

{echo}

Digital Magazine for Web Designer & Developer

Issue #8 • October 2013 • echo.myanmarlinks.net

INTRODUCTION TO COMPOSER

PHP DATA OBJECTS
(PDO)

NEW FEATURES
FROM PHP 5.4

PHP EXCEPTIONS

AJAX
FILE UPLOAD

PHP
INHERITANCE



INTERVIEW WITH
KO KO YE

Table of Content

Introduction to Composer	10
Build your own development environment in Ubuntu	22
Interview with Ko Ko Ye	35
RPi Introduction	44
Building an HTML5 Contact Form	48
Raspberry Pi Setup	53
PHP Data Objects (PDO)	61
Ajax file upload	69
PHP Exceptions	80
New Features from PHP 5.4	87
jQuery User Interface	92
PHP Inheritance	98

Echo Team

Editor

Thiha

Cover Designer

Hein Zaw Htet

Layout Designer

Hein Zaw Htet

Proof Reader

Echo Team

Website

echo.myanmarlinks.net

Facebook

fb.me/EchoMagazine.mm

Email

echo.emagz@gmail.com

{ echo } is the product of
MyanmarLinks Web Development Team

Preface

ကျွန်တော်တို့ ဒီတစ်ပတ်တော့ Featured အနေနဲ့ PHP ကို ရွေးချယ်ထားပါတယ်။ PHP ဟာ အစာများကိုပိုင်းမှာ တိုးတက်မှုအရမ်းမြန်ဆန် လာနေပါတယ်။ အတော်ပိုင်းကာလတွေတုန်းက PHP Developer တွေ အနေနဲ့ ကိုယ်လေ့ ကိုယ်ထိုး ပဲခူးရောက်ရောက်ဆိုပြီး သွားနေခဲ့ကြပါတယ်။ ကိုယ်တို့တွင်ထားတဲ့ Framework ကို ကိုယ်နည်းကိုယ်ဟန်နဲ့ တည်ဆောက်ခဲ့ကြပြီး ကိုယ်လမ်းကိုယ်ဖောက် ကိုယ်ခရီးကိုယ်လျှောက် ဆိုသလို ခရီးဆက်နေခဲ့ရင်းက Symfony Framework ရဲ့ ဦးဆောင်မှုနောက်မှာ ပြန်လည် စုစုပေါင်းမြဲလာပါတယ်။ အဲဒီလို ပြန်လည် စုစုပေါင်းမှုရှိလာတဲ့ အနေအထားမှာ နိုက်တည်းက အများပြီးရှိနေတဲ့ အင်အားစုတွေဟာ တစ်နေရာတည်းမှာ စုစုပေါင်းအခါ အလွန်ကြီးမားတဲ့ အပြောင်းအလဲနဲ့ အင်အားစုတွေ ဖြစ်လာပေါ်ပါတယ်။ အဲဒီလို စုစုပေါင်းမြဲလာတဲ့ အပြင် နို့မှုလ Language ဖြစ်တဲ့ PHP Version အဆင့်ဆင့်ကို မြန်မြန်ဆန်ဆန် မြင့်တင်ပြီး Object-Oriented Feature တွေကို အဆင့်မြှင့်မြင့် အထောက်အပံ့ပေးနိုင်အောင် လုပ်ဆောင်လာပါတယ်။ နောက်ထပ် PHP ကို ပြည့်စုံစေတဲ့ လမ်းကြောင်းတစ်ခုကတော့ အခြား ROR, NodeJS စတာတွေကနေ နည်းပညာတွေ စီးဝင်လာတာ ဖြစ်ပါတယ်။

အဲဒီလို စီးဝင်လာတဲ့ အတွက် PHP ရဲ့ စွမ်းဆောင်ရည် တစ်စထက် တစ်စ ပိုမိုကောင်းမွန် လာခဲ့ပြီး အလွယ်တကူ ပေါင်းစပ်အသုံးပြန်ပါတယ်။ Component အစိတ်အပိုင်းတွေအဖြစ် အသုံးပြန်ပါတယ်။ အရင်လို Framework တစ်ခုတည်းကို မို့ခိုပြီး ရေးရတာ မဟုတ်တော့ပဲ Framework တွေရဲ့ Component တွေကို စိတ်ကြိုက် ချိတ်ဆက် အသုံးပြန်ပါတယ်။ နောက်ထပ် အရေးကြီးတဲ့ PHP ရဲ့ အရွှေ့ကတော့ Programming Concept, Development Concept တွေအပေါ်မှာ အရင်ကထက်ပိုပြီး ချဉ်းကပ်လာနိုင်တာ ဖြစ်ပါတယ်။ ကျွန်တော်တို့ {echo} ရဲ့ ပုံမှန် တင်ဆက်နေကြ အခန်းကဏ္ဍတစ်ခု အဖြစ် အင်တာပျူးကဏ္ဍကိုလဲ တင်ဆက်ထားပြီး ဒီတစ်ပတ် ရွေးချယ်ဖြစ်သူကတော့ ကိုကိုရဲ ဖြစ်ပါတယ်။ Resberry Pi နဲ့ ပတ်သက်တဲ့ အကြောင်းအရာတွေနဲ့ Ubuntu နဲ့ ပတ်သက်တဲ့ အကြောင်းအရာတွေကို ပြည့်ပြည့်စုံ မေးမြန်ထားပါတယ်။ အဲဒီအပြင် ကိုကိုရဲကိုယ်တိုင် ရေးသားထားတဲ့ Resberry Pi နဲ့ ပတ်သက်တဲ့ အကြောင်းအရာတွေကိုလဲ ထည့်သွင်းပေးထားပါတယ်။ ကျွန်တော်တို့ အနေနဲ့ { echo } Digital Magazine ကို Web Development အပိုင်းမှာ အထောက်အပံ့ပေးနိုင်စွဲ ရည်ရွယ်ချက်နဲ့ လစဉ်ပုံမှန် တင်ဆက်လာတာ အခု အကြိမ်ဟာ (၈) ကြိမ်မြောက်ဖြစ်ပါတယ်။ အားလုံး စိတ်၏ ချမ်းသာခြင်း ကိုယ်၏ ကျိုးမာခြင်းနဲ့ ပြည့်စုံကြပါစေလို့ ဆုမွန်ကောင်း တောင်းအပ်ပါတယ်။

သီဟ



World Wide Myanmar

One Stop IT Station

inquiry@worldwidemyanmar.com

<http://www.worldwidemyanmar.com>

"

Offering feature rich, on-demand reliable and secure web hosting services is why our customers stay here. We are constantly working to improve our services and the product line we offer.

- Min Aung

Founder, WWM Company

"

Not happy with the service? Can't get going with the product? We would be surprised, as we have the most advanced hosting platform in the world, but if you are unhappy then we offer a no-fuss money-back guarantee.

- Su Su Myint

Marketing, WWM Company

REGISTER

DOMAIN NAME

HOSTING COMPANY

PREMIUM WEB HOSTING

PROFESSIONAL

WEB DESIGN WEB DEVELOPMENT

EDUCATION

COMPUTER TRAINING CENTRE

Our services include website designing and development, custom web application development, ecommerce web solutions, SEO services, Domain Name registration and Web hosting, Reseller Web Hosting, VPS, Dedicated servers in myanmar.



- WE CAN HELP YOU DEFINE YOUR NEEDS AND FULFILL THOSE NEEDS.
- WE LISTEN OUR CLIENTS 24 HOURS / 7 DAYS.
- LET US SOLVE YOUR BUSINESS PROBLEMS.

- CALL US: 09 4500 43919 (Myanmar) , +60 16 342 3594 (Malaysia)
- VISIT US: No. 231A, 1st Floor, Ngu War Street, Ahlone Township, Yangon, Myanmar.
- KNOW US: www.worldwidemyanmar.com



ZawGyi Mart

Online Shopping & Delivery Service

အိမ်တိုင်ရှာ အကောက် (အော့) ပို့ဆောင်ပေးပါသည်။



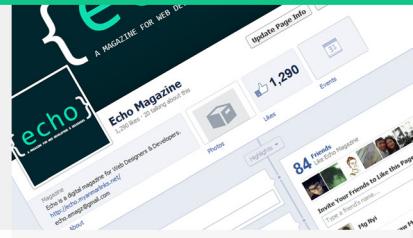
www.zawgymart.com



INTERACT WITH US

Find Us
on Facebook

www.facebook.com/EchoMagazine.mm



Leave Comments
on Issue Pages

www.echo.myanmarlinks.net



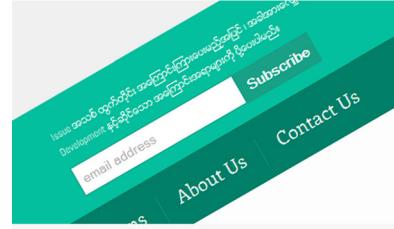
Contact Us
from Our Site

www.echo.myanmarlinks.net/contact



Subscribe
Mailing List

www.echo.myanmarlinks.net



MYANMAR THEMES

WWW.MYANMARTHEMES.COM

ADVERTISING INFO

FULL PAGE

A4

HALF PAGE

A4

QUATER

A4

20000 KYATS

12000 KYATS

8000 KYATS

{echo} ၏အသာချက်များ

အစောင်အရေအတွက်အကန်အသက်ပရှိခြင်း

တိက်ရိုက် Link များထည့်သွင်းနိုင်ခြင်း

ကြော်ပြော်များကိုကဲလာကြပ်စုံဖြင့်ထည့်သွင်းနိုင်ခြင်း

အဆုံးဖြန့်ဝေသည့်အတွက်လုပ်တို့များခြင်း



echo-june-2013.zip
JbKw
20 MB • 12774 Views
20 days ago



echo_june_2003.zip
std4
20 MB • 6068 Views
21 days ago

နောက်ဆုံး(၆၅) Issue ကို Download ရမယ့်အတွက်ဖြစ်ပါသည်
ကို ၂၀၁၅ ခုနှစ်၊ ဇန်နဝါရီလ ၁၇ ရက်နေ့တွင်ဖော်ပြန်ပါသည်။

ကြော်ညာများကို ၂၈ ရက်စောင်းနောက်စုံးထားပြီး ပေးပို့နိုင်ပါသည်

တိုက်ဘာအခန်းစာ

ပို့ဆောင်ရွက်မှုများ၊ ပုဂ္ဂန်တောင်မြို့များ၊ ရန်ကုန်တိုင်း

နှင့် - ဝါယာမြေ၊ ဝါယာမြေ

Email - echo.emagz@gmail.com

Website - <http://echo.myanmarlinks.net>

Myanmar Links

Professional Web Development

လူပြီးမင်းတို့၏ လုပ်ငန်းနှင့် ကိုက်ညီသည့် အရည်အသွေးကောင်းမွန်သော Web Application များ၊
Website များ ရေးသားပေးလျက်ရှိပါသည်...



Web Application
Development



Website
Development

Website တစ်ခုကို ဈေးအသက်သာဆုံး
တည်ဆောက်လိုပါသည်:

www.myanmarthemes.com မှ သင်ကြိုက်ယှဉ်သက်ရှု Theme
တစ်ခုချော့ချယ်ထိယူပြီး တည်ဆောက်နိုင်ပါသည်။



095166242
24 HOURS A DAY



Domain
Registration

Myanmar Links

Building 1, Room 8

Botahtaung Pagoda Road, Pazundaung Township

Tel.: 095166242, 095083160

Email: info@myanmarlinks.net

Website: www.myanmarlinks.net



Myanmar Links

Professional Web Development

● Web Professional Foundation

ကုန်ပျော်အခြေခံပြီး Web Professional တစ်ယောက်အဖြစ်
အသက်မွေးဝမ်းကြောင်း ပြရှိသူများကို အခြေခံမှုပြီး Website
တစ်ခုကို ကောင်းမွန်စွာ ရေးဆွဲနိုင်သည်အထိ စနစ်တကျ
သင်ကြားပေးပါသည်။

● Web Professional Advanced

Web Professional Advanced တွင် PHP ကို အစိတ်အသေး
သင်ကြားပေးပြီး Project တစ်ခုပြီးဆုံးအောင်
လုပ်ဆောင်ရပါသည်။



Projector

ဖြင့်သေချာစွာ ရှင်းပြခြင်း

Web Professional Advanced တန်းခွဲသစ်ကို
အောက်တိုဘာ (၄) ရက်နေ့တွင် ဖွင့်လှစ်ပါမည်။



095083160



Practical Project

များကို သင်ကြားသူနှင့်တော်
လုပ်ဆောင်စေခြင်း



Video Screencast

များဖြင့်
သင်ကြားပေးခြင်း

Myanmar Links

Building 1, Room 8

Botahtaung Pagoda Road, Pazundaung Township

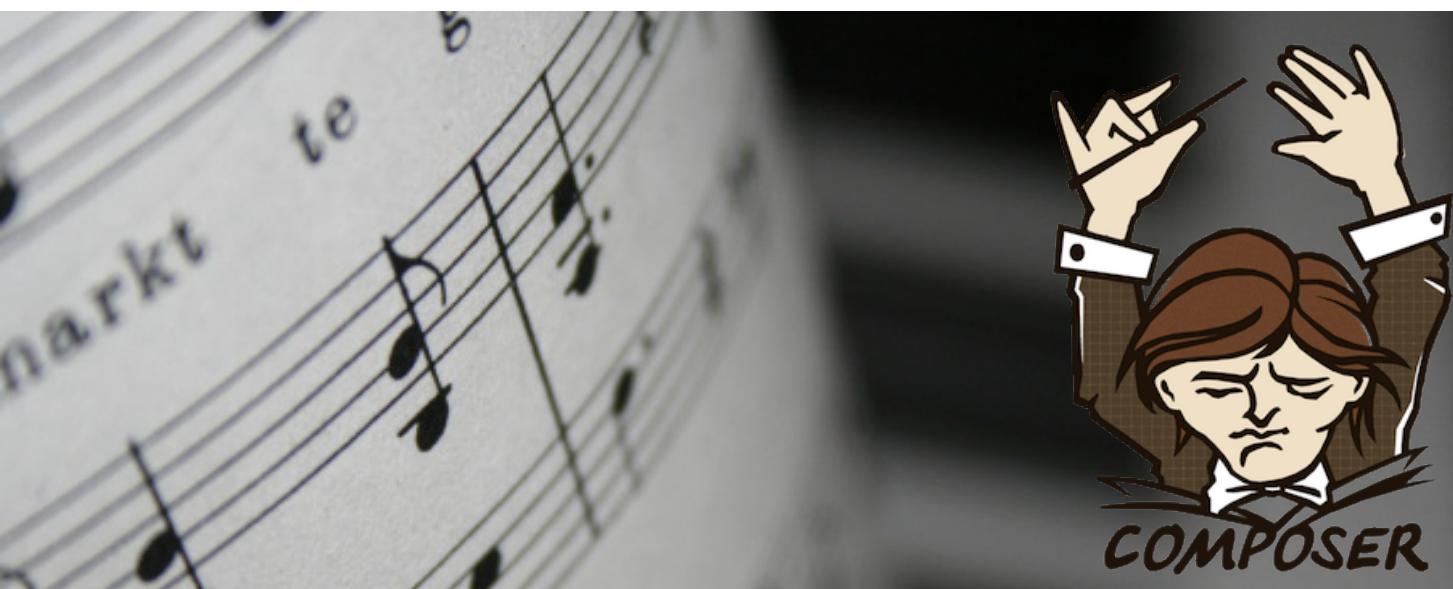
Tel.: 095083160

Email: info@myanmarlinks.net

Website: www.myanmarlinks.net

■ Introduction to Composer

by Thiha



about

THIHA

Founder of Myanmar Tutorials and Myanmar Links

soethiha.me

[/soethihanaung](https://www.facebook.com/soethihanaung)

