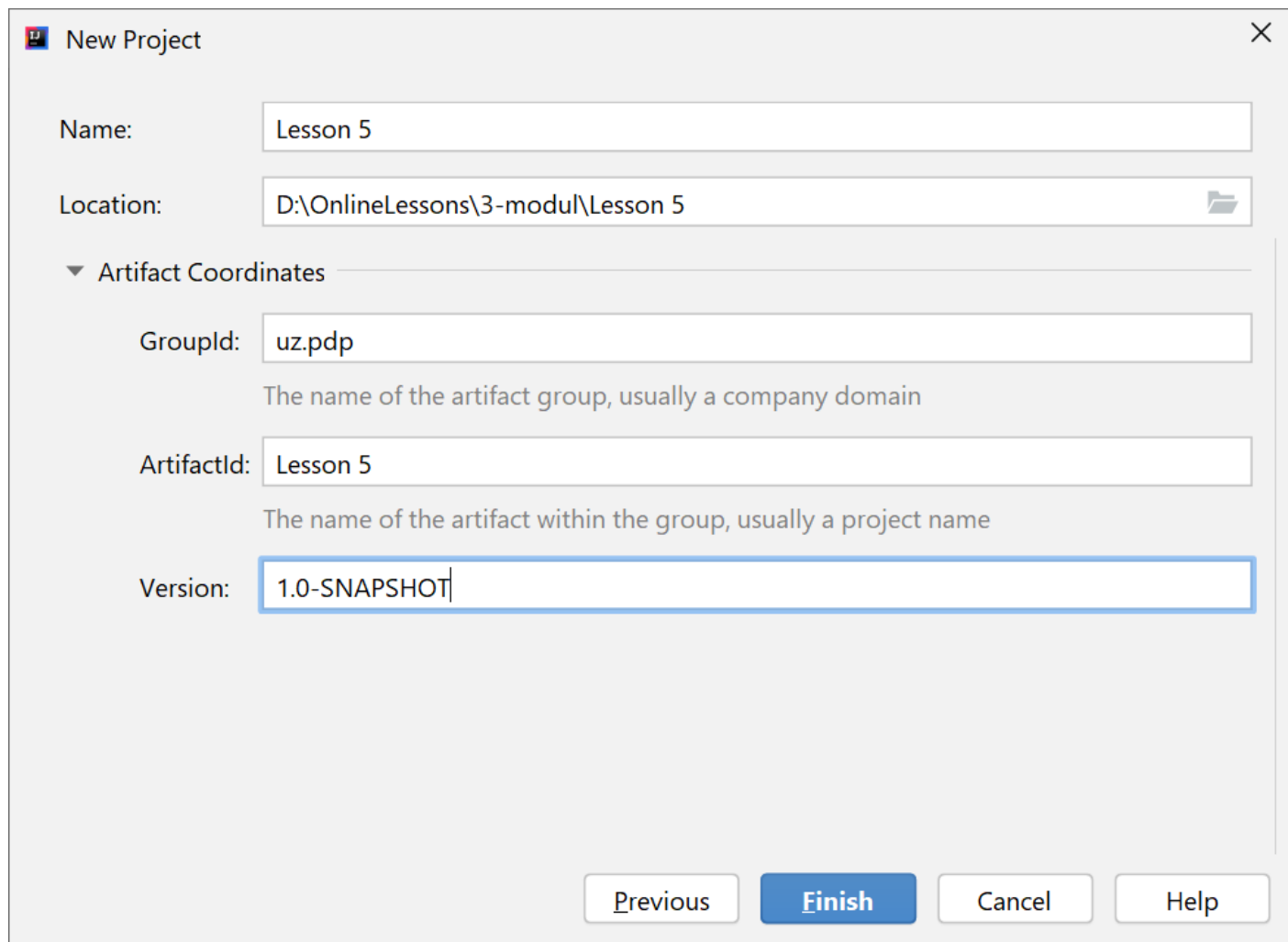
- Maven- projectni yig'ishni avtomatlashtiruvchi instrumentdir. Maven bilan asosan java developerlar ishlashadi. Shuningdek C++, Ruby, Scala, PHP va boshqa dasturlash tillari uchun ham integrasiyalari mavjud.
- Aslida kichik projectlarni commandalar satri orqali ham yig'ish mumkin. Lekin Projectni hajmi kattalashgan sari unda ishlatiladigan kutubxonalar ham ko'payib boradi va projectni kommandalar satri orqali yig'ish ma'lum bir qiyinchiliklarni keltirib chiqaradi. Maven aynan shu ishlarni yengillashtirish uchun yaratilgan texnologiyadir.



