

# Tutorijal 5

## Zadatak 1

U ovom zadatku potrebno je implementirati "Builder pattern". Kao primjer možete koristiti neke simbolične klase koje bi mogle da se efektivno iskoriste u ovom pattern-u. Ovaj zadatak treba da ima minimalno jednu model klasu i builder klasu koja će se vezati za model klasu. Builder klasa treba da ima sve attribute kao i model klasa uz dodatno definirane setter-e za sve te attribute. Svaki od tih setter-a vraća referencu na builder klasu.

## Zadatak 2

U ovom zadatku potrebno je implementirati "Factory pattern". Ovaj pattern se zasniva na "factory" klasi koja će nuditi korisniku da na osnovu nekih specifikacija vrati neki objekat iz "fabrike". Dakle, korisnik neće direktno instancirati objekat neke željene klase, nego će zatražiti od "factory" klase da vrati instance tih željenih objekata, ali na osnovu neke specifikacije. Zamislite to kao da neki kupac traži "open source" operativni sistem (što bi bila specifikacija), a "factory" klasa na osnovu toga vrati instancu "Andorid" klase.

## Zadatak 3

U ovom zadatku potrebno je implementirati "Singleton" pattern. Najbolji način za Implementaciju ovog pattern-a bi bila simulacija konekcije na bazu podataka. Tačnije, implementirajte klasu koja će predstavljati konekciju na bazu podataka, a samim tim, Singleton ćete uraditi na način da vraćate svaki put istu instancu baze podataka, te ta klasa koju ste implementirali kao konekciju na bazu ne smije se instancirati zasebno, nego samo preko odgovarajuće statičke metode (nikakvo znanje o bazama vam nije potrebno za rješenje ovog zadatka jer ovo je samo najbitniji primjer primjene Singleton pattern-a).