[@phpcrazy](https://twitter.com/phpcrazy)

[/phpcrazy](https://github.com/phpcrazy)

Intro

Composer ဆိတ်သာ PHP အတွက် Dependency Management အတွက် အသုံးပြန်စဲတဲ့ Tool တစ်ခု ဖြစ်ပါတယ်။ ကျွန်ုတ်တို့ Project တွေ ရေးတဲ့အခါမှာ မို့မို့ အသုံးပြထားတဲ့ Library တွေကို Composer နဲ့ အသုံးပြုပြီး ကြော်ပေးလို့ ရာဘလို သူအလိုလို အခွန်လိုင်းက download ချ ထည့်သွင်းပေးတဲ့ နေရာမှာလဲ အသုံးပြန်ပါတယ်။

Dependency Management

Composer ဆိတ်ဘက် Package Manager မဟုတ်ပါဘူး၊ ဒါပေမယ့် Package တွေကို ကိုင်တွယ် ဖြေရှင်း ပေးတဲ့ tool တစ်ခုပါ။ Project တစ်ခုခြင်းမီအလိုက် မို့မို့ထားရတဲ့ library တွေကို folder တစ်ခုထဲကို download ချပေးမယ်၊ လိုအပ်တာတွေ အကုန်ထည့်သွင်းပေးမှာ ဖြစ်ပါတယ်၊ (ပုံမှန် အားဖြင့်တော့ vendor ဆိတ် တဲ့ folder ထဲကို ထည့်ပေးပါတယ်)

Composer ရဲ့ အိုင်ဒီယာက အသစ်တော့ မဟုတ်ပါဘူး။ node.js တို့ ROR တို့မှာ ရှိတဲ့ npm, bundler တို့နဲ့ သဘောတရားခြင်း တူပါတယ်။

Composer အနေနဲ့ အောက်ပါ ပြဿနာတွေကို ဖြေရှင်းပေးနိုင်ပါတယ်

1. သင့်အနေနဲ့ project တစ်ခု တည်ဆောက်တဲ့အခါ မို့မို့ထားတဲ့ libraries တွေ အများကြီး ရှိနေတယ်ဆိုရင်
2. သင့်ရဲ့ project မှာ အသုံးပြထားတဲ့ libraries တွေဟာ အခြား Library တွေကို မို့မို့နေတယ်ဆိုရင်
3. သင့်ရဲ့ project မှာ သင်ကိုယ်တိုင် တည်ဆောက်ထားတဲ့ အရာတွေ အပေါ်မှာ မို့မို့ ကြော်ချင်တယ် ဆိုရင်
4. composer အနေနဲ့ ဘယ် version ကို install လုပ်ရမယ်ဆိုတာ ရှာဖွေပေးမှာ ဖြစ်ပြီး install ပြုလုပ်ပေးပါလိမ့်မယ်။ (download ချပေးပါလိမ့်မယ်)

Declaring Dependencies

ကျွန်ုတ်တို့ Project တစ်ခု ရေးတယ်ဆိုပါစို့။ အဲဒီ Project မှာ logging အတွက် Library တစ်ခုကို အသုံးပြုဖို့ လိုနေပါတယ်။ အဲဒီအတွက် monolog ဆိတ် တွေကို အသုံးပြုဖို့ ရွေးချယ်လိုက်ပါတယ်။ (ကျွန်ုတ်တို့ composer အပေါ် မှတည်ပြီး အသုံးပြုလိုရတဲ့ library တွေကို စုစည်းပေးထားတဲ့ site တစ်ခု ရှိပါတယ်။ အဲဒီကတော့ <https://packagist.org/> ပါ။



Packagist

The PHP package archivist.

[Submit Package](#)

Packagist is the main [Composer](#) repository. It aggregates all sorts of PHP packages that are installable with Composer.
[Browse packages](#) or [submit your own](#).

Getting Started

Define Your Dependencies

Put a file named `composer.json` at the root of your project, containing your project dependencies:

```
{
    "require": {
        "vendor/package": "1.3.2",
        "vendor/package2": "1.*",
        "vendor/package3": ">=2.0.3"
    }
}
```

Install Composer In Your Project

Publishing Packages

Define Your Package

Put a file named `composer.json` at the root of your package, containing this information:

```
{
    "name": "your-vendor-name/package-name",
    "require": {
        "php": ">=5.3.0",
        "another-vendor/package": "1.*"
    }
}
```

This is the strictly minimal information you have to give.

အဲဒီမှာ သွားရှာမယ်ဆိုရင် နောက်ဆုံးထွက်တဲ့ Version အမည်တွေက အစ သိရမှာ ဖြစ်ပါတယ်။

ကျွန်တော်တို့ ရှေ့ဆက်လိုက်ရအောင် ...

monolog ကို အသုံးပြုဖို့ အတွက် `composer.json` ဆိုတဲ့ file တစ်ခုကို တည်ဆောက်လိုက်ပြီး ကိုယ့် project အသုံးပြုမယ် web server ထဲကို ထည့်ပေးလိုက်ဖို့ လိုပါတယ်။ အဲဒီအပြင် composer ကိုလဲ `install` လုပ်ဖို့လိုပါတယ်၊ ဘယ်လို `install` လုပ်ရမယ် ဆိုတာကိုတော့ <http://getcomposer.org/doc/00-intro.md> မှာ လေ့လာနိုင်ပါတယ်။ `composer.json` ဆိုတဲ့ file မှာ အောက်မှာ ရေးထားတဲ့အတိုင်း ရေးလိုက်ပါ၊ ဒီနေရာမှာ တစ်ခု သတိထားဖို့ လိုတာက `composer.json` ဟာ JSON Format ကို လိုက်နာဖို့ လိုပါတယ်။ single quote နဲ့ ရေးလို့ မရပဲ double quote နဲ့သာ ရေးရမှာ ဖြစ်ပါတယ်။

```
{
    "require": {
        "monolog/monolog": "1.6.*@dev"
    }
}
```

ပြီးရင် composer ကို `install` လုပ်ပေးဖို့ လိုပါမယ်။

```
soethiha@ubuntu:~/Public/thiha.me/public/echo_composer$ composer install
Loading composer repositories with package information
Installing dependencies (including require-dev)
- Installing psr/log (1.0.0)
  Loading from cache

- Installing monolog/monolog (dev-master 1518320)
  Cloning 1518320bec1f1583a0c514491528d171501dc781

monolog/monolog suggests installing mlehner/gelf-php (Allow sending log messages
to a GrayLog2 server)
monolog/monolog suggests installing ext-amqp (Allow sending log messages to an A
MQP server (1.0+ required))
monolog/monolog suggests installing ext-mongo (Allow sending log messages to a M
ongoDB server)
monolog/monolog suggests installing doctrine/couchdb (Allow sending log messages
to a CouchDB server)
monolog/monolog suggests installing raven/raven (Allow sending log messages to a
Sentry server)
Writing lock file
Generating autoload files
soethiha@ubuntu:~/Public/thiha.me/public/echo_composer$
```

composer.json file ရှိတဲ့ အထဲကို command line က သွားပြီး composer install လို့ ရှိက်လိုက်မယ် ဆိုရင်
ပုံမှာဖြတော်များတဲ့အတိုင်း monolog ကို install လုပ်သွားတာ တွေ့ရပါလိမ့်မယ်။ composer အနေနဲ့ monolog ကို download ချင်သောမက
autoload.php ဆိုတဲ့ file တစ်ခုကိုလဲ စီစဉ်ပေးပါလိမ့်မယ်။

```
untitled (echo_composer) - Sublime Text (UNREGISTERED)

FOLDERS
▼ echo_composer
  ▼ vendor
    ► composer
    ► monolog
    ► psr
    autoload.php
  composer.json
  composer.lock
```

ကိုယ့်အနေနဲ့ ကိုယ့်ရဲ့ project မှာ monolog ကို သုံးချင်တယ်ဆိုရင် autoload.php file ကို require လုပ်ပေးရုံပါပဲ။

ကျွန်တော် အပေါ်မှာ တည်ဆောက်ထားတဲ့ ပုံစံအရ echo_composer ဆိုတဲ့ folder ထဲမှာ composer.json ရှိနေပါတယ်။ ကျွန်တော်ကတော်မူတဲ့ Composer က install လုပ်လိုက်လို့ အွန်လိုင်းက download ချလိုက်တာတွေ ဖြစ်ပါတယ်။ ကျွန်တော်တို့ monolog ကို သုံးချင်တယ်ဆိုရင် index.php မှာ ကျွန်တော် ရေးထားတဲ့အတိုင်း ရေးပြီး စမ်းလို့ရပါပြီ။

```
<?php

require "vendor/autoload.php";

use Monolog\Logger;
use Monolog\Handler\StreamHandler;

$log = new Logger('name');
$log->pushHandler(new StreamHandler(__DIR__ . '/log/project.log',
Logger::WARNING));

$log->addWarning('Your site has been under attacked!');
$log->addError('Stupid Error!');

?>
```

ကျွန်တော်တို့ composer.json မှာ ရေးခဲ့တဲ့ အပိုင်းကို ပြန်သွားကြရအောင်

```
{
    "require": {
        "monolog/monolog": "1.6.*@dev"
    }
}
```

require ဆိုတာကတော့ dependencies အတွက် ကိုယ့် project မှာ သယ် library တွေ လိုမယ် သယ် Library တွေ သုံးမယ်ဆိုတာ ပြပါတယ်၊ တစ်ဖက်မှာ package name ကို ရေးရပြီး နောက်တစ်ဖက်မှာ Package Version ကို ရေးပေးရပါတယ်။

1.0.* ဆိုရင် 1.0.0, 1.0.1, 1.0.2 စသည်ဖြင့် install လုပ်ပေးပါလို့ ဆိုလိုပါတယ်၊ ကိုယ့်အနေနဲ့ package version ကို ပုံစံ လေးမျိုးနဲ့ သတ်မှတ်နိုင်ပါတယ်။

Exact version: သင့်အနေနဲ့ သင့် project အတွက် dependency library ကို တိကျွတဲ့ version သတ်မှတ်နိုင်ပါတယ်၊ ဥပမာ - 1.0.2

Range: comparison operators တွေဖြစ်တဲ့ >, >=, <, <=, != စတာတွေကို အသုံးပြုပြီး project မှာ လိုအပ်မယ့် dependency library ရဲ့ version တွေကို သတ်မှတ်နိုင်ပါတယ်။ ဥပမာ - >= 1.0။ သင့်အနေနဲ့ multiple range ကို သတ်မှတ်ချင်တယ်

ဆိုရင်တော့ , နဲ့ ခံပြီး သတ်မှတ်နိုင်ပါတယ်။ ဥပမာ - >=1.0, < 2.0

Wildcard: သင့်အနေနဲ့ * ဆိုတဲ့ wildcard pattern ကို အသုံးပြုပြီး သင်လိုတဲ့ package version ကို သတ်မှတ်နိုင်ပါတယ်။ ဥပမာ - 1.0.* ဆိုရင် >=1.0, <1.1 နဲ့ အဓိပ္ပာယ်တူပါတယ်။

Next Significant Release (Tilde Operator): ဒါကတော့ Package version အသစ်ထွက်တိုင်း သုံးမယ်ဆိုတဲ့ သဘောဖြစ်ပါတယ် ~ ဆိုတဲ့ pattern နဲ့ သုံးရပါတယ်။

ပုံမှန်အားဖြင့် stable release တွေကိုသာ အသုံးပြုကြပြီး ကိုယ့်အနေနဲ့ Development Version, Alpha Version, Beta Version စတာတွေကို သုံးချင်တယ်ဆိုရင် @dev, @beta စသည်ဖြင့် ထည့်ပြီး သုံးနိုင်ပါတယ်။

Autoloading

ကျွန်ုတ်တို့ အနေနဲ့ ကျွန်ုတ်တို့ လိုအပ်တဲ့ library တွေကို composer.json မှာ require keyword နဲ့ ကြော်လိုက်တယ် ဆိုရင် autoload.php ဆိုတဲ့ file တစ်ခုကို composer က ဖန်တီးပေးပါလိမ့်မယ်။ ကျွန်ုတ်တို့ project ရေးတဲ့အခါမှ require 'vendor/autoload.php' ဆိုပြီး ရေးလိုက်တာနဲ့ ကျွန်ုတ်တို့ သုံးချင်တဲ့ library တွေကို သုံးလို့ ရပါပြီ။ ဒါပေမယ့် ကျွန်ုတ်တို့မှာ သူများ ရေးပြီးသား library တွေတင် မကပါဘူး၊ ကိုယ်တိုင်ရေးထားတဲ့ Library တွေလဲ ရှိနေနိုင်ပါတယ်။ အဲဒီအတွက် ကိုယ်ရေးထားတဲ့ Library တွေကိုလဲ composer.json မှာ ထည့်သွင်း ကြော်ပေးစွဲ လိုပါမယ်။

ကိုယ့် Project မှာ ကိုယ်ရေးထားတဲ့ Library တွေကို ထည့်သွင်းကြော်ပေးတဲ့ နေရာမှာ လိုအပ်ချက်အရ သုံးမျိုး ရှိနိုင်ပါတယ်။

- Namespace:** ကိုယ့် project မှာ namespace အနေနဲ့ autoload လုပ်ချင်တယ်ဆိုရင် PSR-0 standard အရ autoload လုပ်နိုင်ပါတယ်။
- Class:** ကိုယ့် project မှာ class ကို တစ်ခါတည်း ချိတ်ဆက်ပြီး autoload လုပ်နိုင်ပါတယ်။
- File:** ကိုယ့် project အတွက် လိုမယ့် file ကို တန်းပြီး autoload လုပ်နိုင်ပါတယ်။