# Maven yordamida project yaratish



# Maven yordamida project yaratish



New Project

Name: Lesson 5

Location: D:\OnlineLessons\3-modul\Lesson 5

▼ Artifact Coordinates

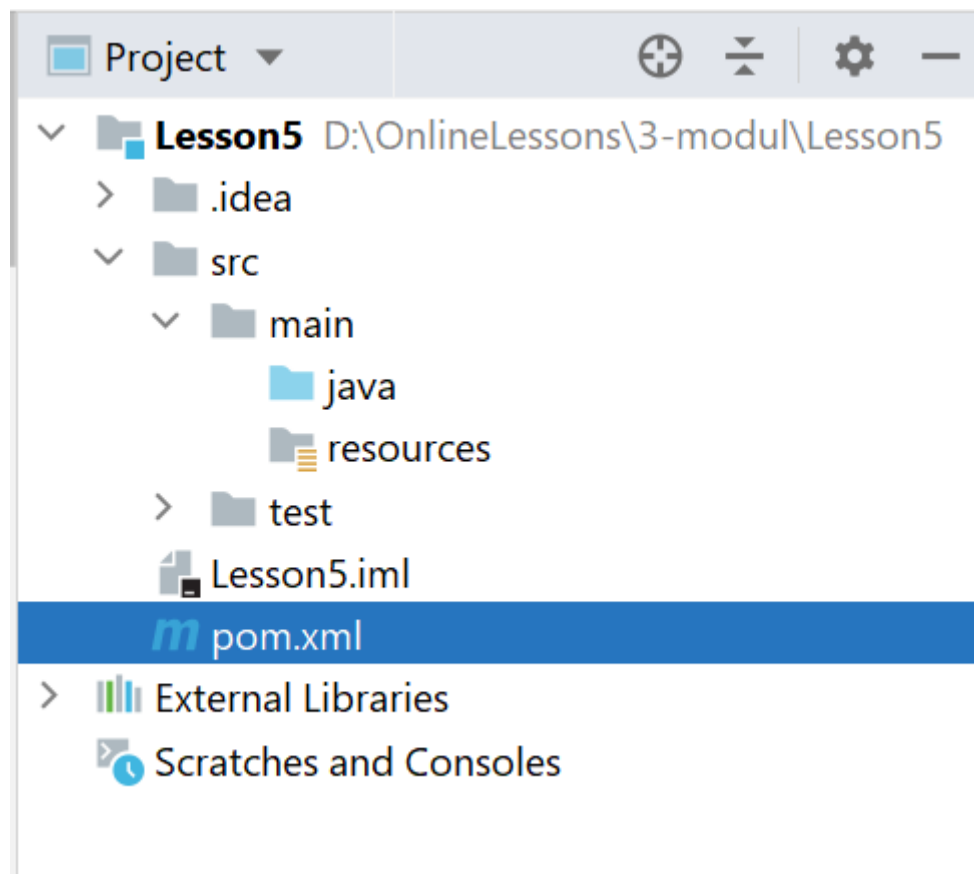
GroupId: uz.pdp  
The name of the artifact group, usually a company domain

ArtifactId: Lesson 5  
The name of the artifact within the group, usually a project name

