



## Apache Maven

[Apache Maven](#) je alat namenjen izgradnji i upravljanju Java projektima. Primaran cilj Maven alata je veća razumljivost stanja i brži rad sa Java projekata. Kako bi ispunio ovaj cilj, bavi se:

- Olakšavanjem procesa izgradnje projekta, povlačenjem svih potrebnih biblioteka, skripti i fajlova koje projekat zahteva, skrivanjem kompleksnih detalja od developera.
- Pružanjem jedinstvenog sistema izgradnje koji se koristi istovetno za svaki od projekata na kojima se radi.
- Pružanjem lako preglednih informacija o samom projektu, od tehničkih potreba, lakog pokretanja testova pa do kontakt informacija autora.
- Podsticanjem dobrih praksi pri radu na projektima, odvajanjem nadležnosti projekta od alata upravljanja, pružajući određene konvencije koje se moraju ispoštovati (*opinionated software*) poput konvencija o lokacijama fajlova, njihovom imenovanju, ...

Zasnovan je na konceptu *POM (Project Object Model)* fajlova, najčešće zapisanih putem XML-a. Ovi fajlovi sadrže informaciju o projektu i konfiguracionim detaljima, poput neophodnih biblioteka, verzije projekta, načina pokretanja i izgradnje. POM fajlove Maven koristi da pruži podršku standardnom životnom ciklusu Java projekta - kompajliranje, pokretanje testova, kreiranje JAR fajla, ... Ostale, nestandardne funkcionalnosti se mogu realizovati u vidu *plugin*-a, čime je korišćenje Mavena dodatno obogaćeno.

## Instalacija

### Windows

1. Skinuti binarnu arhivu sa sajta: <https://maven.apache.org/download.cgi>.
2. Otpakovati je unutar željene lokacije.
3. Dodati putanju ka programu unutar PATH sistemskih ili korisničkih promenljivih. (<https://www.java.com/en/download/help/path.html>)
4. Po potrebi, ako prethodno nije nameštena - dodati JAVA\_HOME putanju u sistemske ili korisničke promenljive operativnog sistema. (Moguće je proveriti pozivanjem komande `echo %JAVA_HOME%` iz terminala koja će vratiti putanju ka instalaciji Jave).

### Linux / macOS

1. Iskoristiti sistemski *package manager* za konkretnu distribuciju, na primer Aptitude za Ubuntu ili Debian, DNF za Fedoru, Pacman za Arch ili Homebrew za macOS.

U slučaju da se Maven ne nalazi u ponuđenim paketima:

1. Skinuti binarnu arhivu sa sajta: <https://maven.apache.org/download.cgi>
2. Otpakovati je unutar željene lokacije.
3. Dodati putanju (označenu sa \$MAVEN\_PUTANJA\$) u varijable operativnog sistema sa export `PATH=$MAVEN_PUTANJA$/bin:$PATH` u datoteku `"/etc/profiles"`.
4. Po potrebi, ako prethodno nije nameštena - dodati JAVA\_HOME putanju u varijable operativnog sistema. (Moguće je proveriti pozivanjem komande `echo %JAVA_HOME%` iz terminala koja će vratiti putanju ka instalaciji Jave).

# Pokretanje Maven-a

Nakon što je instalacija Maven-a završena, moguće je proveriti da li je pravilno izvršena otvaranjem nove sesije u terminalu i pozivom “mvn --version” komande. Ispis ove komande je verzija Maven-a, lokacija odakle se pokreće, Java verzija koja se koristi kao i arhitektura i tip operativnog sistema.

## Kreiranje projekta

Pozicionirati se u direktorijumu unutar koga je potrebno kreirati projekat. Kreiranje prvog projekta može se odraditi komandom: `mvn archetype:generate -DgroupId=rs.ac.uns.ftn.osavezbe01 -DartifactId=Vezbe01 -DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4 -DinteractiveMode=false`

Nakon prvog pokretanja, proces će potrajati neko vreme jer Maven u pozadini skida potrebne biblioteke i smešta ih na disk. Po završetku procesa, kreiraće se novi projekat sa sledećom strukturom:

```
Vezbe01
|-- pom.xml
|-- src
|   |-- main
|       |-- java
|           |-- rs
|               |-- ac
|                   |-- uns
|                       |-- ftn
|                           |-- osavezbe01
|                               |-- App.java
|-- test
    |-- java
        |-- rs
            |-- ac
                |-- uns
                    |-- ftn
                        |-- osavezbe01
                            |-- AppTest.java
```

`src/main` sadrži izvorni kod projekta a `src/test` direktorijum testove. Fajl `pom.xml` je konfiguracioni fajl Mavena za ovaj konkretan projekat i prikazuje Project Object Model. Fajl je podeljen na odeljke:

