

# Metode dd () dan dump ()

Laravel menyediakan dua metode preddine dd () dan dump () untuk debugging. Anda dapat memanggil metode ini pada contoh koleksi. Anda dapat membaca dokumen resmi untuk metode [dump \(\)](#) dan [dd \(\)](#) Dalam versi Laravel sebelumnya, ketika melakukan debugging koleksi, kami akan menetapkan variabel ke koleksi, tetapi tetap membuang variabel saat kami mengubah koleksi. Ini tidak akan terjadi lagi di Laravel 5.5 karena sekarang kita dapat memanggil `dd()` atau `dump()` secara langsung pada instance koleksi. Metode ini membuat proses debug menjadi lebih mudah dan sederhana.

Bayangkan Anda memiliki koleksi pengguna yang mengalami serangkaian transformasi dan kami ingin memeriksa koleksi di setiap langkah, maka ini akan dilakukan:

## Menggunakan dump ()

```
$ users = Users::all();
$ users->dump()
    -> sortBy('created_at')
    -> dump()
    -> take(10)
    -> pluck('name')
    -> dd() // here dd will dump the result and stops the execution.
    -> take(1);
```

## Output:



```
Collection {#237 ▼
  #items: array:50 [▶]
}

Collection {#225 ▼
  #items: array:50 [▶]
}

Collection {#210 ▼
  #items: array:10 [▼
    0 => "Howell Larson"
    1 => "Adah Brakus IV"
    2 => "Orin Hodkiewicz III"
    3 => "Annalise Conroy"
    4 => "Miss Jayda Lakin"
    5 => "Jacklyn Ziemann"
    6 => "Miss Jayda Brekke MD"
    7 => "Tess Langosh"
    8 => "Mr. Isidro Cartwright"
    9 => "Shaina Gutkowski"
  ]
}
```

## Perbedaan antara dd () dan dump ()

Kedua fungsi digunakan untuk debugging. Tetapi memiliki satu perbedaan. membuang () hasil output dan kemudian melanjutkan pemrosesan. dd () output hasil dan berhenti proses segera (dd singkatan dump dan mati).