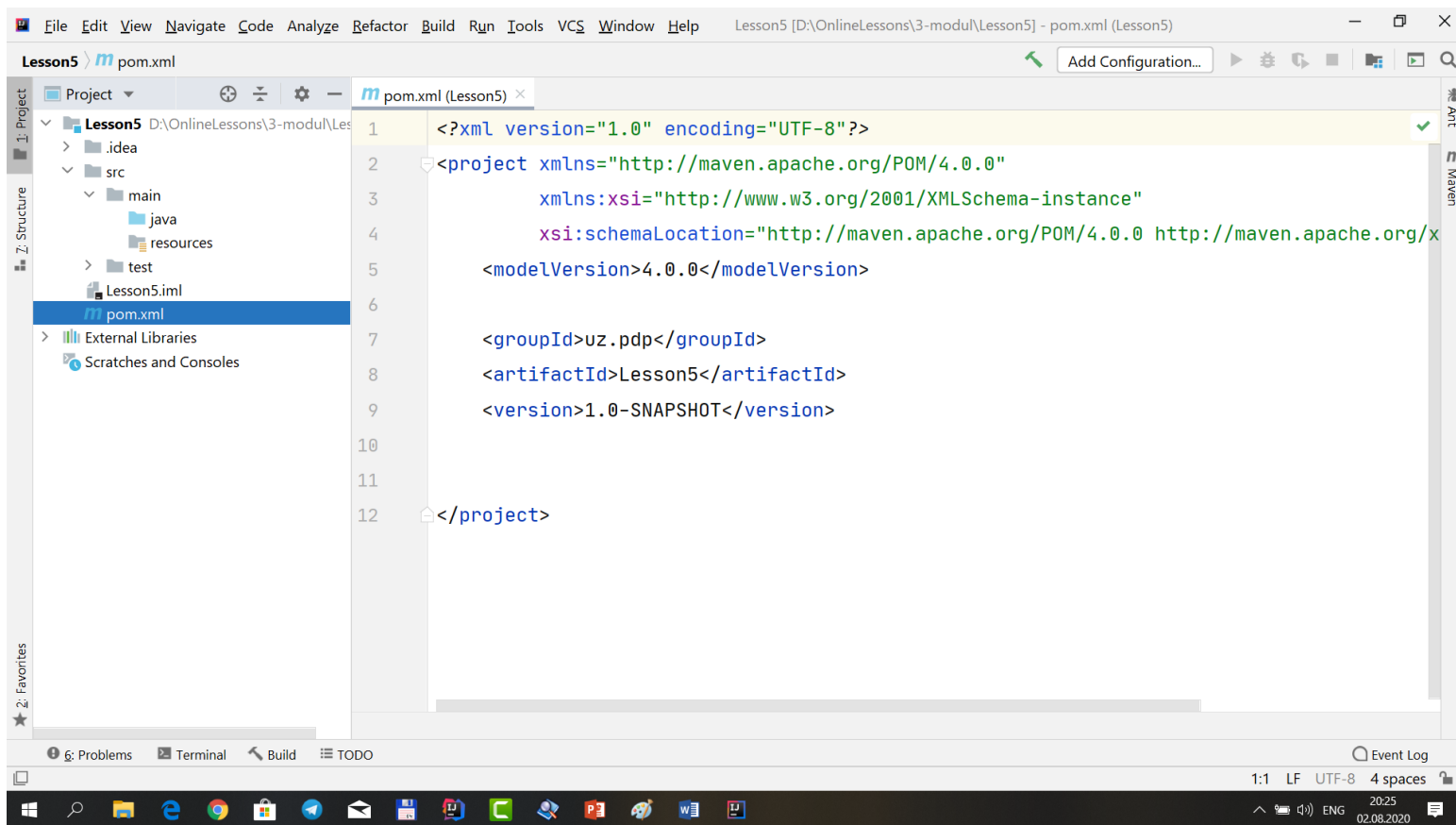
Version: 1.0-SNAPSHOT

Previous Finish Cancel Help

# Project structure



# Pom.xml



# Dependency

Mavenda tashqi kutubxonalar qo'shish pom.xml faylida dependencies tagi orqali amalga oshiriladi.

Har bir kutubxona dependency lar orqali ko'rsatiladi. Java bo'yicha ko'pchilik kutubxonlar <https://mvnrepository.com/> markaziy repositoryda saqlanadi.