အဲဒီ သုံးခုကို ကျွန်ုတ် နမူနာတစ်ခုနဲ့ ရှင်းပြေသွားပါမယ်။

Namespace Autoload

ပထမ ဦးဆုံးနမူနာ ရေးကြည့်မယ့် အပိုင်းကတော့ Namespace Autoload ပဲ ဖြစ်ပါတယ်။ Composer အနေနဲ့ Namespace ကို autoload လုပ်ဖို့အတွက် PSR-0 Standard ကို လိုက်နာထားပါတယ်။ ကျွန်တော်တို့ ဘုံးမပုံ Library က src/Car/Toyota.php ဆိုပါမဲ့ ...

```
[  
    "require": {  
        "monolog/monolog": "1.6.*@dev"  
    },  
    "autoload": {  
        "psr-0": {"Car\\": "src/"}  
    }  
]
```

ကျွန်တော် အပေါ်မှာ ရေးထားတဲ့ ပုံစံအတိုင်း ရေးပေးဖို့ လိုပါမယ်။ folder structure ကတော့ အောက်မှာ ပြထားတဲ့ ပုံအတိုင်း ဖြစ်ပါတယ်။

The screenshot shows a terminal window with the following details:

- Title Bar:** ~/Public/thiha.me/public/echo_composer/src/Car/Toyota.php (echo_composer) - S
- Left Panel (Folders):**
 - echo_composer
 - log
 - src
 - Car
 - Toyota.php
 - vendor
 - composer.json
 - composer.lock
 - index.php
- Right Panel (Code Editor):**
 - File: Toyota.php
 - Content:

```
1 <?php  
2  
3 namespace Car;  
4  
5 class Toyota {  
6  
7     public $name = '';  
8  
9     public function __construct($name) {  
10         $this->name = $name;  
11     }  
12 }  
13  
14 ?>
```

src က folder name ပါ။ Car ဆိုတဲ့ folder က namespace ရဲ့ အမည်ဖြစ်ပါတယ်။ Toyota.php ကတေသာ Class name အတွက် file ဖြစ်ပါတယ်။ composer.json မှာလဲ update လုပ်ပြီးပြီ namespace, class တွဲလဲ ဆောက်ပြီးပြီ ဆိုရင်တော့ command line က သွားပြီး

```
composer dumpautoload
```

ဆိုပြီး autoload file ကို update လုပ်ပေးနဲ့ လိပ်ပါတယ်။ ပြီးရင်တော့ ကျွန်တော်တို့ Project မှာ Car/Toyota ကို သုံးလိုရပါပြီ။ ကျွန်တော် နှမာနာအနေနဲ့ သုံးပြုပါမယ်။

```
require "vendor/autoload.php";  
  
use Car\Toyota;  
  
$car = new Toyota('Belta');  
echo $car->name;
```

run ကြည့်လိုက်မယ်ဆိုရင် Belta ဆိုပြီး ထွက်လာပါလိမ့်မယ်။ ကျွန်တော်တို့ အနေနဲ့ Car ဆိုတဲ့ Namespace အောက်မှာ Nissan, Mazda ဆိုတဲ့ Class တွေ ရှိလာမယ်ဆိုရင် Car ဆိုတဲ့ Folder အောက်မှာ ထည့်သွားရုံပါပဲ။ ဥပမာ အနေနဲ့ Car ဆိုတဲ့ folder အောက်မှာ Nissan.php ဆိုပြီး ထပ်ရေးကြည့်ပါမယ်။

```
<?php  
  
namespace Car;  
  
class Nissan {  
    public $name = '';  
  
    public function __construct($name) {  
        $this->name = $name;  
    }  
}  
?>
```

အဲဒီ Nissan ဆိုတဲ့ Class ကို သုံးချင်တယ်ဆိုရင် တစ်ဖက်စာမျက်နှာမှာ ပြထားတဲ့အတိုင်း ရေးပြီး သုံးနိုင်ပါတယ်။

```

require "vendor/autoload.php";

use \Car;

$toyota_car = new Car\Toyota('Belta');
$nissan_car = new Car\Nissan('AD Van');
echo $toyota_car->name . "<br />";
echo $nissan_car->name;

```

Use နေရာမှာ \Car ဆိုတဲ့ Namespace ကိုပဲ ကြော်ထားပါတယ်၊ ပြီးတော့မူ Car\Toyota Car\Nissan ဆိုပြီး
သုံးလို့ရပါတယ်။ run ကြည့်မယ်ဆိုရင်

Belta

AD Van

ဆိုပြီး result ထွက်လာပါလိမ့်မယ်။

Class Autoload

Namespace autoload အနေနဲ့ သွားနိုင်သလို Class ကိုတစ်ခါတည်း ညွှန်ပြီးလဲ သုံးနိုင်ပါတယ်။ classmap လို့ composer မှာ သုံးပါတယ်။

ကျွန်ုတ်တို့ Project မှာ Namespace အနေနဲ့ သုံးတာ မဟုတ်ပဲ Class ကို တိုက်ရိုက်ညွှန်ပြီး သုံးတာလဲ ဖြစ်နိုင်ပါတယ်။
အဲဒီလို သုံးချင်တဲ့အပါ classmap နဲ့ composer.json မှာ ကြော်ပြီး သုံးနိုင်ပါတယ်။

```

{
    "require": {
        "monolog/monolog": "1.6.*@dev"
    },
    "autoload": {
        "psr-0": {"Car\\": "src/"},
        "classmap": [
            "config"
        ]
    }
}

```

classmap ဆိုပြီး autoload မှာ ထပ်ဖြည့်ထားပါတယ်။ ဆိုလိုတဲ့ အမိမျိုးကတော့ config ဆိုတဲ့ folder ထဲက file တွေအားလုံးကို စစ်မယ်၊ အဲဒီ file တွေထဲက class တွေကို mapping လုပ်မယ်လို့ အမိမျိုးပါတယ်။ ကျွန်တော်တို့ စမ်းသပ်ကြည့်ရအောင်

config ဆိုတဲ့ folder ထဲမှာ Database.php ဆိုတဲ့ file တစ်ခု တည်ဆောက်လိုက်ပါမယ်။ အဲဒီအထဲမှာတော့ ကျွန်တော်ရေးထားတဲ့အတိုင်း ရေးလိုက်ပါ။

```
<?php

class DB {
    public $dbname = '';

    public function __construct($dbname) {
        $this->dbname = $dbname;
    }
}

?>
```

အဲဒီလို ရေးပြီးရင် command line ကို သွားပြီး composer dumpautoload ဆိုပြီး ထပ်ခေါ်ပေးဖို့ လိုပါတယ်။

composer dumpautoload

အဲဒီလို ခေါ်လိုက်မှသာ လိုအပ်တဲ့ class တွေကို mapping လုပ်ပေးမှာ ဖြစ်ပါတယ်။ အဲဒီလို classmap လုပ်ပြီးပြီဆိုရင်တော့ သုံးလို့ ရပါပြီ၊ ထုံးစံအတိုင်း index.php ကို သွားပြီး စမ်းကြည့်ရအောင် ...

```
$db = new DB('test');
echo $db->dbname;
```

runကြည့်မယ်ဆိုရင် DB ဆိုတဲ့ class ကို autoload လုပ်ပြီးသား ဖြစ်သလို သုံးလို့ရနေတာ တွေ့ ရပါလိမ့်မယ်။ ဒီနေရာမှာ တစ်ခု သတိထားဖို့လိုတာက config ဆိုတဲ့ folder ထဲမှာ တစ်ခုခု တည့်မယ် ဆိုတာနဲ့

composer dumpautoload

ပြန်လုပ်ပေးဖို့ လိုတာပါ။ အဲဒီလို အသစ်ထည့်တိုင်း classmap ပြန်မလုပ်ချင်ရင် Namespace Autolaod နဲ့ သုံးနိုင်ပါတယ်။

File Autoload

Namespace Autoload, Class Autoload ပြီးရင်တော့ File Autoload ပဲ ကျွန်ပါတော့တယ်။ ဒါကတော့ ကျွန်တော်တို့ PHP မှာ သုံးနေကြအတိုင်းပဲ ဖြစ်ပါတယ်။ RebornCMS မှာဆိုရင်တော့ helper function တွေကို file autoload နဲ့ သုံးထားပါတယ်။ ကျွန်တော်တို့ Projectမှာလဲ file autoload ကို စမ်းကြည့်ရအောင်

```
{  
    "require": {  
        "monolog/monolog": "1.6.*@dev"  
    },  
    "autoload": {  
        "psr-0": {"Car\\": "src/"},  
        "classmap": [  
            "config"  
        ],  
        "files": [  
            "helper/helper.php"  
        ]  
    }  
}
```

files ဆိုပြီး autoload မှာ helper/helper.php ဆိုတဲ့ file ကို ချိတ်လိုက်ပါတယ်။ helper ဆိုတဲ့ folder တစ်ခုကို တည်ဆောက်လိုက်ပြီး helper.php ဆိုတဲ့ file တစ်ခုကို helper folder ထဲမှာ ဆောက်လိုက်ပါ။ ပြီးရင် ကျွန်တော် ရေးထားသလို ရေးလိုက်ပါ။

```
<?php  
  
function Currency_Converter($amount) {  
    return $amount * 980;  
}  
  
?>
```

အဲဒီလို ဆောက်ပြီးသွားပြီ ဆိုရင်တော့ ထုံးစံအတိုင်း composer dump-autoloader ဆိုပြီး ပြန်လုပ်ပေးနဲ့ လိုပါလိမ့်မယ်။ ပြီးရင် Currency_Converter ဆိုတဲ့ functionကို စမ်းကြည့်ပါမယ်။ index.php မှာ ထပ်မံပ္ပါယ်ပြီး ရေးကြည့်ပါမယ်။

```
$kyats = Currency_Converter(13000);  
echo $kyats;
```

run ကြည့်မယ်ဆိုရင်တော့ `kyats` ရဲ့ value ကို ရိုက်ထွက်ပြေတာ တွေ့ရပါလိမ့်မယ်။

ကျွန်ုင်တော်တို့ ဒီလောက်ဆိုရင် Composer ရဲ့ Dependency Management အဖိုင်းရယ်၊ autoloading အဖိုင်းတွေ ဖြစ်တဲ့ namespace autoload, class autoload, file autoload အဖိုင်းတွေကို သဘော ပေါက်နားလည်လိမ့်မယ်လို့ ယူဆပါတယ်။

ရှေ့ဆက်ပြီး Composer ကို အသုံးပြုပြီး Package တစ်ခု တည်ဆောက်တဲ့ ပုံစံကို ဆက်ရေးမှာ ဖြစ်ပါတယ်။

reference : [Composer Documentation](#)

■ Build your own development environment in Ubuntu

by Thiha



about

THIHA

Founder of Myanmar Tutorials and Myanmar Links



soethiha.me



[/soethihanaung](https://www.facebook.com/soethihanaung)



[@phpcrazy](https://twitter.com/@phpcrazy)



[/phpcrazy](https://github.com/phpcrazy)

ကျွန်တော်တို့ အနေနဲ့ Web Development လုပ်တွေမယ်ဆိုရင် Development Environment ဆိုတာ မဖြစ်မနေတည်ဆောက်ရမှာ ဖြစ်ပါတယ်။ အဲဒီလို့ တည်ဆောက်တဲ့ နေရာမှာလဲ Operating System အပေါ် အခြေခံပြီး တည်ဆောက်ကြရပါတယ်။ Microsoft Visual Studio ကို အသုံးပြုပြီး ရေးရတဲ့ ASP.NET စတာတွေအတွက်တော့ Windows OS အပေါ်မှာပဲ သုံးရမှာ ဖြစ်ပေမယ့် Open Source (ဥပမာ - Apache, MySQL, PHP, Ruby on Rails, NodeJS) စတာတွေ အတွက်ကတော့ Linux အခြေပြု OS တွေနဲ့ အသုံးပြုပြီး Development လုပ်တာ အကောင်းဆုံး ဖြစ်ပါတယ်။ ဘာကြောင့်အဲဒီလို့ လုပ်သုတေသနသဲ့ဆိုတဲ့ အချက်တွေကို လေ့လာကြည့်မယ်ဆိုရင်

- ကျွန်တော်တို့ ရေးထားတဲ့ Application တွေကို အမှန်တကယ် အွန်လိုင်းပေါ် တင်တဲ့အခါ အသုံးပြုရမယ့် ဆာဟတွေဟာ Linux Server တွေ ဖြစ်နေပါတယ်။
 - Windows OS ကို သုံးပြီးရေးမယ်ဆိုရင် File တစ်ခုခြင်းထိ အလိုက် အသုံးပြုရမယ့် Permission အပိုင်းတွေကို သိနိုင်မှာ မဟုတ်တဲ့အတွက် အွန်လိုင်းပေါ် တင်တဲ့အခါ File Permission ပြဿနာတွေ တက်နိုင်ပါတယ်။ File Permission ဆိုတာ Application Security အတွက်ပါ အရမ်း အရေးကြီးတဲ့အတွက် File Permission ကို မသိတာ အလွန်အရေးကြီးတဲ့ ပြဿနာဖြစ်ပါတယ်။
 - Linux OS မှာ Owner Permission အပိုင်းတွေ ပါဝင်ပြီးဖြစ်တဲ့အတွက် ဘယ် file ကိုတော့ ဘယ်သူက Access လုပ်ခွင့်ရှိတယ်၊ (Read, Write, Execute လုပ်ခွင့်ရှိတယ်) စသည်ဖြင့် ခွဲခြားထားနိုင်ပါတယ်။ Windows ဖက်မှာတော့ အဲဒီလို လုပ်လိုရမှာ မဟုတ်ပါဘူး။
 - Linux OS အပေါ်မှာ အသုံးပြုမယ်ဆိုရင် (ဥပမာ Nginx, Apache) စတာတွေဟာ တစ်သားတည်းကျနေတဲ့ ခံတာချက်ကို ပေးစွမ်းနိုင်ပါတယ်။
 - Linux OS အပေါ်မှာ ရေးမယ်ဆိုရင် နောက်ထပ်ရလာတဲ့ အားသာချက် တစ်ခုကတော့ Command Line Interface နဲ့ အသားကျ ထိတွေ့ပြီးသား ဖြစ်နေတာပါ။ အဲဒီလို Command Line Interface အပိုင်းကို ထိတွေ့ပြီးသား ဖြစ်တဲ့အတွက် အွန်လိုင်းပေါ်မှာ အမှန်အကန် တည်ဆောက်တဲ့အခါ ပိုမိုလွယ်ကူသွားစေပါတယ်။

ကျန်တော် ဒါ ဆောင်းပါးမှာတော့ Linux OS တစ်ခု ဖြစ်တဲ့ Ubuntu OS အပေါ်မှာ Development Environment တစ်ခုကို သယ်လို တည်ဆောက်သွားမယ်ဆိုတာ ကျန်တော် အတွေ့အကြံ အပေါ်မှာ မှတ်ညီဖြီး ရေးပြသွားမှာ ဖြစ်ပါတယ်။

