**Jurus 2:**

**Kuasai Struktur Data**

#### **Pengertian Struktur Data**

**Dalam istilah ilmu komputer**, **struktur data adalah** cara penyimpanan , pengorganisasian , dan pengaturan data di dalam media penyimpanan komputer sehingga data tersebut dapat digunakan secara efisien.   
  
**Dalam teknik pemrograman**, struktur data berarti tata letak data yang berisi kolom-kolom data,baik itu kolom yang tampak oleh pengguna (user) ataupunkolom yang hanya digunakan untuk keperluan pemrograman yang tidak tampak oleh pengguna. Setiap baris dari kumpulan kolom-kolom tersebut dinamakan catatan (record). Lebar kolom untuk data dapat berubah dan bervariasi. Ada kolom yang lebarnya berubah secara dinamis sesuai masukan dari pengguna dan juga ada kolom yang lebarnya tetap.   
  
Dengan sifatnya ini, sebuah struktur data dapat diterapkan untuk pengolahan database, misalnya untuk keperluan data keuangan, atau untuk pengolah kata (word processor) yang kolomnya berubah secara dinamis. Contoh struktur data dapat dilihat  pada file-file spreadsheet, database, pengolahan kata, gambar yang dikompres, dan pemampatan file (kompres) dengan teknik tertentu yang memanfaatkan struktur data.

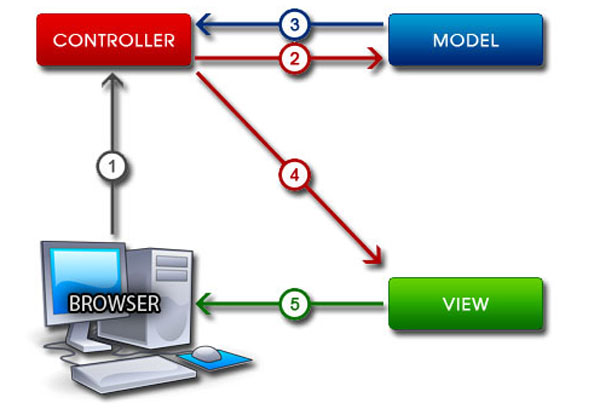
#### **Klasifikasi Data**

Pada garis besarnya, data dapat dikategorikan menjadi:  
  
A.**Tipe Data Sederhana**, yang terdiri dari:  
  
– Data Sederhana Tunggal, misalnya Integer, Real, Boolean, dan Karakter.  
– Data Sederhana Majemuk, misalnya String.  
  
B. **Struktur Data**  
  
– Struktur Data Sederhana, misalnya Array dan Record.  
– Struktur Data majemuk, terdiri dari:  
      o Linier, misalnya: Stack, Queue, dan Linier Linked List.  
      o Nonlinier, misalnya Binary Tree, Binary Search Tree, Graph, dll.  
  
Pemakaian struktur data yang tepat di dalam proses pemrograman akan menghasilkan algoritma yang lebih jelas dan tepat, sehingga menjadikan program secara keseluruhan lebih efisien dan sederhana.

**Jurus 3:**

**Kuasai Konsep Pemrograman MVC,**





Sebelum belajar lebih jauh mengenai laravel, kita mulai terlebih dahulu dari konsep dasar yang digunakan, yaitu konsep MVC (Model, View, Controller). Laravel merupakan salah satu framework yang memisahkan antara data (**Model**) dari tampilan (**View**) dan cara bagaimana memprosesnya (**Controller**).

Apa itu Model?  **Model** adalah sebagai penghubung antara Controller dengan database yang berguna untuk mengambil data. Sementara  **Controller**, dapat diartikan sebagai kendali atau pengendali diambil dari kata controll bisa juga diartikan jembatan antara view dan model. Controller memiliki tugas mengendalikan ataupun menjembatani antara apa yang di-*request*yang selanjutnya Controller akan memprosesnya. Pada Laravel file Controller berada di app/Http/Controllers. Adapun **View** digunakan untuk menerima dan merepresentasikan data kepada user. Letak file View berada pada folder Resources/views.

**Struktur laravel**

Berikut adalah beberapa struktur dari laravel :

* **App**Berisi kumpulan logika dan alur sistem yang akan dibuat.
* **Bootstrap**Direktori ini berisi beberapa file kerangka *framework* laravel termasuk *autoload*yang befungsi untuk mengoptimasi kinerja sistem yang dihasilkan.
* **Config**Mencakup seluruh konfigurasi *framework* mulai dari *database, app, mail*, dan lain sebagainya.
* **Database**Sebagai folder penampung file *migrations* dan *seeds* yang berhubungan langsung ke pengolahan data dalam database.
* **Public**Sebagai folder yang akan diakses oleh *public/users* nantinya. Folder ini juga berisi file-file assets (css/js/images/dll)
* **Resource**Folder ini berisi semua *resource* untuk bagian *frontend.*
* **Routes**Folder ini digunakan untuk menentukan format url yang digunakan untuk mengakses halaman yang dibuat
* **Storage**Berisi *compiled blade templates, session, cache, logs* dan file lainnya yang di-*generate*otomatis oleh *framework*.
* **Tests**Berisi semua *test* yang kita buat untuk aplikasi.
* **Vendor**Berisi seluruh library-library yang digunakan dalam *framework* laravel maupun yang diinstall melalui composer.

**Menguasai Route**