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>uz.pdp</groupId>
  <artifactId>Lesson5</artifactId>
  <version>1.0-SNAPSHOT</version>
  <dependencies>
    <!-- https://mvnrepository.com/artifact/commons-codec/commons-codec -->
    <dependency>
      <groupId>commons-codec</groupId>
      <artifactId>commons-codec</artifactId>
      <version>1.14</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/com.google.code.gson/gson -->
    <dependency>
      <groupId>com.google.code.gson</groupId>
      <artifactId>gson</artifactId>
      <version>2.8.6</version>
    </dependency>
  </dependencies>
</project>
```

# Json

- JavaScript Object Notation
- Minimal
- Textual
- Subset of JavaScript



# Json

- A Subset of ECMA-262 Third Edition.
- Language Independent.
- Text-based.
- Light-weight.
- Easy to parse.



# Json

- Chinese
- English
- French
- German
- Italian
- Japanese
- Korean

# Json

- ActionScript
- C / C++
- C#
- Cold Fusion
- Delphi
- E
- Erlang
- Java
- Lisp
- Perl
- Objective-C
- Objective CAML
- PHP
- Python
- Rebol
- Ruby
- Scheme
- Squeak

# Json Syntax

JSON syntax JavaScript object syntax dan keb chiqqan holda olingan:

- Data is in name/value pairs => ma'lumot bu name/value juftligi
- Data is separated by commas => ma'lumotlar “,” bilan ajratiladi
- Curly braces hold objects => objectlar { } bilan ifodalanadi
- Square brackets hold arrays => arraylar [ ] bilan ifodalanadi

# Json Element

Json Data – name/value juftligi shaklida yoziladi.

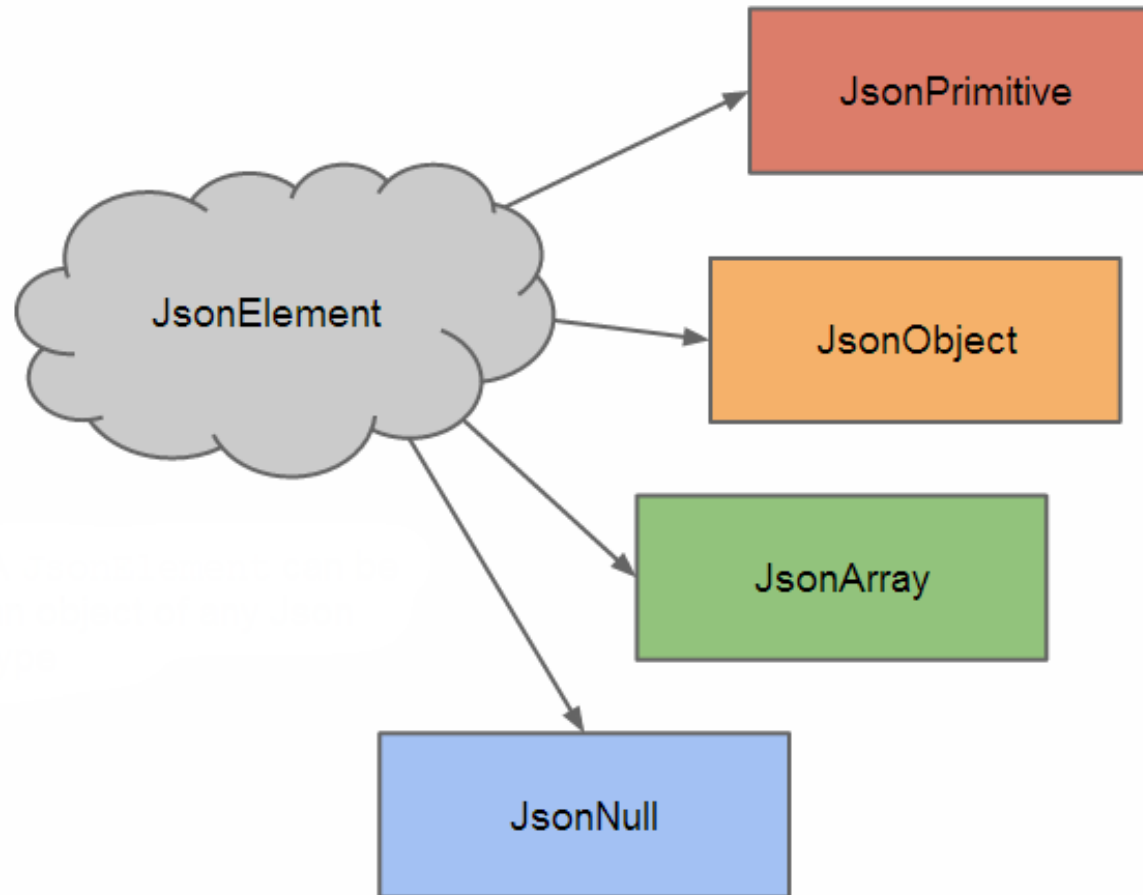
Name/value juftligi – qo'shtirnoq ichidagi o'zgaruvchi nomi, ":" va valuedan tashkil topadi.