Dual Boot or Single Boot

ကျွန်ုတ်တို့အနေနဲ့ ပုံမှန် အသုံးပြုနေခြင် OS က Windows ဖြစ်နေတယ် ဆိုရင် Dual Boot အနေနဲ့ သုံးမလား၊ Single Boot အနေနဲ့ သုံးမလားဆိုတာ စဉ်းစားဖို့ လိပါတယ်။ Single Boot ဆိုတာ လုံးဝ Windows ကို မသုံးတော့ပဲ Ubuntu ကိုပဲ သုံးမလား၊ ဒါမူမဟုတ် Windows လဲသုံး Ubuntu လဲ သုံးတဲ့ အနေအထား Dual Boot အနေနဲ့ သွားမလား ဆိုတာ ပထမဗီးဆုံး စဉ်းစားဖို့ လိပါတယ်။

Single Boot အနေဖြင့် သုံးမယ်ဆိုရင်တော့ လွယ်ပါတယ်။ Ubuntu ကို Clean Install လုပ်လိုက်တာနဲ့ ရပါတယ်။ Dual Boot အနေနဲ့ သုံးမယ်ဆိုရင်တော့ Windows ကို အရင်တင် ပြီးမှ Ubuntu ကို တင်သင့်ပါတယ်။ အဲဒီလိုမှုမဟုတ်ပဲ Ubuntu ကို အရင်တင်ပြီး Windows ကို တင်မယ်ဆိုရင် Dual Boot ဖြစ်အောင် လုပ်ရတာ ဖို့ပြီး လက်ဝင်ပါတယ်။

Development Environment Vs Playing Ground

ဒါကတော့ တကယ့်လက်တွေမှာ တွေ့ကြိုရတဲ့ အနေအထား အပေါ်မှာ ဖြစ်လေ့ဖြစ်ထ ရှိတဲ့ အလေ့အထတွေပါ။ Ubuntu ကို Development Environment အတွက် သုံးမယ်လို ရည်ရွယ်ပြီး တည်ဆောက်တယ် ဆိုပေမယ့် တကယ့်လက်တွေ့ အသုံးပြုတဲ့ အခါ Playing Ground တစ်ခုလို ဟိုဟာ စမ်းထည့် ခိုးထည့်လုပ်ရင် ကိုယ်က Linux OS တွေ အကြောင်း သေသေချာချာ မသိတော့ ပျက်သွားလိုက်၊ ပြန် Install လုပ်လိုက်နဲ့ သံသရာ လည်တတိပါတယ်။ အထူးသဖြင့် Ubuntu အနေနဲ့ Effect လန်းလန်းတွေ Apple လို ဒီဇိုင်းတွေ စသည်ဖြင့် အများကြီးရှိတဲ့ အတွက် အဲဒါတွေကို စမ်းထည့်ကြည့်ကြပါတယ်။ အဲဒီလို စမ်းထည့်ကြည့်ရင်း ကိုယ့်စက်က Driver တွေနဲ့ မကိုက်တော့ပဲ ပျက်သွားတဲ့ အခါ ဘယ်ကနေ စလုပ်ရမှန်း၊ ပြန်ပြင်ရမှန်း မသိတော့ပဲ အစကာနေ ပြန်သွေးရတဲ့ သံသရာ လည်စေပါတယ်။

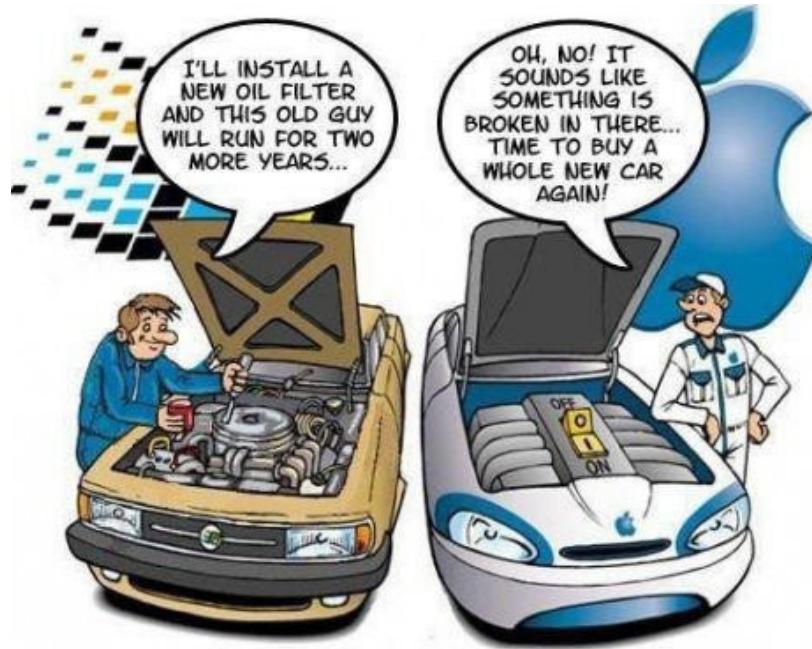
နောက်ထပ် အရေးကြီးတဲ့ အချက်တစ်ခုကတော့ Linux OS ကို Windows OS လို သဘောထားပြီး သုံးမိတာပါ။ ကိုယ်ရေးထားတဲ့ Script က Permission ပြသုနာကြောင့် သုံးမရဘူး။ အဲဒီတော့ ကြားဖူး နားဝရှိတဲ့ chmod -R 777 ... ဆိုပြီး ပေးလိုက်တာတွေ၊ Windows ကို သဘောထားပြီး User Permission ကို မသိပဲ သုံးမိတာတွေကြောင့် သုံးရတာ စိတ်ပျက် လာတတ်ပါတယ်။

အဲဒီအပြင် Windows မှာ WYSIWYG ပုံစံ Point and Click တွေ အသုံးများလာတဲ့ အတွက် Command Line Interface ကို မရင်းနှီးတော့ပဲ Linux OS ကို အသုံးပြုတဲ့ အခါမှာ အခက်အခဲတွေ ရှိလာပြန်ပါတယ်။

ကျွန်ုတ် သင်တန်းမှာ Web Professional Advanced အတန်းအတွက် Ubuntu ကို မသုံးမနေရ သတ်မှတ်ပြီး သုံးခိုင်းတဲ့ အခါမှာ ပြသုနာ ပေါင်းစုတက်လာပြီး အကြိမ်ပေါင်း မောက်များစွာ ပြန်တင်ရတဲ့ သူတွေ ရှိပါတယ်။ ဘာကြောင့် အဲဒီလို ဖြစ်ရသလဲဆိုတဲ့ အချက်တွေကို လေ့လာကြည့်မယ်ဆိုရင်

- Linux OS ကို Development Environment အတွက်ပဲ သုံးတာ မဟုတ်ပဲ၊ လုချင်တာ၊ ဟိုဟာ ခိုးထည့်ပါတယ်။ လျောက်ထည့်ချင်တာ၊
- Windows သုံးသလို သွားသုံးတာ
- Hardware Driver မသိတဲ့ ပြသုနာ (အဲဒီ အပိုင်းမှာလ လူတိုင်းအတော်လေးကို ပြသုနာ တက်ပါတယ်။ ကိုယ့် Hardware မှာ အမိကကျေတဲ့ အစိတ်အပိုင်းတွေဖြစ်တဲ့ Wireless Adapter Driver, LAN Driver မရတာ စတာတွေပါ ကြံတွေရနိုင်ပါတယ်။)
- Command Line Interface ကို မရင်းနှီးတာ
- File Permission, User Permission စတာတွေကို မရင်းနှီးတာ

ဒါပေမယ့် အဲဒီလို ဖြစ်လာရတဲ့ အနေအထားတွေ အတွက် အကောင်းဆုံး ကတွန်းတစ်ခုကို ကြည့်လိုက်မယ်ဆိုရင် ပိုမြို့
သိသာ မြင်သာရှိသွားမယ် ထင်ပါတယ်။



Linux™



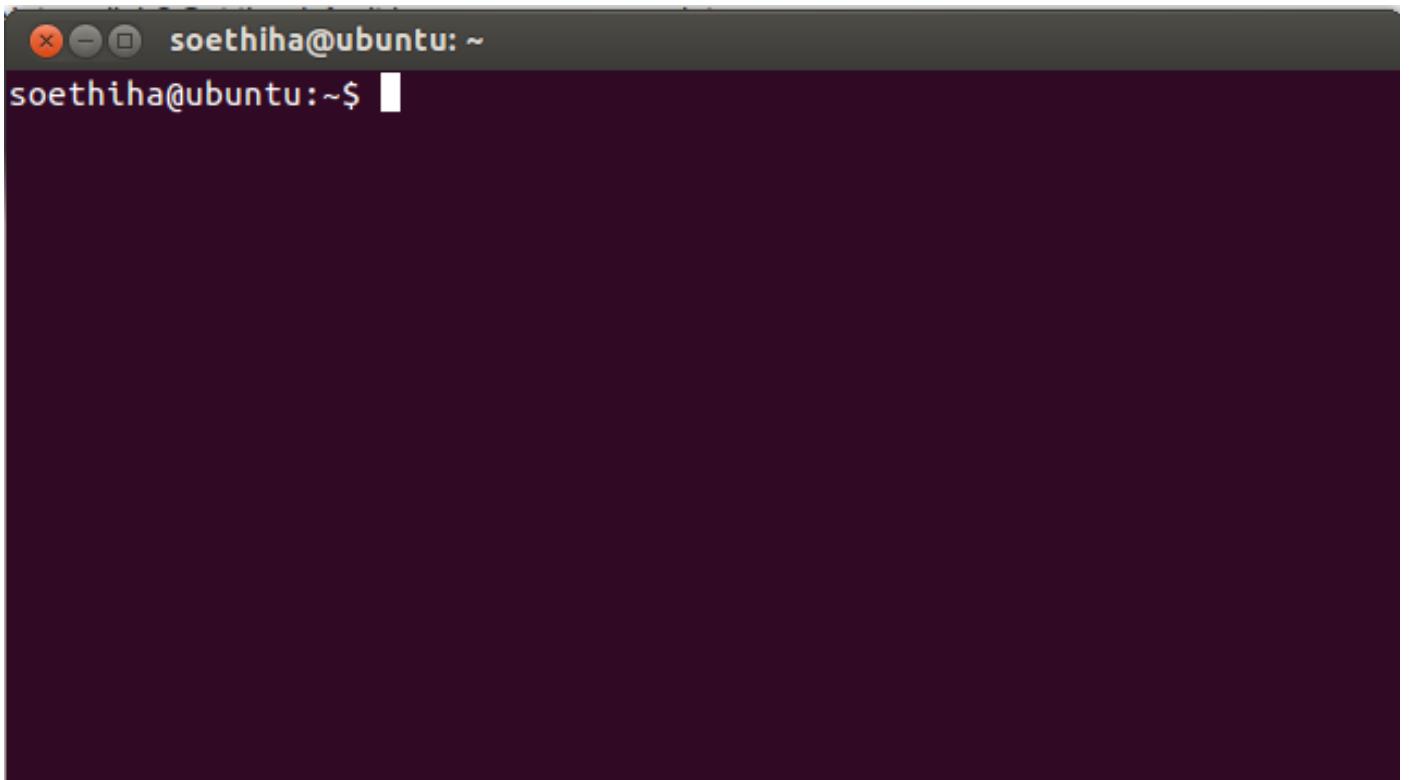
Linux ကို သုံးတယ်ဆိုတာ တစ်စစ် ဖြစ်နေတာကို ပြန်ဆက်ပြီး တည်ဆောက်ရတဲ့ ပုံစံနဲ့ တူပါတယ်။ ဒါပေမယ့် Ubuntu ကတော့ အဲဒီလို တစ်စစ် မဟုတ်ပဲ အတိုင်းအတာတစ်ခုအထိ ပြည့်စုံအောင် စီစဉ်ပေးထားပါတယ်။

First Step toward Development Environment

ကိုယ့်စက်ထဲမှာ Ubuntu သွင်းပြီးပြီးရင် ပထမဗျားဆုံး အနေနဲ့ ကိုယ့်စက်ကို OS Version ရဲ့ နောက်ဆုံး အဆင့်ကို ရောက်အောင် အဆင့်မြှင့်တင်ထားဖို့ လိုပါတယ်။ အဲဒီအတွက် Command Line Interface ကို သုံးဖို့လိုပါတယ်။

CTRL + ALT + T

အဲဒီ Short-cut Key ကို နိုင်လိုက်မယ်ဆိုရင် Terminal Windows ထွက်လာပါလိမ့်မယ်။



OS Updating and Upgrading

OS Updating နဲ့ OS Upgrading ကတေသာ ပထမဌီးဆုံး သုံးမယ်ဆိုရင် လုပ်ပေးဖို့ လိုပါလိမ့်မယ်။ ဒါပေမယ့် ပြဿနာရှိတာက ပထမဌီးဆုံး တင်တာဆိုရင် Update & Upgrade လုပ်တဲ့ အခါ အွန်လိုင်းလိုမှာ ဖြစ်သလို အချိန်အတော်ကြာကြာ ပေးပြီး လုပ်ဖို့ လိုပါတယ်။ နောက်ပြီး အင်တာနှင်း ကော်နှင်းရှင်ကောင်းမှုသာ မြန်မြန်ဆန်ဆန် ပြီးနိုင်ပါတယ်။ အဲဒီလို ၁၁ OS ကို Update & Upgrade လုပ်ဖို့အတွက်

```
sudo apt-get update
```

အဲဒီလို ရိုက်လိုက်ပြီး enter ခေါက်လိုက်မယ်ဆိုရင် OS မှာ သွင်းထားတဲ့ Application တွေရဲ့ Source တွေကို နောက်ဆုံး အခြေအနေဖြစ်အောင် လုပ်ပေးပါလိမ့်မယ်။ update ဆိုတာ source တွေနဲ့ ချိတ်ဆက်ပေးတာဖြစ်ပါတယ်။ အဲဒီလို update လုပ်ပေးတာနဲ့တင် ကျွန်ုတ်တို့ Development Environment ကို တည်ဆောက်လို့ ရပါပြီ။ ဒါပေမယ့် အင်တာနှင်းလဲ ကောင်းတယ်၊ အချိန်လဲ ပေးနိုင်တယ်ဆိုရင်တော့

```
sudo apt-get upgrade
```

ဆုံးပြီး ရိုက်လိုက်ပါ။ Upgrade လုပ်ဖို့လိုတဲ့ application တွေ အကုန်လုံး Upgrade လုပ်ပေးပါလိမ့်မယ်။