- `modelVersion` - verzija POM modela
- `groupId` - oznaka organizacije kojoj projekat pripada, najčešće reverzno ime domena organizacije
- `artifactId` - naziv projekta
- `version` - trenutna verzija izvornog koda projekta, koja se zajedno sa `groupId` i `artifactId` koristi kao identifikator
- `packaging` - izlazni format pri izgradnji projekta
- `dependencies` - projekti neophodni za izgradnju ovog projekta, najčešće biblioteke drugih izvora
- `parent` - omogućuje nasleđivanje POM fajla, definišući neki drugi fajl kao roditeljski projekat
- `properties` - dinamičke promenljive koje se definišu kao “<naziv>vrednost</naziv>”, a u fajlu referenciraju kao {naziv}
- `modules` - spisak modula koje projekat sadrži, ovo se definiše ako je projekat parent projekat

Pri pozivanju komande za kreiranje projekta, ostavljen je njen cilj (*goal*), komandom `archetype:generate` čiji oslonac je bio na već predefinisanim tipu projekata sa unapred poznatim potrebnim bibliotekama i konfiguracionim fajlovima. Ciljevi se sintaksno zapisuju kao `plugin:goal`.

Unutar Maven-a, ovako predefinisani tipovi projekata se kreiraju pluginovima poznatim kao arhetipovi (*archetype*). Oni pružaju template na osnovu koga će projekat biti definisan i kreiran i na osnovu kojeg će se on dalje razvijati. Ovo je od velikog značaja pri radu u timu, na više projekata čime se pojednostavljuje proces prelaska sa jednog projekta na drugi jer su sami projekti međusobno konzistentniji. Neki od standardnih arhetipova projekata su:

- Maven-archetype-archetype
- Maven-archetype-j2ee-simple
- Maven-archetype-plugin
- Maven-archetype-quickstart
- Maven-archetype-site
- Maven-archetype-webapp

Punu listu je moguće pronaći na sajtu <https://maven.apache.org/archetypes/index.html>. Svaki od template-ova pruža olakšice pri kreiranju određenog tipa aplikacije i zgodno ih je koristiti kada se zna tačna svrha korišćenja aplikacije.

## Izgradnja projekta

Izgradnja (build) projekta se može pokrenuti komandom: `mvn package`

Pokretanjem ove komande, u direktorijumu `target`, kreiraće se arhivirana, JAR verzija aplikacije, spremna za upotrebu. Za razliku od prethodnog puta je Maven alatu prosleđen cilj, sada mu se prosleđuje faza životnog ciklusa projekta koji treba da se izvrši (`package`). Faza je samo jedan od procesa u životnom ciklusu, koji je definisan kao uređeni skup faza izvršavanja. Svaka faza izvršava jedan ili više ciljeva vezanih za nju, pri čemu su ciljevi konkretne operacije poput kompajliranja, izvršavanja testova, pakovanja, ...

Sam Maven prepoznaje tri ugrađena životna ciklusa: `clean`, `default`, `site`.

[Clean životni ciklus](#) se bavi uklanjanjem fajlova i foldera koji Maven generiše prilikom upravljanja projekta i za cilj ima vraćanje radne sredine na staro, pre izmena. Sastoji se od faza:

- Pre-clean
- Clean
- Post-clean

[Default životni ciklus](#) bavi se validacijom, kompilacijom, testiranjem, izgradnjom, instalacijom i postavkom projekata. Sastoji se od faza:

- Validate
- Initialize
- Generate-sources
- Process-sources
- Compile
- Process-classes
- Generate-Test-Sources
- Process-Test-Resources
- Test-Compile
- Process-Test-Classes
- Test
- Prepare-Package
- Package
- Pre-Integration-Test
- Integration-Test
- Post-Integration-Test
- Verify
- Install
- Deploy

[Site životni ciklus](#) se bavi generisanjem projektne dokumentacije i sastoji se od faza:

- Pre-Site
- Site
- Post-Site
- Site-Deploy

Pozivom jedne faze, izvršiće se i sve pre nje. Na primer, pre verifikacije projekta, proveriće se njegova validnost sintakse i potrebnih konfiguracionih parametara, on će se inicijalizovati, kompajlirati, izvršiće se navedeni testovi i projekat izgraditi i zapakovati u arhivu koju je moguće pokrenuti i verifikovati. Za svaku od faza, izvršiće se neki cilj.

## Pokretanje projekta

Pokretanje projekta je moguće izvršiti komandom:

```
java -cp target/Vezbe01-1.0-SNAPSHOT.jar rs.ac.uns.ftn.osavezbe01.App
```

Ovo će pokrenuti JAR fajl sa datom ulaznom putanjom i na terminalu ispisati "Hello World!".

## Generisanje dokumentacije

Generisanje dokumentacije je moguće odraditi pokretanjem site faze iz istoimenog životnog ciklusa: `mvn site`

Ovo će generisati HTML dokumentaciju projekta u `target/site` direktorijumu.