

## KREACIJSKI PATERNI

### Singleton

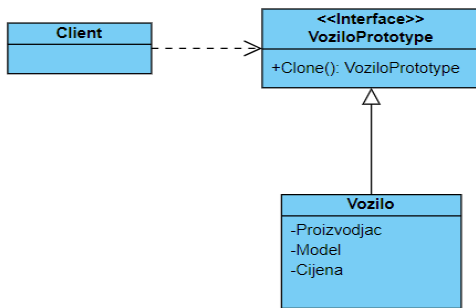
Singleton patern osigurava da se klasa može instancirati samo jednom i da osigura globalni pristup kreiranoj instanci klase

Singleton patern možemo implementirati za komuniciranje sa vanjskim servisima, gdje bi se jedna singleton klasa brinula o integraciji servisa. U slučaju da budemo koristili neki vanjski servis koji je povezan sa datumom iznajmljivanja vozila tj. kalendarima (Calendy) mogli bismo implementirati singleton klasu koja će predstavljati kalendar događaja

### Prototype

Ovaj patern služi za kreiranje novih objekata klonirajući već postojeći objekat. Tako se dozvoljava kreiranje prilagođenih objekata bez poznavanja klase ili detalja o kreiranju objekta. Koristi se za sakrivanje konkretne klase od klijenta, editovanje klase za vrijeme izvršavanja, te da broj klasa u sistemu bude u minimumu.

Prototype patern ćemo iskoristiti nad klasom Vozilo. Razlog za ovo je što je u mnogim metodama potrebno praviti kopije vozila dobijenog iz baze radi daljnje obrade ali ne želimo promijeniti originalni. Zbog toga se to olakša kreirajući Prototype patern za Vozilo te ga samo koristimo gdje je potrebno.



### Factory Method

Uloga Factory Method paterna je da omogući kreiranje objekata na način da podklase odluče koju klasu instancirati. Različite podklase mogu na različit način implementirati interfejs. Odgovarajuća podklasa se instancira preko posebne metode na osnovu informacija od strane klijenta.

Primjer primjene:

S tim da smo se odlucili da imamo vise klasa za razlicite tipove vozila, mogli bi smo implementirati factory class koji će instancirati objekte vozila na temelju korisničkog zahtjeva. Na taj način bi izbjegli da svugdje u kodu moramo eksplicitno instancirati konkretne klase vozila.

## Abstract Factory

Abstract Factory patern omogućava da se kreiraju familije povezanih objekata bez specificiranja konkretnih klasa

Patern je iskorišten kod kod klase Account, jer sve nasljeđene klase iz Account klase će moći koristiti metode te klase.

Dodatno se može dodati abstract factory za lokalizaciju web aplikacija koja podržava više jezika. Možemo imati Abstract Factory LokalizacijaFactory koja definira apstraktne metode poput stvoriOpis(),stvoriDugme() I zatim kreirati konkretne factories kao što je npr. EngleskiLokalizacijaFactory,NjemačkiLokalizacijaFactory, koja stvara objekte na datim jezicima.

## Builder patern

Uloga Builder paternu je odvajanje specifikacije kompleksnih objekata od njihove stvarne konstrukcije

Ovaj patern ćemo iskoristiti za filtriranje vozila tokom pretrage. Na klasu Filter dodani su atributi za sortiranje u odabiru minimalne I maksimalne cijene. S obzirom da ne moraju svi atributi biti iskorišteni kod filtriranja, na ovaj način izbjegavamo inicijalizaciju svih polja.