ဒီနေရာမှာ တစ်ခု သတိထားဖို့ လိုတာက အမြဲတမ်း Update & Upgrade လုပ်ဖို့ မလိုဘူးဆိုတာပါ။ ကိုယ့် Development Environment က အတိုင်းအတာ တစ်ခုအထိ ပြည့်စုံ သွားပြီဆိုရင် ကော်နှင်းရှင် မကောင်းတဲ့ အခါ အွန်လိုင်းပြီ မကောင်းတဲ့ အနေနဲ့ ပုံမှန် တစ်ပတ်တစ်ခါလောက် Update & Upgrade လုပ်ပေးသင့်ပါတယ်။

Setting up Host Name

Development Environment တစ်ခု တည်ဆောက်တဲ့ နေရာမှာ ပထမဌီးဆုံး လုပ်သင့်တဲ့ အပိုင်းကတေသာ System's Hostname တစ်ခု သတ်မှတ်တာပါ။ အဲဒီလို မသတ်မှတ်ထားရင် install လုပ်ခါစက နာမည်အတိုင်း ဖြစ်နေပါလိမ့်မယ်။ System's Hostname ဆိုတာ Domain Name တို့နဲ့ ဘာမှုမဆိုင်ပါဘူး။ လူတစ်ယောက်မှာလဲ ကိုယ့်နာမည် ရှိသလို ကွန်ပူးတာမှာလဲ နာမည် ရှိသင့်ပါတယ်။ အမှန်တကယ် အွန်လိုင်းမှာ တင်မယ်ဆိုရင်တော့ System's Hostname ကိုရေး FQDN လို့ခေါ်တဲ့ Fully Qualified Domain Name တစ်ခုကိုပါ သတ်မှတ်ဖို့ လိုပါလိမ့်မယ်။ System's Hostname ကို အမည်ပေးတဲ့ နေရာမှာ မိမိစိတ်ကြိုက် နာမည်ပေးလို့ရပါတယ်။ ဒါပေမယ့် www လို နာမည်မျိုးတော့ မပေးသင့်ပါဘူး။

ကျွန်ုတ်တို့ ဖွင့်ထားတဲ့ Terminal မှာပဲ ဆက်ပြီး ရိုက်ရမှာ ဖြစ်ပါတယ်။

```
echo "gotekyar" > /etc/hostname  
hostname -F /etc/hostname
```

အဲဒီလို ရိုက်လိုက်မယ်ဆိုရင် System's Hostname ကို gotekyar လို့ သတ်မှတ်ပေးလိုက်တာ ဖြစ်ပါတယ်။ (ဒီအပိုင်းမှာ တစ်ခု သတိထားဖို့ လိုတာက root access ရှိစွဲ လိုတာပါ။ အဲဒီအတွက်

```
su root
```

ဆိုပြီး အေရင် ရိုက်ထည့်လိုက်ပါ။ ဒါပေမယ့် Ubuntu မှာ ပထားနိုင်းဆုံး Install လုပ်တဲ့အခါ root user ကို disable လုပ်ထားလေ့ရှုပါတယ်။ အဲဒီအတွက် root user ကို ပြန်သတ်မှတ်ပေးဖို့ လိုနိုင်ပါတယ်။

```
su passwd root
```

အဲဒီလို ရိုက်ထည့်လိုက်မယ် ဆိုရင် ပထားနိုင်းဆုံး အခုလက်ရှိ user ရဲ့ password ကို အေရင်တောင်းပါလိမ့်မယ်။ ပြီးရင် root user ရဲ့ Password ကို နှစ်ခါ ထပ်တောင်းပါလိမ့်မယ်။ အဲဒီမှာ ကိုယ်ကြိုက်တဲ့ password တစ်ခုကို သတ်မှတ်ပေးနိုင်ပါတယ်၊ အဲဒီလို သတ်မှတ်ပေးပြီး սူ ဆိုပြီး ပြန်ခေါ်လိုက်မယ်ဆိုရင် root password ကို တောင်းပါလိမ့်မယ်၊ ပြီးရင်တော့ အပေါ်မှာ ပြောထားတဲ့ hostname တွေကို ချိန်းလို့ ရပြီ ဖြစ်ပါတယ်)

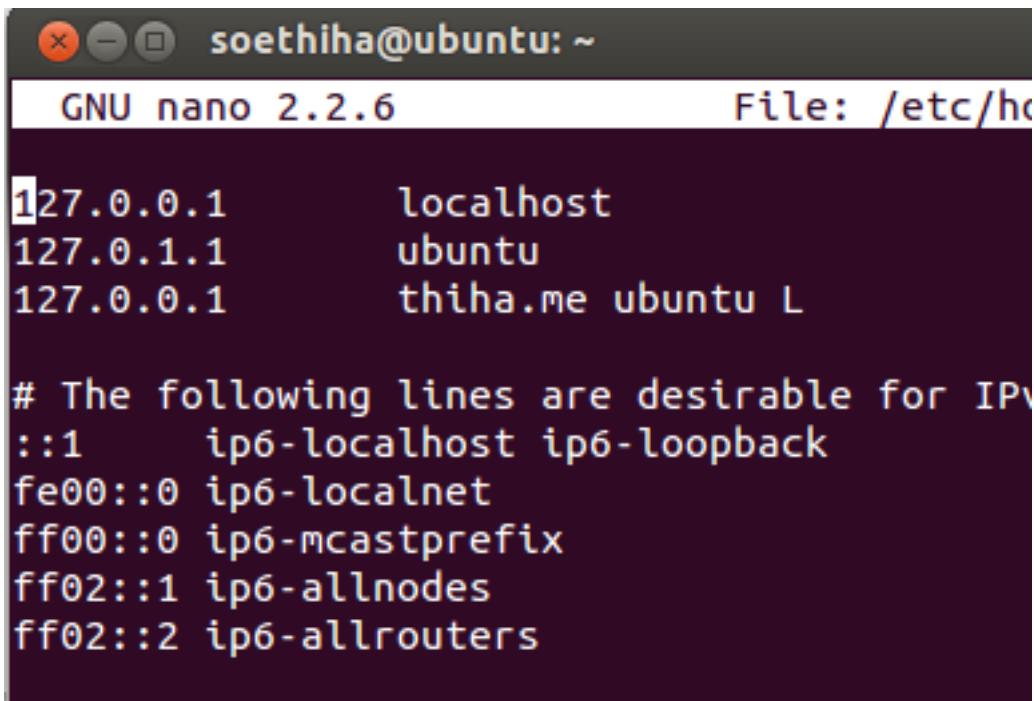
Setting up Local Domain

ကျွန်ုတ်တို့အနေနဲ့ အွန်လိုင်းမှာဆိုရင် တကယ့် Domain တစ်ခုနဲ့ ချိတ်ဆက်ရမှာ ဖြစ်ပါတယ်။ ကိုယ့် သုံးမယ့် Development Environment မှာတော့ တကယ့် ဖို့မိန်းမဟုတ်ပဲ မိမိစိတ်ကြိုက် ရွှေးချယ်ထားတဲ့ domain တစ်ခုကို သတ်မှတ်ပေးနိုင်ပါတယ်။ အဲဒီကို Virtual Host လို့ ခေါ်ပြီး <http://localhost/> လို့ ခေါ်ရမယ့် နေရာမှာ <http://myname.dev> ဆိုပြီး ပြောင်းလဲ အသုံးပြန်ပါတယ်။ Virtual Host နဲ့ သုံးတယ်ဆိုတာ အလွန်ကောင်းတဲ့ အလေ့အကျင့် တစ်ခုဖြစ်ပြီး မိမိစိတ်ကြိုက် သတ်မှတ်လိုတဲ့ ဖို့မိန်းတွေကို Web Host Folder နဲ့ တိုက်ရှိက်ချိတ် သုံးနိုင်ပါတယ်။ ဥပမာ -

web host folder ထဲမှာ ရှိတဲ့ test/test/testdomain ဆိုပြီး ဆောက်ထားမယ်ဆိုရင် localhost က ခေါ်တဲ့အခါ <http://localhost/test/test/testdomain> ဆိုပြီး ခေါ်ရပါမယ်။ အဲဒီကို Virtual Host ဆောက်ထားပြီး test/test/testdomain ကို တိုက်ရှိက် point လုပ်ထားနိုင်ပါတယ်။ အဲဒီလုပ်ထားမယ် ဆိုရင် localhost မှာင့် မရှုပ်တော့ပဲ <http://myvirtualhost.dev> ဆိုပြီး နောက်မှာ folder name တွေ မပါပဲ ခေါ်လို့ ရပါလိမ့်မယ်။

```
sudo nano /etc/hosts
```

အဲဒီလို ရိုက်လိုက်တယ် ဆိုရင် အောက်မှာ ပြထားတဲ့အတိုင်း ပေါ်လာပါလိမ့်မယ်။ nano ဆိုတာ text editor တစ်ခုဖြစ်ပါတယ်။ လိုအပ်တာတွေ ဖြည့်ပြီး ပြန်ထွက်ချင်တယ်ဆိုရင် Ctrl + X ဆိုပြီးထွက်နိုင်ပါတယ်။



```
soethiha@ubuntu: ~
GNU nano 2.2.6
File: /etc/hosts

127.0.0.1      localhost
127.0.1.1      ubuntu
127.0.0.1      thiha.me ubuntu L

# The following lines are desirable for IPv6 capable hosts
::1      ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

ကျွန်တော်ကတော့ ကျွန်တော့ local domain ကို thiha.me လို့ အမည်ပေးထားပါတယ်။ အဲဒီနေရာမှာ မိမိစတုကြိုက် အမည် ပေးနိုင်ပါတယ်။ ဒါပေမယ့် တစ်ခု သတိထားဖို့ လိုတာက google.com လို့ ပေးလိုက်တယ်ဆိုရင် 127.0.0.1 လို့ခေါ်တဲ့ Loopback IP မှာ ချိတ်ထားတဲ့ အတွက် အပြင်ဖက်က google.com ကို ခေါ်လို့ ရတော့မှာ မဟုတ်ပါဘူး။ ဒါကြောင့် အားလုံးလဲလွှတ်အောက် thiha.dev thiha.local စသည်ဖြင့် ပေးနိုင်ပါတယ်။

```
127.0.0.1      thiha.me ubuntu L
```

အဲဒီလို ရေးထားတာကို လေ့လာကြည့်မယ်ဆိုရင် 127.0.0.1 ကတော့ IP Address ပါ။ အမှန်တကယ် အွန်လိုင်းမှာ ဆိုရင်တော့ တကယ့် IP ကို ထည့်ရမှာ ဖြစ်ပါတယ်။ thiha.me ကတော့ local domain ရဲ့ အမည်ပါ။ ubuntu ကတော့ System's Hostname ဖြစ်ပါတယ်။ ကျွန်တော်က ubuntu လို့ ပေါ်ရင် ubuntu ဖြစ်ပြီး gotekyar လိုပေးခဲ့ရင် gotekyar လို့ ရေးရမှာ ဖြစ်ပါတယ်။

ဒါကတော့ local domain သတ်မှတ်တာပဲ ရှိပါသေးတယ်။

LAMP Installation

LAMP ဆိတာကတော့ Linux, Apache, MySQL and PHP ဆိတာကို အတိကောက် ပေးထားတာ ဖြစ်ပါတယ်။ LAMP ကို တစ်ခြား Alternative အနေနဲ့ XAMPP for Linux ဆိတာနဲ့ သုံးနိုင်ပါတယ်။ ဒါပေမယ့် တကယ်တန်း သုံးသင့်တဲ့ပုံစံက Native Installation ပုံစံနဲ့ သုံးသင့်ပါတယ်။ အဲဒီမှာ Update လဲ အလွယ်တကူ လုပ်လို့ရမှာ ဖြစ်ပါတယ်။ **အခု ရှုံးဆက်သွားမယ့် Installation Step** တွေ အားလုံးဟာ အင်တာနက်ကနေ တိက်ရှိက် installation လုပ်မှာ ဖြစ်တဲ့အတွက် အင်တာနက်ကွန်နက်ရှင် ရှိနေဖို့ လိုပါတယ်။

Apache Installation

ပထမဦးဆုံး Apache Web Server ကို install လုပ်မယ်ဆိုရင်

```
sudo apt-get install apache2
```

ဆိုပြီး ရိုက်ထည့်လိုက်မယ် ဆိုရင် apache2 ကို install လုပ်သွားပါလိမ့်မယ်။ အဲဒီလို Apache2 ကို install Ubuntu Version အလိုက် ကွဲပြားရှိနိုင်ပါတယ်။ Ubuntu 13.04 မှာ ဆိုရင်တော့ Apache 2.2.22 Version ဖြစ်ပါလိမ့်မယ်။ အဲဒီလို Apache Web Server ကို install လုပ်ပြီးပြီဆိုရင်တော့ Virtual Host ထောက်လို့ ရပါပြီ။

Configuring Name-based Virtual Hosts

ကျွန်တော်တို့ အပေါ်မှာ domain တစ်ခု သတ်မှတ်ခဲ့တာ မှတ်မိပါလိမ့်မယ်။ ကျွန်တော်ကတော့ thiha.me ဆိုပြီး အမည်ပေးထားပါတယ်။

အခုရှုံးဆက်ပြီး ရေးမယ့် Terminal Command တွေအားလုံးကို root user အနေနဲ့ မလုပ်သင့်ပဲ original user အနေနဲ့ပဲ သုံးသင့်ပါတယ်။

```
sudo a2dissite default
```

အပေါ်မှာ ပြထားတဲ့ command ကတော့ Apache Server က သွင်းထားတဲ့ default point ကို ဖျက်လိုက်တာပါ။ ပုံမှန်အတိုင်းဆိုရင် /var/www ဆိုတဲ့ folder ကို Point ထောက်ထားလေ့ ရှိပါတယ်။ အဲဒီလို ဖျက်လိုက်တယ်ဆိုတာ ကိုယ့် user access ရှိတဲ့ နေရာမှာ web server ကို point ထောက်ဖို့အတွက် ဖြစ်ပါတယ်။

```
cd ~
```

အဲဒီလို ရိုက်လိုက်မယ်ဆိုရင် အချက်ရှိပါတယ်။ ဥပမာ -
ကျွန်ုတ်သော့ username က soethiha ဆိုရင် /home/soethiha ဆိုတဲ့ folder ထဲကို ရောက်သွားပါတယ်။ ကိုယ့် folder ဖြစ်တဲ့အတွက်
Read/Write/Execute ကြိုက်တာ လုပ်လို့ရပါတယ်။

```
cd Public
```

အဲဒီလို ရိုက်လိုမယ်ဆိုရင် Public ဆိုတဲ့ folder ထဲကို ရောက်သွားပါလိမ့်မယ်။ ပြီးရင်

```
mkdir -p thiha.me/{public, log, backup}
```

အဲဒီလို ရိုက်လိုက်တယ်ဆိုရင် thiha.me ဆိုတဲ့ folder တစ်ခုရယ် folder အောက်မှာ public, log နဲ့ backup ဆိုတဲ့ folder
တွေ တည်ဆောက်သွားပါလိမ့်မယ်။

