Constructor Function

Pada bagian-bagian sebelumnya, kita membuat objeknya secara langsung. Bagaimana kalau kita ingin membuat objek yang lain dengan property dan method yang sama? Kita bisa saja buat objek secara langsung dua seperti ini:

var orangSatu = {

namaDepan: 'Fajar',

namaBelakang: 'Sobirin',

namaLengkap: function() {

console.log(this.namaDepan + ' ' + this.namaBelakang);

}

};

var orangDua = {

namaDepan: 'Saleh',

namaBelakang: 'Solahudin',

namaLengkap: function() {

console.log(this.namaDepan + ' ' + this.namaBelakang);

}

};

Kode di atas tidak akan efektif jika kita ingin membuat banyak objek dengan property dan method yang sama. Karena kita harus menulis ulang kode yang sama berkali-kali. Lalu bagaimana solusinya? kita bisa menggunakan constructor function. Berikut contoh penggunaan constructor function:

function Orang(namaDepan, namaBelakang) {

// property

this.namaDepan = namaDepan;

this.namaBelakang = namaBelakang;

// method

this.namaLengkap = function() {

console.log(this.namaDepan + ' ' + this.namaBelakang);

}

};

var orangSatu = new Orang("Fajar", "Sobirin");

var orangDua = new Orang("Saleh", "Solahudin");

orangSatu.namaLengkap();

orangDua.namaLengkap();

Pada kode di atas kita membuat objek baru dengan kata kunci new, lalu diberikan nilai parameter namaDepan dan namaBelakang, jadi berapapun objek yang ingin kita buat cukup gunakan kata kunci new saja. Pada kode diatas kita juga membuat nama fungsi yang berawalan huruf besar Orang (ini berguna untuk menandakan bahwa fungsi itu adalah constructor function).

Perhatikan kode di atas yang telah ditandai (baris 1 sampai 10), bagian yang ditandai itu berfungsi untuk mengenerate objek seperti layaknya contoh pertama.