```
"name" : "John"
```

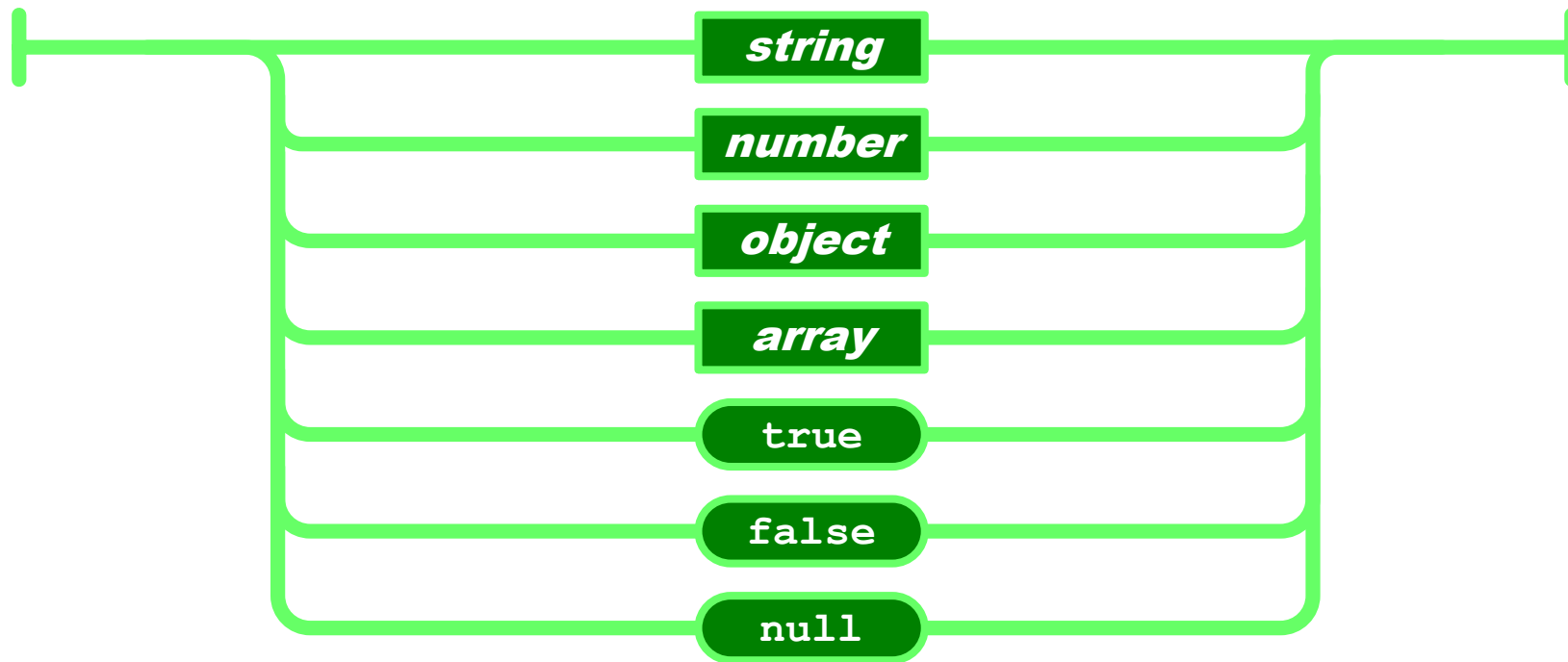
The JSON format is almost identical to JavaScript objects.

