



DISKUSIKAN MATERI

LAPORKAN MATERI

Daftar Modul

Masukkan kata kunci

Persetujuan Hak Cipta

Modul 1: Introduction Course (Gratis)

Prasyarat Kelas (Gratis)

Apa yang Akan Kita Pelajari? (Gratis)

Tools Requirement

Modul 2: ECMAScript 6 (Gratis)

Running Snippet Code (Gratis)

Persiapan Project Latihan (Gratis)

Instance

Setelah kita membuat class pada JavaScript, lantas bagaimana cara membuat instance dari class tersebut? Tapi sebelumnya, apa itu *instance*? Instance merupakan objek yang memiliki properti dan method yang telah ditentukan oleh *blueprint*-nya (class), atau singkatnya adalah objek yang merupakan hasil realisasi dari sebuah *blueprint*.

Sama seperti *constructor function*, untuk membuat *instance* dari class pada ES6 kita gunakan keyword **new**.

```
1. class Car {
2.   constructor(manufacture, color) {
3.     this.manufacture = manufacture;
4.     this.color = color;
5.     this.enginesActive = false;
6.   }
7. }
8.
9. const johnCar = new Car("Honda", "Red");
```

Pembuatan class menggunakan ES6 lebih ketat dibandingkan dengan constructor function, di mana dalam pembuatan *instance* wajib menggunakan keyword **new**. Jika kita tidak menuliskannya, maka akan terjadi error seperti ini:

```
1. class Car {
2.   constructor(manufacture, color) {
3.     this.manufacture = manufacture;
4.     this.color = color;
5.     this.enginesActive = false;
6.   }
7. }
8.
9. const johnCar = Car("Honda", "Red");
10.
11. /* error:
12. TypeError: Class constructor Car cannot be invoked without 'new'
13. */
```

Kita juga dapat membuat banyak instance dari class yang sama, dan tentunya objek yang kita buat memiliki karakteristik (properti dan method) yang sama. Walaupun sama, namun nilai dari propertinya bersifat unik atau bisa saja berbeda. Contohnya seperti ini:

```
1. class Car {
2.   constructor(manufacture, color) {
3.     this.manufacture = manufacture;
4.     this.color = color;
5.     this.enginesActive = false;
6.   }
7. }
8.
9. const johnCar = new Car("Honda", "Red");
10. const adamCar = new Car("Tesla", "Black");
11.
12. console.log(johnCar.manufacture);
13. console.log(adamCar.manufacture);
14.
15. /* output:
16. Honda
17. Tesla
18. */
```

Variabel **johnCar** dan **adamCar** merupakan sebuah objek dari **Car**. Tentu keduanya akan memiliki properti **manufacture**, **color**, dan **enginesActive**. Namun pada output kita melihat bahwa nilai dari properti kedua objek tersebut berbeda, karena kita dapat memberikan nilai yang berbeda pada saat objeknya dibuat.

[← KEMBALI KE MATERI SEBELUMNYA](#)[LANJUTKAN KE MATERI BERIKUTNYA →](#)

