

Objektno orijentisani JavaScript – osnovi koje svaki frontend developer mora znati

Uvod

JavaScript je skripting jezik koji je implementovao nekoliko standardnih programskih paradigmi. Svi se složili da je to proceduralan i funkcionalan jezik, ali mnogi osporavaju da je to i objektno-orijentisani jezik. Iako je gotovo sve u JavaScriptu objekat, programeri koji dolaze iz Java ili C# se jednostavno ne slažu sa tim da JavaScript obezbeuje iste OO karakteristike kao i pomenuti jezici. U ovom tekstu pokušati da objasnim osnove OO principa koji važe u JavaScriptu: definisanje objekata i nasleđivanje. Takođe, spomenuto sakrivanje određenih elemenata u objektu, ali ćemo se ovim, kao i ostalim objektno-orijentisanim stvarima baviti detaljnije u narednim postovima. Pisao sam pretpostavkom da ste do sada programirali u ovom jeziku, pa neke osnovnije stvari neće biti detaljno objašnjene.

JavaScript ne podržava klase

Objektno-orijentisani jezici podržavaju definisanje klase, od kojih, nakon instanciranja, nastaju objekti. To su takozvani class-based jezici, kod kojih postoji jasna razlika između klase i objekata. U JavaScriptu, ta razlika nije tako uoljiva – jednostavno, postoje samo objekti. Da budem malo konkretniji: u C# na primer, pomou ključne reči `class` definišete klasu, a onda negde u programu kreirate objekat te klase. U JavaScriptu, objekti se definišu pomou obinih funkcija koje takođe možete u bilo kom trenutku da instancirate na identičan način kao i u C#, pozivanjem odgovarajućeg konstruktora. Imajte na umu da JavaScript spada u takozvane prototype-based (class-less) jezike, pa bi bilo lepo da, paralelno sa ovim tekstom, pročitate i linkovani tekst.

Kreiranje objekata u JavaScriptu

JavaScript ima nekoliko ugrađenih objekata (Array, Boolean, Date, Function, Math, Number, RegExp i String), ali oni nisu predmet ovog posta. Ukoliko želite da pročitate više o tome, preporučujem da pogledate listu ugrađenih objekata u JavaScript. Za nas interesantno pitanje je: Kako ja, kao developer, mogu definisati svoje objekte?

Postoje dva načina za kreiranje objekata u JavaScriptu:

1. Takozvani *Object literal* (molim vas, ukoliko znate, pomozite mi oko prevoda ovog termina u komentar)
2. kreiranjem objekata pomou *konstruktor* funkcije.

U praksi ćete se susretati sa oba načina kreiranja objekta, što vam govori da svaki od njih ima svoje ari. Pa, ponimo sa detaljima.