```
{ "name" : "John" }
```

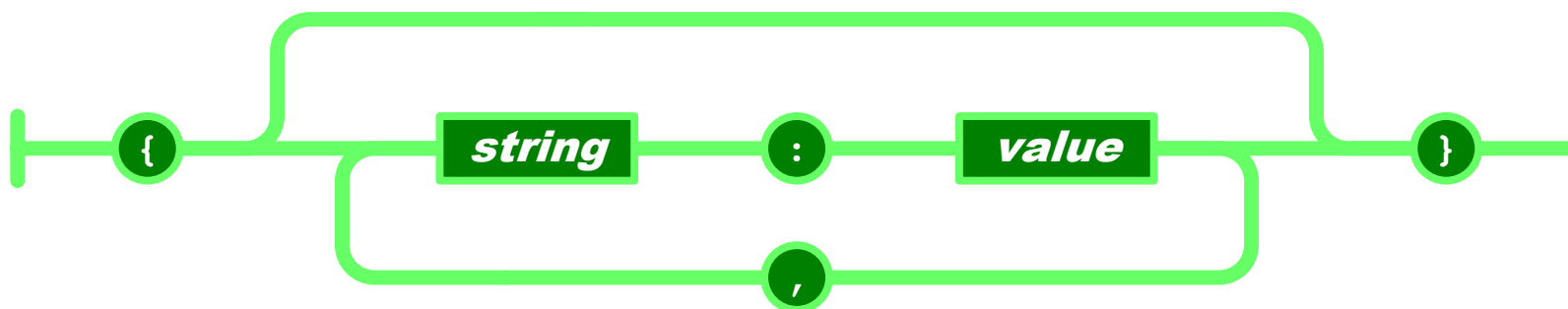
# Json Element



# Values

