

[WWW.BALTICTALENTS.LT](http://WWW.BALTICTALENTS.LT)

---

**JAVA DOC, MAVEN**

# TESTAS #26 - JAVAC, JAR

# JAVADOC

- ▶ javadoc - tai įrankis ateinantis kartu su JDK (Java SE Development Kit) ir skirtas sugeneruoti programos dokumentaciją HTML formate
- ▶ javadoc komandos apsiraso java failuose kartu su programos kodu
- ▶ java kompiliatoriaus atžvilgiu, visos javadoc komandos atrodo kaip paprasti komentarai:

```
/**  
 * Tai programa, kuri kažką skaičiuoja  
 *  
 * @author Jonas Petras  
 * @version 1.0  
 */
```

# JAVADOC

- ▶ Dažniausiai javadoc naudojamos komandos:
  - ▶ **@author** *name-text*
  - ▶ **@version** *version-text*
  - ▶ **@param** *parameter-name description*
  - ▶ **@return** *description*
  - ▶ **@throws** *class-name description* (tas pats kas **@exception**)
  - ▶ **@see** *reference*
  - ▶ **{@link** *package.class#member label* }

# JAVADOC

- ▶ Pati java JDK dokumentacija parašyta naudojant javadoc, pvz:
- ▶ <https://docs.oracle.com/javase/9/docs/api/index.html?overview-summary.html>

# JAVADOC

```
cd demo
```

```
javadoc -d doc src/lt/baltictalents/p26/demo1/Demo1.java
```

čia **-d doc** nurodo, kad sugeneruotą dokumentaciją įrašyti į **doc** katalogą

## NUORODOS

- ▶ <https://en.wikipedia.org/wiki/Javadoc>
- ▶ <http://www.oracle.com/technetwork/articles/java/index-137868.html>
- ▶ [https://www.tutorialspoint.com/java/java\\_documentation.htm](https://www.tutorialspoint.com/java/java_documentation.htm)

# APACHE MAVEN

- ▶ Tai projekto (programinio) valdymo sistema, skirta:
  1. Automatizuotai užtikrinti projekto priklausomybes nuo trečių šalių bibliotekų
  2. Automatizuoti testavimą
  3. Automatizuoti diegimo procedūras
- ▶ <https://maven.apache.org/index.html>
- ▶ Diegimo instrukcija: <https://maven.apache.org/install.html>



# MAVEN

- ▶ Įsidiegiam **maven**
- ▶ Intellij: stovim ant projekto - “**Add Framework Support**” - “**Maven**”
- ▶ Redaguojam atsiradusį **pom.xml** failą nustatydami kokios java kompiliatoriaus **source** ir **target** versijos
  
- ▶ **Pastabos:**
  - ▶ Projekto pavertimo į **maven** projektą metu, visi java failai iš katalogo **src/lt/...** perkeliama į katalogą **src/main/java/lt/...**, todėl atitinkamai reiktų pasitaisyti programose naudojamų failų kelius iš “**src/lt/**” į “**src/main/java/lt/**”
  - ▶ Taip pat į **src/test/java** katalogą reikia perkelti testavimo paketus ir klases

## MAVEN KONFIGURAVIMAS - POM.XML

```
<build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <version>3.7.0</version>
      <configuration>
        <source>1.8</source>
        <target>1.8</target>
      </configuration>
    </plugin>
  </plugins>
</build>
```

## MAVEN JAR + MANIFEST

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-jar-plugin</artifactId>
  <version>3.0.2</version>
  <configuration>
    <archive>
      <manifest>
        <mainClass>lt.baltictalents.p25.demo1.Demo1</mainClass>
      </manifest>
    </archive>
  </configuration>
</plugin>
```

- ▶ <https://maven.apache.org/plugins/maven-jar-plugin/>
- ▶ <https://maven.apache.org/shared/maven-archiver/index.html>

## MAVEN - TREČIŲ ŠALIŲ BIBLIOTEKOS (3RD PARTY LIBRARIES)

- ▶ Norint įsidiesti kažkokią biblioteką, reikia susirasti aprašą skirtą įdiegimui **maven** projekte
- ▶ Tarkime norime įsidiesti **gson**. **Maven** saugykloje (<https://mvnrepository.com/>) randame, kad **junit** įdiegimui reikia įdėti į **pom.xml** failo elementą `<dependencies>...</dependencies>` (sukurti tokį jei faile jo nėra):

```
<dependency>  
  <groupId>com.google.code.gson</groupId>  
  <artifactId>gson</artifactId>  
  <version>2.8.2</version>  
</dependency>
```

# MAVEN JAR + BIBLIOTEKOS

```
<plugin>
  <artifactId>maven-assembly-plugin</artifactId>
  <configuration>
    <archive>
      <manifest>
        <mainClass>It.baltictalents.p26.demo2.Demo2</mainClass>
      </manifest>
    </archive>
    <descriptorRefs>
      <descriptorRef>jar-with-dependencies</descriptorRef>
    </descriptorRefs>
  </configuration>
</plugin>
```

- ▶ <http://maven.apache.org/plugins/maven-assembly-plugin/>
- ▶ `mvn clean compile assembly:single`

## PRATIMAS

- ▶ Įsidięgti **maven**
- ▶ Pasirašykite programą kuri kažką išveda json formatu naudojant **gson** biblioteką
- ▶ Paruoškite jar failą ir pabandykit jį vykdyti

## UŽDAVINYS

Panaudokite ICU4J biblioteką ir pabandykite parašyti programą kurią paleidžiant galima kaip parametrą reikia nurodyti kalbą ir skaičių ar kelis skaičius ir išvestų juos kaip tekstus. pvz, jei:

```
java -jar programa.jar lt 10 155
```

tai išvedame:

```
dešimt
```

```
šimtas penkiasdešim penki
```