1. Large Dataset တွေကို memory နည်းနည်းဖြင့် သုံးရန် LazyCollection ကို အသုံးပြုပါသည်။

2. Eloquent model ကိုသုံးတဲ အခါ N+1 query problem ကိုမဖြစ်အောင် Eager Loading နည်းစနစ်ကို သုံးသင့်ပါသည်။ Eager Loading တွင် query အရေအတွက် အနည်းငယ်ဖြင့် လိုအပ်သော data များကို ဆွဲယူ အသုံးပြု နိုင်ခြင်း ကြောင့် N+1 query problem ဖြစ်ပေါ်ခြင်း မရှိနိုင်ပါ။

3. ORM သည် Object Relational Mapper ဟု ခေါ်ပြီး Laravel Framework တွင် default အနေဖြင့် ပါဝင်ကာ လူကြိုက်များသော feature တစ်ခုဖြစ်သည်။ Database ထဲရှိ data များအား handle လုပ်နိုင်ပြီး controller class များနှင့် ချိတ်ဆက်ကာ insert,create, update,edit, delete စသော လုပ်ငန်းများကို query များအသုံးပြု၍ data များအား ထိန်းချုပ်လုပ်ဆောင်ပါသည်။

4. Chunk function သည် large group of data set များအား smaller group of data set (သေးငယ်သော data အစုအဖွဲ့ ) အနေဖြင့် ခွဲထုတ်ခြင်းအလုပ်ကို လုပ်ဆောင်ပါသည်။ အကယ်၍ record ဆိုတဲ့ data group လိုက်ကြီးထဲမှ data များကို ၃ခု တစ်ခါစီ ဆွဲထုတ်ချင်တယ်ဆိုပါစို့ ။ အဆိုပါအခြေအနေမျိုးတွေမှာ chunk function အားအသုံးပြုနိုင်ပါသည်။

Example :

$records = collect(['orange', 'blue', 'green', 'red', 'yellow', 'purple']);

$chunks = $records->chunk(3);

  print\_r($chunks);

/\*

Output:

Array

(

     [0] => Array

         (

             [0] => orange

             [1] => blue

             [2] => green

         )

     [1] => Array

         (

             [3] => red

             [4] => yellow

             [5] => purple

         )

)

 5. $collection = collect([1 , 2, 3, 4, 5, 6]);

$lessthanthree = $collection->filter(function ($value, $key) {

return $value < 3;

});

$greaterthanthree = $collection->filter(function ($value, $key) {

return $value >= 3;

});

$lessthanthree->all();

$greaterthanthree->all();

6. PHP file တစ်ခုမှ အကြောင်းအရာများကို တခြား PHP file တစ်ခုသို့ ထည့်သွင်း သောအချိန်တွင် include or require function တစ်ခုခုကို အသုံးပြုနိုင်ပါသည်။ Function နှစ်ခုလုံးသည် PHP မှ built-in function များဖြစ်ပီး တူညီသော action များကို အလုပ်လုပ်ပေးပါသည်။ တကယ်တမ်းတွင် တူညီတယ် ထင်ရ သော်လည်း အလုပ်လုပ်ပုံကွာခြားပါသည်။ include သည် error တစ်ခုခုဖြစ် ရာတွင် warning သာပြပြီး php script များကို ဆက်လက်လုပ်ဆောင်ပေးပါသည်။

require သည် php script များကို ဆက်လက်လုပ်ဆောင်ခြင်း မရှိတော့ဘဲ ရပ်

တန့်လိုက်ပြီး error အနေဖြင့် ပြန်ပြပါသည်။

7. Facade pattern ကို အသုံးပြုခြင်းအားဖြင့် laravel project တစ်ခုမှာ ပါဝင်တဲ့ services တွေကို developer တွေအနေနဲ့ အသုံးပြုနိုင်လာပါမယ်။ ထို့အပြင် အာ့ services တွေမှာ ပါတဲ့ ရှုပ်ထွေးသော codes များသည်လည်း ပို၍ ဖတ်ရလွယ်စေ ပါမယ်။