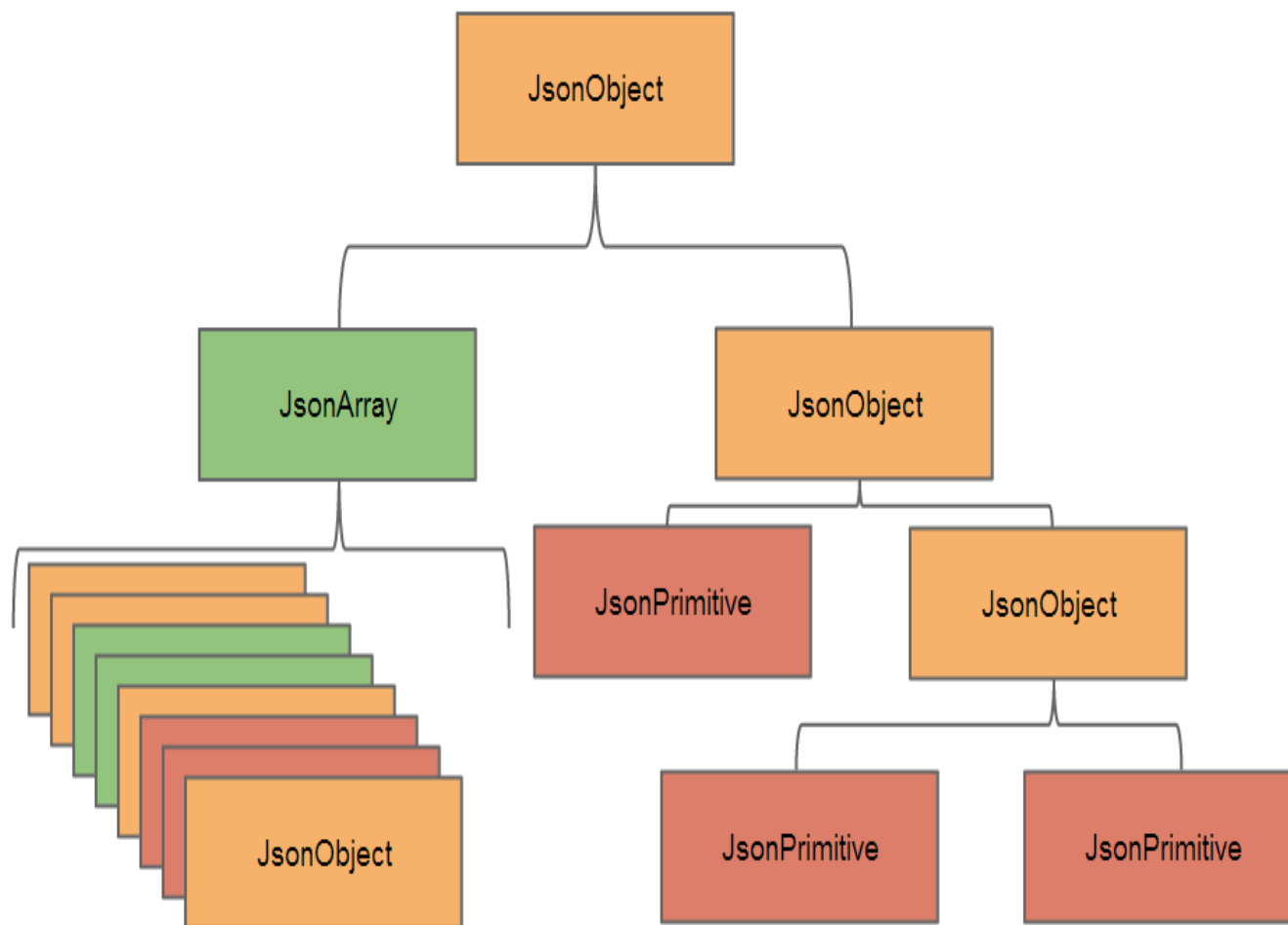


# Json Object





# Json Object

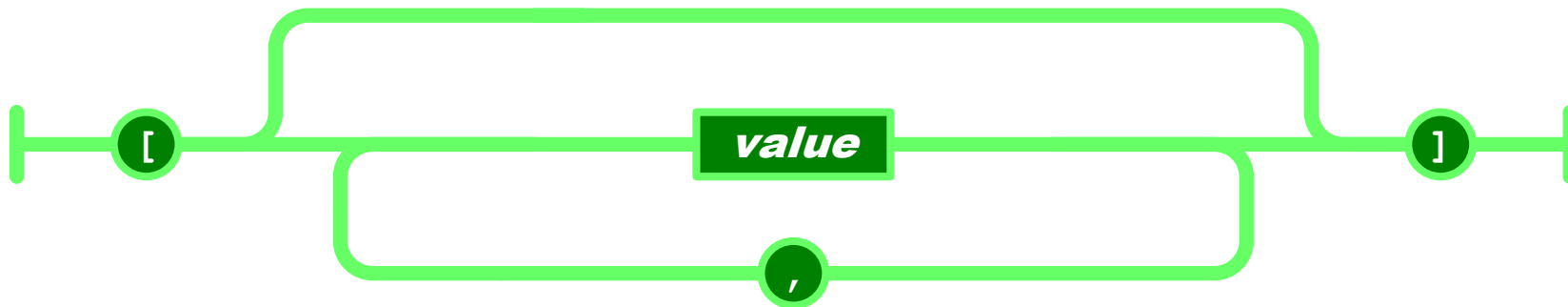


# Json Object

