

## Domaći zadatak - Maven (rok je 3.6.2017. za sve grupe)

**NAPOMENA:** Ideja ovog domaćeg zadatka je da uspešno pretvorite jedan Eclipse (Ant) Java projekat u Maven Java projekat i to ručnom transformacijom korak po korak. Kao polazna tačka korišćiće se projekat biblioteka2017 koji je rađen na vežbama i dograđivan u toku dodatne nastave. Kao uzor kako izgleda Maven projekat se može koristiti primer sa časa koji se može naći ovde:

<https://github.com/bojantomic/mavenbiblioteka2017>

Skinuti u Eclipse ceo kod projekta biblioteka2017 iz GitHub repozitorijuma

<https://github.com/bojantomic/biblioteka2017.git>

Nakon toga napraviti u Eclipse okruženju još i nov Maven projekat sa sledećim koordinatama:

group-id: rs.ac.bg.fon.jgrass

artifact-id: mvnbiblioteka2017

version: 1.0

Ako je Eclipse dodao neke probne klase (obične ili test) u ovaj Maven projekat, potrebno ih je obrisati.

Prekopirati sve obične klase (dakle, NE test klase) iz biblioteka2017 projekta u mvnbiblioteka2017 projekat u odgovarajući source folder Maven projekata.

Pri tome, potrebno je klase prekopirati i u odgovarajuće pakete u okviru tog source foldera. Ako je, na primer, klasa u starom projektu bila u paketu sistemske\_operacije, onda je novi paket u Maven projektu rs.ac.bg.fon.jgrass.mvnbiblioteka2017.sistemske\_operacije. Naravno, potrebno je promeniti i naziv paketa u deklaraciji u .java fajlovima da bi se klasa kompajlirala kako treba.

Prekopirati sve ikonice (zapravo samo je jedna) iz biblioteka2017 projekta u mvnbiblioteka2017 projekat u odgovarajući source folder Maven projekata (ako taj source folder ne postoji u Maven projektu, napraviti ga).

Uraditi prvi build projekta (mvn install) i proveriti da se sve kompajliralo kako treba i da je napravljen i jar fajl sa odg. nazivom i sadržajem (class fajlovi i ikonica).

Napraviti lokalni repozitorijum kojim se verzionira ovaj Maven projekat, kao i prazan repozitorijum na GitHub-u koji se zove „mvnbiblioteka2017“.

Postaviti da se ignorišu fajlovi .classpath i .project kao i .settings folder iz Eclipse projekta. Ako nisu vidljivi ovi fajlovi, namestiti prvo filter u project explorer-u u Eclipse-u.

Uraditi commit sa jasno naznačenom porukom o tome šta je urađeno do sada.

Prekopirati sve TEST klase iz biblioteka2017 projekta u mvnbiblioteka2017 projekat u odgovarajući source folder Maven projekata. Pri tome, potrebno je klase prekopirati i u odgovarajuće pakete u okviru tog source foldera.

Namestiti da projekat ima zavisnost (Maven dependency) ka odgovarajućoj verziji JUnit biblioteke da bi testovi mogli da se kompajliraju i izvrše, npr. JUnit 4.11.

Ponovo pokrenuti mvn install i proveriti da se svi testovi izvršavaju kako treba, kao i da u JAR fajl NISU spakovane test klase.

Uraditi commit sa jasno naznačenom porukom o tome šta je urađeno do sada.

Dodati u projekat i biblioteke (tj. odgovarajuće Maven dependency-je) GSON 2.8.0 kao i itext 2.1.7 i sačuvati sve promene.

Uraditi poslednji commit, sa jasno naznačenom porukom o tome šta je urađeno do sada.

Po završetku domaćeg zadatka, uraditi konačni push na GitHub repozitorijum (sa svim commit-ovima) i poslati email [sa linkom na GitHub repozitorijum](#) Bojanu Tomiću ([tomicb@fon.bg.ac.rs](mailto:tomicb@fon.bg.ac.rs)). Rok za izradu ovog domaćeg zadatka je dve nedelje tj. 3.6.2017. za sve grupe.