8. Model::get()->count() ကို အသုံးပြုသင့်ပါသည်။

9. Eager Loading တွင် query အရေအတွက် အနည်းငယ်ဖြင့် လိုအပ်သော data များကို ဆွဲယူ အသုံးပြု နိုင်ပါသည်။ ထို့အပြင် query တစ်ကြောင်းတည်းဖြင့် Relationship ချိတ်ထားသော Table များ၏ data များကိုလည်း ခေါ်နိုင်ပါသည်။ သို့ဖြစ်ရာ database အပေါ် အကြိမ်များစွာ သက်ရောက်မှု မှ လျော့ချနိုင်ပါသည်။ ထို့ကြောင့် Eager Loading အသုံးပြုခြင်းသည် application (သို့) server တွေ၏ Performance အား သိသိသာသာ ကောင်းလာစေနိုင်ပါသည်။

10. Sample query using eager loading

‘Order::with(‘product’)->where(‘id’,1)->get();’

11. Hacking လုပ်တဲ့သူက User အတုတွေ အများကြီး ဖန်တီးပြီး server တစ်ခုကို down အောင် ပြုလုပ်တာမျိုးကို DDos attack ဟူ အဓိပ္ပါယ် ဖွင့်ဆိုချင်ပါသည်။ DDos attack ခံရလျှင် server က လေးသွားပြီး တကယ် အသုံးပြုမည့် User များ အသုံးပြု၍ မရနိုင်တော့ပါ။ ထို့ကြောင့် Google Captcha ဆိုတာမျိုး ခံထားရပါ တယ်။ Google Captcha သည် User များအား human or robots ဖြစ်မဖြစ် စစ်ဆေးပေးပြီး User account အတုများ ဖြစ်အောင် Hacker များမှ ပြုလုပ်ခြင်း အား ကာကွယ်ပေးနိုင်ပါသည်။

12. Web browser (သို့) Web application တွေရဲ့ Client Request တွေက အဆင်ပြေနေသော်လည်း server ဘက်တွင် Error ဖြစ်နေသော အခြေအနေမျိုး တွေမှာ 500 Internal Server Error ဆိုတာမျိုး ဖြစ်ပေါ်တတ်ပါသည်။ ဘယ်လို Error မှန်းမသိတဲ့အခါ (သို့မဟုတ်) ဘာ Error လည်း တိတိကျကျ မပြောပြချင်တဲ့ အခါမျိုးတွေမှာ server ဘက်က Error ဖြစ်နေပါတယ်ဆိုတဲ့ သဘောမျိုးနဲ့ 500

server error ဆိုပြီး ပြန်ပေးတတ်ပါသည်။

13. လက်ရှိအချိန်တွင် Security bug fixes ပေးနေသော php version များမှာ version 8.1.3, version 8.0.28 နှင့် version 7.4.28 တို့ ဖြစ်သည်။

14. Laravel Eloquent တွင် soft delete သည် data များကို database ထဲတွင် အပီး တိုင် ဖျက်ပစ်လိုက်တာမျိုး မဟုတ်ပဲ restore ပြန်လုပ်နိုင်အောင် ပြုလုပ်ထားခြင်း ဖြစ်သည်။ တစ်နည်းအားဖြင့် relationship ရှိသော table များတွင် မတော်တဆ data တစ်ခုခုကို ဖျက်မိလိုက်ပါက ပြန်လည်ရယူနိုင်အောင် softDeletes() အား အသုံးပြုနိုင်ပါသည်။

15. Laravel Migration မှာ နောက်ဆုံး migrate လုပ်ခဲ့တဲ့ဖိုင်ကို

‘php artisan migrate:rollback’ command ရေး၍ နောက်ပြန်ဆုတ်နိုင်ပါသည်။

16. Laravel တွင် Cache clear လုပ်ရန်အတွက်

‘php artisan cache:clear’ command ကို run ပေးရပါသည်။