```
sudo chmod a+rwx ~
```

အဲဒီလို ရေးလိုက်မယ်ဆိုရင် ကိုယ့်ရှဲ ကိုယ်ပိုင် folder ကို user အားလုံးက read လုပ်ခွင့်ရှိသလို access လုပ်ခွင့်လဲ
ရှိသွားမှာ ဖြစ်ပါတယ်။

```
sudo chmod -R a+rwx ~/Public
```

ဒီ command ကတော့ ရှိသမျှ user အားလုံး ~/Public ဆိုတဲ့ folder ကို read/access လုပ်ခွင့် သတ်မှတ်ပေးလိုက်တာပါ။
အဲဒီလို သတ်မှတ်ပေးမှသာ Apache Server အနေနဲ့ Point ထောက်လို့ ရမှာ ဖြစ်ပါတယ်။ Apache Server အနေနဲ့ သီးသန့် account
တစ်ခု အနေနဲ့ Linux Folder တွေကို access လုပ်ပါတယ်။ Ubuntu မှာ ဆိုရင်တော့ www-data ဆိုတဲ့ account name နဲ့ အသုံးပြုတာ
ဖြစ်ပါတယ်။

အခုမှ Virtual Host ထောက်တဲ့ အပိုင်းကို ရောက်မှာ ဖြစ်ပါတယ်။

```
sudo nano /etc/apache2/sites-available/example.com
```

example.com ဆိတဲ့ နေရာမှာ မိမိပေးလိုတဲ့ domain နာမည်ပေးလို ရပါတယ်။ ကျွန်တော်ကတော့ thiha.me လိုပေးထားတော့ site-available/thiha.me ပြော၍ ပြီးရင် example.com ဆိတဲ့ နေရာမှာ Local Domain Section မှတုန်းက ပေးထားတဲ့ အမည်ကို အစားထိုးပြီး ကျွန်တော် ရေးထားတဲ့အတိုင်း ရိုက်ထည့်လိုက်ပါ။

```
# domain: example.com
# public: /home/example_user/Public/example.com/

<VirtualHost *:80>
    # Admin email, Server Name (domain name), and any aliases
    ServerAdmin webmaster@example.com
    ServerName www.example.com
    ServerAlias example.com

    # Index file and Document Root (where the public files are
    # located)
    DirectoryIndex index.html index.php
    DocumentRoot /home/example_user/Public/example.com/public

    # Log file locations
    LogLevel warn
    ErrorLog /home/example_user/Public/example.com/log/error.log
    CustomLog /home/example_user/Public/example.com/log/access.log
    combined
</VirtualHost>
```

ဒီနေရာမှာ example.com ဆိတာကို etc/hosts မှာ ပေးထားတဲ့ နာမည်နဲ့ အစားထိုးဖို့ မမေ့ပါနဲ့။ example_user ဆိတဲ့ နေရာမှာလဲ ကိုယ့် အခုလက်ရှိသုံးနေတဲ့ user name ကို ပြောင်းဖို့ မမေ့ပါနဲ့။

အဲဒီလို ရိုက်ထည့်ပြီးပြီဆိုရင် Ctrl + X ကို နိပ်ပါ။ save မလားလို မေးရင် သူ့နာမည်အတိုင်း ထားပြီး save လိုက်ပါ။

```
sudo a2ensite example.com
```

အပေါ်မှာ ပြထားတဲ့အတိုင်း a2ensite ဆိုပြီး ရိုက်လိုက်မယ်ဆိုရင် ကိုယ့် local domain ကို enable ပေးပါလိမ့်မယ်။

```
sudo service apache2 restart
```

အပေါ်ကအတိုင်း ရှိက်ထည့်လိုက်မယ် ဆိုရင်တော့ apache ကို restart ချွှေး ကိုယ့်၏ local domain နဲ့ Virtual Host သတ်မှတ်လို့ ပြီးပြီ ဖြစ်ပါတယ်။

MySQL Installation

```
sudo apt-get install mysql mysql-server
```

အဲဒီ command ကို ရှိက်လိုက်မယ်ဆိုရင် MySQL Server ကို Install လုပ်သွားပါလိမ့်မယ်။ အဲဒီလို့ Install လုပ်နေရင်း root user ရဲ့ password ကို တောင်းတဲ့အခါ ထည့်ပေးလိုက်ပါ။ အွန်လိုင်းက Server မှာ MySQL ကို install လုပ်မယ်ဆိုရင်တော့

```
sudo mysql_secure_installation
```

ဆိုတာကို Run ပေးဖို့ လိုပါလိမ့်မယ်။ အဲဒီလို့ run လိုက်မယ်ဆိုရင်

- anonymous user accounts တွေကို ဖယ်ရှားပေးပါလိမ့်မယ်။
- remote root login ကို disable လုပ်ပေးပါလိမ့်မယ်။
- test database ကို ဖယ်ရှားရှင်းလင်းပေးပါလိမ့်မယ်။

PHP Installation

ရှုံးဆက်ပြီး Install လုပ်ရမယ့် အပိုင်းက PHP ပါ။

```
sudo apt-get install php5 php-pear php-mysql
```

အဲဒီလို့ ရှိက်လိုက်မယ်ဆိုရင် PHP ကို install လုပ်သွားပါလိမ့်မယ်။ အဲဒီလို့ php install လုပ်ပြီး သွားပြီဆိုရင်တော့ PHP အလုပ်လုပ်ကြောင်း သိနိုင်စိုးအတွက်

ကျွန်ုတ်တို့ local domain က point ထောက်ထားတဲ့

/home/example_user/Public/example.com/public

ဆိုတဲ့ folder ထဲကို သွားပြီး index.php ဆိုတဲ့ file တစ်ခု တည်ဆောက်လိုက်ပါ။ ပြီးရင် အောက်မှာ ပြထားတဲ့အတိုင်း ရေးထည့်လိုက်ပါ။

```
<?php  
phpinfo();  
?>
```

အဲဒီလိုဂိုက်ထည့်ပြီး browser မှာ example.com လိုဂိုက်ထည့်လိုက်မယ်ဆိုရင် php နဲ့ပတ်သက်တဲ့ အချက်အလက်တွေကို ပြပါလိမ့်မယ်။ အဲဒီဆိုရင်တော့ PHP, Apache, MySQL installation step တွေ အားလုံးပြီးပြီ ဖြစ်ပါတယ်။

phpMyAdmin Installation

ကျွန်တော်တို့အနေနဲ့ MySQL နဲ့ ပတ်သက်တာတွေကို ကိုင်တွယ် ဆောင်ရွက်ချင်တယ်ဆိုရင် command line interface ကနေ ဆောင်ရွက်လို့ ရနိုင်ပါတယ်။ အဲဒီအပြင် browser ကနေ တစ်ဆင့်လ ဆောင်ရွက်လို့ ရနိုင်ပါတယ်။ အဲဒီအတွက် phpmyadmin ဆိုတာကို install လုပ်ပေးဖို့ လိုပါလိမ့်မယ်။