```
{"name": "Jack B. Nimble", "at large": true, "grade": "A", "level": 3,  
"format": {"type": "rect", "width": 1920, "height": 1080, "interlace": false,  
"framerate": 24}}
```

```
{  
  "name":      "Jack B. Nimble",  
  "at large":  true,  
  "grade":     "A",  
  "format": {  
    "type":      "rect",  
    "width":     1920,  
    "height":    1080,  
    "interlace": false,  
    "framerate": 24  
  }  
}
```

# Array



# Array

```
["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"]
```

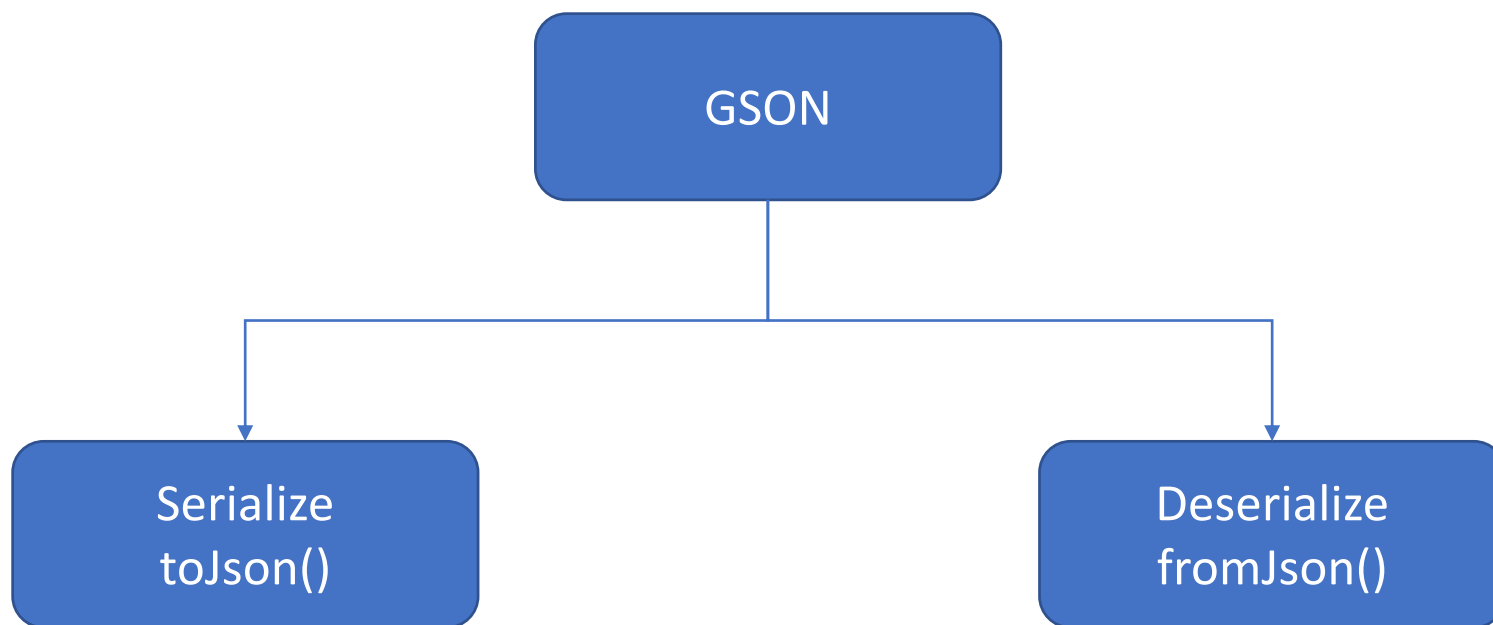
```
[  
  • [0, -1, 0],  
  • [1, 0],  
  • [0, 0, 1]  
]
```

# Mapping between JSON and Java entities

JSON	Java
string	java.lang.String
number	java.lang.Number
true   false	java.lang.Boolean
null	null
array	java.util.List
object	java.util.Map

# GSON

```
<!-- https://mvnrepository.com/artifact/com.google.code.gson/gson -->  
<dependency>  
  <groupId>com.google.code.gson</groupId>  
  <artifactId>gson</artifactId>  
  <version>2.8.6</version>  
</dependency>
```





# Gson objectini yaratish

- `Gson gson=new Gson();`
- `GsonBuilder gsonBuilder=new GsonBuilder();`  
`Gson gson=gsonBuilder.create();`

# GsonBuilder

- `setPrettyPrinting()`
- `excludeFieldsWithoutExposeAnnotation()`
- `setDateFormat("dd/MM/yyyy")`

# Project Lombok

- Lombok bu –dastur kodini kompilyasiya bo'lishdan oldin o'zgartirish orqali Java ga qo'shimcha imkoniyat qo'shuvchi proektdir;
- Project Lombok bu- getter, setter, equals, hashCode, toString hamda boshqa methodlarni avtomatik yaratib beruvchi va resurslarni avtomatik boshqarishga mo'ljallangan kompilyator pluginidir.

# Sozlash

1. Pluginni o'rnatish
2. Kutubxonani qo'shish

```
<!-- https://mvnrepository.com/artifact/org.projectlombok/lombok -->  
<dependency>  
  <groupId>org.projectlombok</groupId>  
  <artifactId>lombok</artifactId>  
  <version>1.18.12</version>  
  <scope>provided</scope>  
</dependency>
```

# Annotations

@NoArgsConstructor

@AllArgsConstructor

@Getter

@Setter

@ToString

@Data

**E'TIBORINGIZ UCHUN RAXMAT**