```
sudo apt-get install phpmyadmin
```

အဲဒီလို install လုပ်နေတဲ့ အချိန်မှာ လိုအပ်တဲ့ password တွေ account information တွေ ထောင်းပါလိမ့်မယ်။ အဲဒီတွေကို ဖြည့်ပေးလိုက်မယ်ဆိုရင် phpmyadmin ကို install လုပ်သွားပါလိမ့်မယ်။

ရှေ့ဆက်ပြီး Coding Environment, Browser Environment တည်ဆောက်တဲ့ အပိုင်းတွေ ဆက်ရေးလီးမှာ ဖြစ်ပါတယ်။

■ Interview with Ko Ko Ye

by Thiha



about
THIHA

Founder of Myanmar Tutorials and Myanmar Links

soethiha.me

[/soethihanaung](https://www.facebook.com/soethihanaung)

[@phpcrazy](https://twitter.com/phpcrazy)

[/phpcrazy](https://www.instagram.com/phpcrazy)

ကိုကိုရဲအနေနဲ့ Ubuntu ကို လူတွေ သုံးလာအောင် သိလာအောင် နေရာစံသွားပြီး ဟောပြောပွဲတွေ လုပ်၊ Installation တွေလုပ် စသည်ဖြင့် ပုံစံမျိုးစုံ လုပ်ကိုင်ဆောင်ရွက်နေတာ ကျွန်တော်တို့ သိထားတယ်ပျ။ ကျွန်တော် သင်တန်းမှာ Advanced အတန်းသားတိုင် Ubuntu သုံးရမယ်လို့ သတ်မှတ်လိုက်တော့ တက်လိုက်တဲ့ ပြဿနာများ၊ ဟိုဟာမရဘူး၊ ဒီဟာ မရဘူး၊ ပျက်သွားပြန်ပြီ စသည်ဖြင့် ပြဿနာမျိုးစုံတက်တယ်ပျ။ အတော်လေးကို စိတ်ရည်ရည်ထားပြီး ဖြေရှင်း လုပ်ဆောင်ရတဲ့ ကိစ္စတွေပဲပျ။ အဲဒီတော့ ကိုကိုရဲအနေနဲ့ ဘယ်လို့ အခက်အခဲတွေ ကြံ့တွေ့ရတယ်။ ဘယ်လို့ ကူညီ ဆောင်ရွက်ပေးခဲ့ရတယ်ဆိုတာ ပြောပြုပေးပါဉီးပျ။

Ubuntu ကိုသုံးရာမှာ ပြဿနာများတယ်ဆိုတဲ့ ကိစ္စတွေကို စစ်သုံးတဲ့လူတွေဆီကသာမက Ubuntu ကို Development လုပ်နေသူတွေဆီကပါ ပြောသံကြားရပါလိမ့်မယ်။ အဲတော့ ပြဿနာတွေကို ကျွန်တော်တို့ အရင်းစစ်ကြည့်လိုက်ရအောင်

1. Ubuntu ရဲ UI က စိမ်းတယ်။
2. Installation မလုပ်တတ်ဘူး။
3. အသံမထွက်ဘူး။ WiFi မရဘူး။ Graphic မရဘူး။
4. ဆော့ဝဲတွေ မသွင်းတတ်ဘူး။ ဘာတွေ ရှိမှန်း မသိဘူး။
5. Windows ဘက်က ဆော့ဝဲတွေလိုလည်း မဟုတ်ဘူး။

Ubuntu ရဲ မတူညီတဲ့ UI ကြောင့်ဆိုပြီး စပြောကြည့်ကြမယ်။ Ubuntu ဟာ ၂၀၁၀ ခုနှစ်တည်းကာနေ အခုချိန်ထိ ထုတ်လုပ်လာလိုက်တာ Version ပေါင်း ၂၀ လောက် ရှိတော့မယ်။ နောက်လ October ရောက်ရင် အသစ်တစ်ခုထွက်မယ်။ နောက်ထပ် ၆ လနေပြီး April ရောက်ရင်လည်း အသစ်တစ်ခု ထပ်ထွက်လိုးမှာပဲ။ အဲဒီအခုချိန်ကျရင်လည်း LTS ဆိုတာလေး ဘေးမှာ ပါလာဦးမယ်။ ဘာလို တရာ့နဲ့ တရာ့မတူကြတာလဲ သုံးရတာ အဆင်မပြောရတာလဲ။

Windows မှာဆိုရင် UI တွေက ၉၅ က စပြီး ပြောင်းလဲလာပေမယ့် ၉၈ လဲ ဒီပုံစံဆင်ဆင်လေးပဲ။ ၂၀၀၃ လဲ ဒီပုံစံလေးပဲ။ Me လဲ ဒီပုံစံ XP လဲ ဒီပုံစံပဲ။ နောက်ထပ်လာတဲ့ Vista နဲ့ ၇ ဆိုလည်း အပေါ်ယံ theme သာပြောင်းလဲမှာ နည်းနည်း ရှိတာ။ Start Button တွေ Shortcut တွေ file တွေ Folder ထားတာတွေက သိပ်မှ မပြောင်းလဲတာ။ Windows 8 မှာတော့ တော်တော်လေး ပြောင်းလဲတယ်လို့ ပြောလို့ရပေမယ့် အဲလို့ ပြောင်းလိုက်တာ အဆင်မပြောကြလို့ သိပ်မကြာခင်မှာပဲ အနည်းငယ် ပုံစံပြောင်းပြီး ၈.၁ ကို Update ထုတ်ခဲ့ရတယ်။

თავისუფალი Windows 8.1-ის დროინდების გამო, მათ შემდეგ მის გადატენვის შესახებ მიმდინარეობს ერთ კითხვა: როგორ გადატენოთ მათ დანართზე? ეს კითხვა მარტივია, რადგან მათ შემდეგ მათ გადატენვის შესახებ მიმდინარეობს ერთ კითხვა: როგორ გადატენოთ მათ დანართზე?

ကျွန်တော်တို့ ဘယ်နေရာမှာ Event လုပ်လုပ် ပြောဖြစ်တာ မေးဖြစ်တာက အင်လိပ်စာနဲ့ မြန်မာစာ ဘယ်ဟာက ပိုလွှယ် သလေဆိတဲ့ မေးခွန်း ဖြစ်ပါတယ်။ လွယ်လွယ်နဲ့ မြန်မြန် ဖြေရင်တော့ မြန်မာစာလို့ ဖြေကြပါတယ်။ ထပ်မေးစရာရှိတာက ဘယ်လိုဂူတွေအတွက် လွယ်တာလဲ ဘာကြောင့် လွယ်တယ်လို့ ပြောတာလဲ။ မွေးကတည်းက မြန်မာတွေနဲ့ နေတဲ့သူတွေ သို့မဟုတ် မြန်မာစာ မြန်မာစကားကို ငယ်ငယ်တည်းက သင်ကြားခွင့် ရသူတွေအတွက်ပဲ လွယ်တာပါ။ အင်လိပ်စာမှာ ပျဉ်းဘယ်နှစ်လုံးရှိလဲ။ မြန်မာစာရေးရင် ပုံစုစ်ဆိတဲ့ တွေ ဘယ်နှစ်မျိုး ရှိလဲ ရွှေ့ရွှေ့ငင်တဲ့မှာ တစ်ရွှေ့ရွှေ့ငင်နဲ့ နှစ်ရွှေ့ရွှေ့ငင် ပုံစုတွေက အသေလား။ ဘယ်နှစ်မျိုး ရှိလဲ။ အဲတာမျိုး ပျဉ်းတွေ ဒေါက်ပက်မှာ ကပ်တဲ့ ဝဆွဲတွေ ဒေါက်ကမြင့်တွေ ဟထိုး တစ်ရွှေ့ရွှေ့ငင် နှစ်ရွှေ့ရွှေ့ငင် ဘယ်နှစ်ရှိရှိလဲ။ အပေါ်ဘက်မှာ ထားရတဲ့ လုံးကြီးတင် လုံးကြီးတင်ဆံတ် သေးသေးတင် နောက်ပစ် ဘယ်နှစ်ရှိရှိလဲ စသည်ဖြင့် နည်းနည်းလေး ထပ်မေးရင် မဖြေနိုင်တော့ပါဘူး။ ကကြီးကနေ အ အထိ ရေးပြနိုင်လား ဆိုရင်လည်း တော်တော်များများ မရေးပြနိုင်တော့ပါဘူး။ အင်လိပ်စာမှာ ပျဉ်းဘယ်နှစ်မျိုး ပုံစု ဘယ်နှစ်မျိုး ရှိလဲဆိုရင်တော့ အကုန်ဖြေနိုင်ကြပါတယ်။ တကယ်တော့ အင်လိပ်စာက ပိုပြီးလွယ်ပါတယ်။ မြန်မာဘက်၊ အင်လိပ်စာက်၊ မတတ်သေးတဲ့ သူတယောက်ကို နှစ်ခုစုလုံး တပြုပြုတည်းသင်ရင် အင်လိပ်စာက ပိုပြီး အတတ်မြန်ပါလိမ့်မယ်။ ကျွန်တော်တို့က မွေးကတည်းက သုံးလာရတဲ့ ဘာသာစကား ဖြစ်နေတာကြောင့် အမှတ်သာကြောင့် လွယ်တယ်ထင်နေကြရတာပါ။

အခါသဘောပါပဲ။ Ubuntu ကို ၃ ရက်လောက်ပဲ သုံးပြီး UI ကြောင့် အဆင်မပြောဘူးပြောတဲ့အခါမှာ ဖြေရင်းနှိုးနည်းလမ်းပေးနှိုး
ခက်ခဲပါတယ်။ ဘာကြောင့်လဲဆိုတော့ သူတို့ပြောကြတဲ့ အဆင်မပြောဘူးဆိုတာက သူတို့တေသာက်လုံး သုံးလာတဲ့ Windows နဲ့ မတူဘူးဆိုတဲ့
အဓိပ္ပာယ်ရနေလို့ပါပဲ။ အလွယ်ဆုံးကတော့ Windows သမားတွေ အဆင်ပြောအောင် လုပ်ထားတဲ့ UI တွေကို ပြောင်းသုံးတာပဲ ဖြစ်ပြီးတော့
အကောင်းဆုံးကတော့ ၁ လလောက် ဆက်တိုက် အဆင်ပြုမယ့် UI တစ်ခုကို သုံးတာ (လေ့လာတာ) ပါဖြစ်ပါတယ်။

နောက်ထပ် တစ်ခု ဖြောက်ညွှန်ပါပြီးမယ်။ ပုံစံတူ တိုက်ခန်း တိုက်တန်း မှာ နေခဲ့မယ်ဆိုပါစိုး။ တိုက်တစ်စီး ရွှေ့ချုပ်မှာ ၅ လွှာရှိပြီး တလွှာမှာ ၁၀ ခန်း ရှိပါတယ်။ ကန်ထရှိက် တိုက်တန်း တစ်ခုအနေနဲ့ အခန်းထဲမှာ အိမ်သာ ရေချိုးခန်း အပိုင်ခန်း ညည့်ခန်း မီးဖိုချောင်တွေကို ပုံစံတူပဲ ထားပေးပါလိမ့်မယ်။ တိုက်တစ်စီး ရွှေ့ချုပ်မှာ အခန်းပေါင်း ၅၀ ရှိပြီး အဲဒီ ရပ်ကွက်မှာ တိုက် အလုံး ၅၀ ဆိုတော့ အခန်းပေါင်း ၅၅၀ လောက်ရှိပါလိမ့်မယ်။ အဲဒီမှာ ကြီးပြင်းလာရတဲ့ ကလေးဟာ ကိုယ့်အိမ်မှာနေတဲ့ အတိုင်း သူငယ်ချင်းတွေ အိမ်မှာ မီးဖွင့်တာ မီးပိတ်တာ အိမ်သာနဲ့ ရေချိုးခန်းကို သွားတာ စတာတွေကို ဖေးစရာ မလိုပဲ လုပ်နိုင်ပါလိမ့်မယ်။ အဲဒီ ရပ်ကွက်ထဲမှာ ဘယ်အိမ်ခန်းပဲ တက်တက် ဒီပုံစံနဲ့ ၃ နှစ်လောက် နေပြီးတော့မှ အဗ္ဗားရဲ့ ကိုယ်ပိုင် အိမ်ကို သွားတဲ့အခါမှာ ရေချိုးခန်း အိမ်သာ နဲ့ မီးလုပ်တွေကို ရှာဖိုး သူ့ရပ်ကွက်ထဲမှာလို မလွှယ်တော့ပါဘူး။ သူဟာ အရင်က တိုက်တန်းကို ဆောက်လုပ်တဲ့ ကန်ထရှိက်တာကို ချိုးမှုများပြီး အဲတာနဲ့

တူအောင် မဆောက်နိုင်တဲ့အတွက် အဖွားအိမ်ကို ဆောက်တဲ့ အင်ဂျင်နီယာကို ကဲ့ရဲ့ကောင်း ကဲ့ရဲ့ပါလိမ့်မယ်။ တကယ်တော့ အဖွားရဲ့ ကိုယ်ပိုင်အိမ်က တဗြားသော အိမ်ယာတွေလို ပုစံတူတွေ အများကြီး ဖြစ်ရင်လည်း ဖြစ်မယ်။ အဖွားကိုယ်တိုင် သူနဲ့ အဆင်ပြေမယ့် ပုစံ ဆောက်လုပ်ခိုင်းထားတာလဲ ဖြစ်ချင်ဖြစ်မယ်။ အဖွားအိမ်မဟုတ်ပဲ ကမ်းခြေက အပန်းဖြေစွဲ ဘန်ဂလိုတစ်ခုကိုပဲ ရောက်ရောက် လုပ်ခြေးအတွက် အဓိကထားပြီး တည်ဆောက်ထားတဲ့ ရဲတိက်ကြီးတစ်ခုကိုရောက်ရောက် ပုစံတဲ့ တိက်တန်းက ပုစံတဲ့ အိမ်ယာတွေက လူတယောက်က ချက်ချင်း အကုန်လုံးကို ရင်းနှီးကျွမ်းဝင်မှု မရှိနိုင်ပါဘူး။ သေချာတာက အဆောက်အအို ပုစံ ညံ့တာမဟုတ်ပါဘူး။ ကိုယ်တွေနဲ့ မရင်းနှီးသေးတာ ဖြစ်ပါလိမ့်မယ်။

Windows 8 မှာ အရင် UI နဲ့ မတူတဲ့ အတွက် အဆင်မပြုမှုတွေ များတယ် ဆိုတဲ့ အသံ ထွက်လာပါတယ်။ အဲဒီမှာ ချက်ချင်း Windows 8.1 ကို အမြန်ထုတ်ပေးရတာပါပဲ။ တကယ်တော့ အရင် Version ဟောင်းတွေနဲ့ တူအောင် ပြန်လုပ်ပေးလိုက်ရတာပါ။

အတာ လူတွေ အပြောင်းအလဲကို မကြိုက်ကြေဘူး ပြောင်းလဲစွဲ အချိန်ပေးရ လေ့လာရမှာ မလိုလားဘူး ဆိုတာကို ပြတာပါပဲ။ အပေါ်မှာ ပြောခဲ့သလိုပဲ အင်္ဂလာပိတ်တက မြန်မာစာထက် လွယ်ပါတယ်လို ဝန်ခံကြုပေါ်မယ့် အင်္ဂလာပိတ်တကို အချိန်ပေးပြီး မလေ့လာချင်ကြေဘူး။ မြန်မာလို ပြောလဲ ရနေတာကိုး။ PageMaker က အသစ်လဲ မထွက်တော့ဘူး Update လဲ မလုပ်တော့ဘူး ပိုကောင်းအောင်လုပ်ထားတဲ့ ပိုဂျွယ်တဲ့ InDesign ဖြစ်သွားပြီဆိုပေါ်မယ့် ဒီမှာ PageMaker ပဲ သုံးကြတယ်။ Shortcut တွေ UI တွေ မတူတဲ့ InDesign ကို မလေ့လာချင်ကြေဘူး။ ထပ်သင်ရမှာ မလိုလားဘူး။ တစ်ခါ့။ DTP ဆိုင်တွေမှာ Windows XP, PageMaker, Photoshop CS2 နဲ့ အလုပ်လုပ်နေတုန်းပါပဲ။ တဗြား အကြောင်းပြုချက်မဟုတ်ပါဘူး အသစ်ကို မလေ့လာချင်တာ အချိန်မပေးချင်တာပဲ။ ယူနိုက်ဖို့ ရှိကိုတာ ကောင်းတယ်မှန်တယ်မြန်တယ်ဆိုပေါ်မယ့် ဇော်ဂျိန္တာပါပဲ။ ကိုယ်အသေမှတ်ထားတဲ့ ကိုယ်အကျွမ်းဝင်တဲ့ နယ်ပယ်ကနေ အသစ်တရာကို ပြောင်းလဲစွဲ လေ့လာဖို့ အတွက် အချိန်မပေးချင်ကြတာပါ။ တစ်ကနေ ပြန်စရမယ်ဆိုတာမျိုး မလိုလားကြတာပါ။ Windows XP ကို ဖက်တွယ်နေရာကနေ Windows 7 တွေ Windows 8 တွေကို မသွားချင်တဲ့ လူတစ်ယောက်ဆိုရင်တော့ Ubuntu Unity UI က သူအတွက် လေ့လာဖို့ အချိန် ထပ်ပေးရမယ် လို မြင်နေနိုင်ပါတယ်။ သေချာတာက တပါလောက် သေချာ အချိန်ပေးလေ့လာပြီးရင် Linux ဘက်နေ ပြန်ထွက်ဖို့ တကယ်ခက်ခဲပါတယ်။။ ဘာကြောင့်လဲဆိုတော့ Linux က စိတ်တိုင်းကျ လုပ်ဆောင်ခွင့် ရှိတဲ့အပြင် Innovative ဖြစ်လွန်းလိုပါပဲ။

Ubuntu မှာ အမြဲ ပြဿနာတွေနဲ့ ရင်ဆိုင်ရတယ်လို ပြောပါတယ်။ အများဆုံးတွေကြတဲ့ ပြဿနာတွေကို ဖြောက်ညွှန်ရအောင်။

Installation လုပ်စွဲ ခက်တယ်။

Windows သုံးတဲ့သူ တော်တော်များများက Windows Installation Setup ကိစ္စကို သိမ့်မလိုပါဘူး။ ဆိုင်မှာ Computer ဝယ်ရင် OS သာမက ကြိုက်တဲ့ ဆော့ဝဲပါ အခမဲ့ ထည့်ပေးပါတယ်။ အိမ်မှာ Computer က ပြဿနာတရာဖြစ်ရင် ဖုန်းဆက်လိုက်ရုံပါပဲ။ ၃၀၀၀ ကနေ ၁၀၀၀၀ လောက်ပေးရုံး အိမ်မှာ လာပြီး Services ပေးနေကြပါတယ်။ Windows ကို ကိုယ်တိုင် Installation လုပ်တဲ့သူတွေဆိုတာလည်း Windows ဘက်မှာ သင်တန်း ဘယ်နှစ်လ တက်ပြီး အသံးပြုတာလဲ ဘယ်လောက် ကြာပြီလဲ ဆိုတာ ပြန်ထွက်ကြည့်ရပါမယ်။ ကဲ Ubuntu ဘက်ကို ကြည့်ရအောင် Ubuntu User တစ်ယောက်က Ubuntu ကိုသုံးချင်ပြီဆိုရင် Ubuntu သုံးတတ်တဲ့သူတွေ လိုက်ရှာရမယ် အခြေ ရှိမရှိမေး Installation လုပ်ပေးဖို့ တောင်းဆိုရင် တောင်းဆို ဒါမှ မဟုတ် ကိုယ်တိုင် စမ်းပြီး Installation လုပ်ရပါတွေ့မယ်။ တော်ရုံလူအတွက်တော့ တော်တော် ရင်ခုန်ဖို့ ကောင်းပါတယ်။ DATA တွေ ကိုယ်သုံးနေတဲ့ Windows OS တွေ ပျက်သွားနိုင်တာကိုး။

လူတွေ သတိမထားမိတာက Ubuntu က Installation လုပ်ပြီးမှ သုံးလို့ရတဲ့ အမျိုး အစား မဟုတ်ပါဘူး။ Installation မလုပ်ပဲနဲ့လဲ USB ကပဲ ဖြစ်ဖြစ် CD ကပဲဖြစ်ဖြစ် သုံးနိုင်ပါတယ်။ အဲအချိန်မှာ Software အကုန်လုံးကို ပုံမှန် Installation လုပ်ထားသလိုပဲ

အသုံးပြုလိုရနေမှပါ။ Official အခွေဆိုရင်တော့ မြန်မာယဉ်နီကုဒ်ဖော်လိုက် ကိုဘုတ်ကို ထပ်သွင်းစရာမလိုပဲ သုံးနိုင်ပါသေးတယ်။ ကျွန်တော်တို့ Customize ပြန်လုပ်ထားတဲ့အခွေမှာဆို ကော်ကို ကော အဗြားဖော်လွှာနဲ့အတူ ကိုဘုတ် အခု ၂၀ ကျော် ၃၀ ပါပါတယ်။ အဲဒီလို အသုံးပြုနေရင်နဲ့လည်း Installation ပြုလုပ်နေလိုရပါတယ်။ အဲလိုရတာ Ubuntu ဘက်မှာပဲ ရှိပါတယ်။ Installation လုပ်ရင်း HDD ကို မှားဖျက်မိတယ် ဆိုတာကလည်း တော်တော် ရှားပါတယ်။ HDD နဲ့ ပက်သက်ပြီး ရွေးချယ်စရာဆိုလည်း Along size လိုပေါ်တဲ့ လက်ရှိ OS ရဲ့ ပိုနေတဲ့နေရာကို ယူပြီး Ubuntu တင်ဖို့ကိုရွေးမယ်ဆိုရင် ဘာမှ ပြဿနာရှိမှာမဟုတ်ပါဘူး။ Replace ကိုရွေးမိရင်တော် လက်ရှိ Windows ရဲ့ နေရာမှာပဲ အစားထုံးမှာ ဖြစ်လို့ D: တွေ E: တွေ စတဲ့ Data နေရာတွေ အဗြား Partition တွေကို မထိခိုက်ပါဘူး။ Clean Install လုပ်မယ်ဆိုရင်တော့ HDD တွေထဲက အကုန်လုံးကို ဖျက်မှာဖြစ်ပြီး something else ကိုရွေးရင်တော့ Linux ရဲ့ Partition တွေနဲ့ ပတ်သက်ပြီး အတွေ့အကြုံထားမှ ငိုအဆင်ပြောလိမ့်မယ်။ လက်ရှိ Partition တွေကိုလည်း ခွဲမြားနိုင်ဖို့ လိုပါတယ်။ အဲဒီတော့ နည်းနည်းလေး လေ့လာရုံနဲ့ ပြီးပြည့်စုံနိုင်တာဖိုးပါ။

Software တွေနဲ့ Driver တွေရဲ့ ပြဿနာတွေ

Ubuntu မှာ MP3 MP4 ဖွင့်မရတာကို ညည်းပြတဲ့ သူတွေရှိပါတယ်။ တရာ့ကတော့ တရာ့၊ DELL နဲ့ HP လို Laptop တွေမှာ Wifi Driver ကို မသိတာမျိုးကို စိတ်ပျက်ကြပါတယ်။ ဒါတွေက ထပ်မံ ထည့်သွင်းဖို့လိုနေပါတယ်။ ဒါပေမယ့် အင်တာနက် ရှိနေမယ်ဆိုရင် Update နဲ့ Upgrade လုပ်ရုံနဲ့ တကယ် အဆင်ပြောနိုင်ပြီး ထုတ်ထားလို့ရပါတယ်။ mp3 တွေကို ဖွင့်ဖို့အတွက်ဆိုရင်လည်း mp3 file ကို ကာလစ်ပြီး wizard လေးနဲ့ install သွားရုံပါပဲ။ Wifi လုပ်လိုပါပဲ။ Ethernet တို့နဲ့ အရင် ချိတ်ပြီး Update Upgrade လုပ်ရုံပါပဲ။

Windows XP မှာလည်း Printer Driver တွေ Wireless Driver တွေ သွင်းခဲ့ရတာပါပဲ။ ပြီးတော့ Windows Vista ကို Upgrade လုပ်ဖို့ မလွယ်ကူခဲ့သလို အသစ်ပြန်တင်ထိုင်းလဲ Driver တွေ ပြန်သွင်းရမယ် ဆိုတာ သေချာပါတယ်။ Ubuntu မှာတော့ 12.04 (2012 April) ကိုထည့်သွင်းထားပြီးရင် 12.10 / 13.04 စသည်ဖြင့် Upgrade လုပ်သွားလို့ရပါတယ်။ Driver တွေ Software တွေ ထည့်သွင်းပြီးသား အခြေအနေမှာ Distro တစ်ခုအနေနဲ့ ပြန်ပြီး ထုတ်ထားလို့ရပါတယ်။ Windows မှာလဲ Clone ရှိက်လို့ ရတာပဲ Backup လုပ်လိုရတာပဲလို့ ပြောစရာ ရှိပါတယ်။ Windows ရဲ့ Clone က HDD ရဲ့ Sector တွေကို အကောင်း အဆိုးမရွေး ပြန်ရေးသွားတာဖြစ်လို့ OS ဟာ နေးလာတာမျိုး အကြားကြီးမခဲ့ပဲ Error တက်တာမျိုး တွေ့ရပါလိမ့်မယ်။ Linux ရဲ့ Distro Mode ကတော့ Installation ပြန်လုပ်ရတာဖြစ်လို့ မတူညီတဲ့ Partition တွေ HDD တွေမှာပါ ထောက်ပံ့ပါတယ်။ Graphic Driver တွေဆိုရင်လဲ တော်တော်များများ အထောက်အပံ့ပေးပါတယ်။ Nvidia တို့ ATI တို့အတွက်ဆိုရင် Open Source Driver တွေရှိပါတယ်။ အချို့ Game တွေအတွက်တော့ ကုမ္ပဏီက ထုတ်လုပ်ပေးထားတဲ့ Driver တွေလိုအပ်ပါလိမ့်မယ်။ အဲဒီ Driver တွေ အတွက် အခကြေးငွေးဖို့ မလိုပေးမယ် ပြုပြင်ခွင့်မရှိတဲ့ Close Source တွေဖြစ်ပါတယ်။ အဲအပိုင်းမှာ အခက်အခဲတွေ ရှိနေနိုင်တာမျိုး Askubuntu.com လိုနေရာမှာရှာဖွေ ဖတ်ရှုပြီး ကိုယ်တိုင်ပြုလုပ်နိုင်သလို နီးစပ်ရာ LoCo Team တွေနဲ့ Ubuntu User တွေနဲ့ ဆက်သွယ်ပြီး အကူအညီရလှနိုင်ပါတယ်။

Software တွေက Windows မှာဆိုရင်တော့ နာမည်လေးမှတ်ပြီး ဆိုင်မှာ သွားဝယ်ယုံမှု အဆင်ပြောတယ်လို့ ပြောနိုင်ပါတယ်။ Ubuntu မှာတော့ Internet ရှိမှ အသုံးပြုနိုင်တယ်လို့ တော်တော်များများ သိထားကြပါတယ်။ တကယ်တော့ Ubuntu ပေါ်မှာ သုံးတဲ့ Open Source Software တွေကို ရယူနိုင်မယ့် နည်းလမ်းတွေက များပါတယ်။

- Ubuntu Software Center ကနေ ရယူနိုင်ပါတယ်။
- Portable Software အနေနဲ့ ရယူနိုင်ပါတယ်။
- DEB / RPM စတဲ့ Binary တွေကို တိုက်ရှိက် Download ရယူနိုင်ပါတယ်။

- Source တွေကို ကိုယ်တိုင် Binary ထွက်အောင်ပြုလုပ်ပြီး ရယူနိုင်ပါတယ်။
- တဲ့ စက်တစ်လုံးထဲမှာထည့်ထားတဲ့ Software တွေကို Backup လုပ်ပြီး ရယူနိုင်ပါတယ်။

Software Center ကနေ ရယူမယ်ဆိုရင် Software တွေဟာ စိတ်ချရတဲ့ Software တွေအဖြစ် ရရှိနိုင်ပါမယ်။ Catalog အလိုက် ပညာရေး ကျိုးမာရေး သိပုံနဲ့ စီဝါယူရေးသားဖို့ စသဖြင့် ခွဲခြားပြီး ရွေးချယ် ထည့်သွင်းနိုင်ပါတယ်။ အရမ်း လွယ်ပါတယ်။ အင်တာနက်ရှိရှိနဲ့ ထည့်သွင်းနိုင်ပါတယ်။ command line အနေနဲ့ ဆိုရင်လည်း sudo apt-get install ဆိုတဲ့ နောက်က firefox သွင်းချင်ရင် firefox လို့ ရှိက်ထားရုံပါပဲ။ Ubuntu ရဲ့ နာမည်ကြီးလာစေတဲ့ အချက်များစွာထဲက တချက်မှာ apt-get ပါပါတယ်။

OSS တွေဟာ Linux မှာသာမက Windows / Mac တွေပေါ်မှာလဲ အသုံးပြုနိုင်ပါတယ်။ အခြား OS တွေအတွက်ကတော့ လိုအပ်တဲ့ ဖိုင်တွေ အကုန်လုံးပါဝင်ပြီးသား ကူးယူရွှေ့ပြောင်းနိုင်တဲ့ Portable ပုံစံ ဖြစ်ပါတယ်။ Linux အတွက်လဲ အလားတူ Portable အနေနဲ့ ရယူနိုင်ပါတယ်။

Windows ရဲ့ exe, msi တွေလိုပဲ Linux မှာလည်း deb rmp bin စတဲ့ ဖိုင်တွေကို Download ရယူနိုင်ပါတယ်။

Ubuntuပဲဖြစ်ဖြစ် အခြား Linux OSပဲဖြစ်ဖြစ် လိုအပ်တဲ့ Source တွေကို ယူပြီး Binary အဖြစ် ကိုယ်တိုင် ထုတ်ယူနိုင်ပါတယ်။

အခြားစက်ထဲမှာ ထည့်ထားတဲ့ Application တွေ လိုအပ်တဲ့ Library တွေ အကုန်လုံးကို Backup ပုံစံနဲ့ ယူပြီး လိုတဲ့ စက်မှာ ပြန်လည် အသုံးပြုနိုင်ပါတယ်။ Backup နဲ့ Restore မှာ OS Version တော့ တူဖို့ လိုပါလိမ့်မယ်။

တရီးကတော့ Linux ကို သုံးရကောင်းမှန်းလဲ သိတယ် သုံးသင့်မှန်းလဲ သိတယ်။ ဒါပေမယ့် လက်ရှိ Windows ရဲ့ Software တွေကို မစွမ်းလွှာတိနိုင်ကြသေးပါဘူး။ အဲလိုဂူတွေအတွက် Windows Theme တွေ UI တွေ ရေးပေးနေတဲ့ အဖွဲ့အစည်းတွေရှိပါတယ်။ Wine တို့ Play On Linux တို့လိုမျိုး Windows က Application လိုမျိုး ပြန်သုံးနိုင်ဖို့ လုပ်ပေးနေတာတွေလဲ ရှိပါတယ်။ .NET အတွက် Mono Project လည်း ရှိပါတယ်။ ဒီဘက်က Gimp ကို မသုံးချင်ရင် Photoshop ပြန်ပြောင်းသုံးလို့ရပါတယ်။ Libre Office မသုံးချင်ရင် Microsoft ရဲ့ Office ကိုသုံးလို့ရပါတယ်။ ဒီဘက်က IDE တွေ မကြိုက်ရင် DreamWaver ပြန်သုံးလို့ရပါတယ်။ Apache php / python တွေ မဟုတ်ပဲ ASP.NET ကို ပြန်သုံးလို့ရပါတယ်။ ဒါတွေကို Windows ဘက်မှာထက် ပိုကောင်းတယ်လိုတော့ မပြောပါဘူး။ ဒါပေမယ့် Windows ဘက်က Application ပြန်သုံးလို့ရမှာတော့ သေချာပါတယ်။

အမှန်က ကျွန်ုတ်တို့ ဘယ်လို့ အကူအညီပေးခဲ့လဲဆိုတော့ ပထမအဆင့်အနေနဲ့ သူတို့အတွက် Ubuntu ကို လွယ်လွယ်သုံးနိုင်အောင် အခွဲထုတ်တာ ဂိုက်တွေ စုပေးတာလုပ်ပါတယ်။ ဒုတိယအဆင့် အနေနဲ့ အရင်ကလို သူများသုံးပြီးသားကို အဆင်သင့် သုံးနေတဲ့အဆင့်က တဆင့် ကိုယ်တိုင် ရှာ ကိုယ်တိုင် ပြုပြင် ကိုယ်တိုင် လေ့လာပြီး အသစ်တွေပေါ်မှာ ပျော်မွေ့တတ်တဲ့ စိတ်ကလေး ရှိလာအောင် ကူညီခဲ့တာပါ။ နောက်ပိုင်းမှာ Ubuntu သမားတွေဟာ ပြဿနာတွေ တွေ့တိုင်း Windows ဘက်ကလို စိတ်ညစ်နေဖို့ မဟုတ်ပဲ ပြဿနာကို ဖြေရှင်းရတာ ပျော်တတ်လာပါတယ်။ ပြဿနာဖြေရှင်းနည်းကို မျှဝေရတာ ပိုပျော်ဖို့ကောင်းပါတယ်။ ကိုယ်ဖြစ်ဖူးတဲ့ ပြဿနာကို နောက်လှတွေ ဖြေရှင်းယူရတာတဲ့ ရှိက်ယူတတ်လာပါတယ်။ အဲတာမျိုး Ubuntu မှာ Sponsored လုပ်တာများလာရင် Ubuntu ကလည်း ပြန်အသိအမှတ် ပြုပေးပါတယ်။

ကျွန်တော်တို့လို Open Source Web Developer တွေအနေနဲ့
Linux Platform ကို မသုံးမဖြစ် သုံးနိုင်တယ်လို့ ခံစားမိတယ်ပျော်။
အဲဒီတော့ဗျာ Open Source Web Developer တွေအနေနဲ့
ဘယ်လို Linux Distro မျိုးကို ရွေးချယ်သင့်တယ် ဆိုတာ
သိချင်ပါတယ်ပျော်။ Ubuntu တရော ဗိုလ်ချုပ် တွေအတွက်
ဘယ်လို အထောက်အပံ့တွေ ပေးထားလဲပျော်။

Open Source Web Development အတွက်ဆိုတည်းက Open Source Operation System တွေ Open Source Software တွေနဲ့ အသုံးပြုဖို့ ပိုအဆင်ပြေတယ်ဆိုတာ ပြောဖို့ မလိုတော့ဘူး ဖြစ်နေပါပြီ။ ဘယ် Distro ကို ရွေးသင့်တယ်ဆိုတာက ပြောရခဲ့ပါတယ်။ Ubuntu ကိုသုံးမယ်ဆိုရင်တော့ Desktop အတွက်ကော Server အတွက်ကော အဆင်ပြေပါတယ်။ အခြေခံ Command တွေက အတူတူပဲဆိုတော့ ဘယ် Distro ဆိုတာအတွက် ဤနားမှူး သိပ်မရှုတော့ပါဘူး။ Server ကော Desktop ကော အတွက် တွဲသုံးမယ်ဆိုရင် Ubuntu ကို သုံးစေချင်ပါတယ်။ တကယ်လိုများ Server အတွက်ပဲ သီးသန် ထားချင်တယ်ဆိုရင်တော့ Debian / CentOS / Fedora တွေထဲက ရွေးစေချင်ပါတယ်။ အခုနောက်ပိုင်း Debian ဘက်က Ubuntu နဲ့ RedHat ဘက်ကဆိုရင် CentOS က User တွေအတွက် Hot Distro ဖြစ်နေတာကြောင့် Guide တွေ ရှာရလွယ်သလို သိပ်မကွာတဲ့ Command တွေကြောင့် အသစ်လေ့လာဖို့ အချိန်အများကြီး ပေးစရာ မလိုတော့ပါဘူး။

Ubuntuဘက်က Web Development အတွက် သုံးမယ်ဆိုရင် အရင်က Orchestra နဲ့ အခုနောက်စမ်းကြည့်စေချင်ပါတယ်။ RedHat ဘက်မှာလည်း OpenShift က စေတိစားနေပါတယ်။ Juju ထည့်သွင်းပြီးသား Ubuntu Cloud အနေနဲ့ Wordpress CMS ကို ထည့်သွင်းဖို့ Command ကို ပြပါမယ်။

```
juju deploy wordpress
juju deploy mysql
juju add-relation wordpress mysql
juju expose wordpress
```

ကဲ ဘယ်လို သဘောရလဲ Juju ကို Server ဘက်ကနေ သုံးရင် တဗြား Design တို့ Coding ပိုင်းတို့အတွက် အချိန်ပိုမျိန်ဘူးလား။ ဒါကတော့ Server ပိုင်းပါ။ Open Source IDE တွေက Open Source OS တွေ အတွက်သာမက Windows နဲ့ Mac အတွက်ပါ ထွက်လို့ Ubuntu မှ ပိုကောင်းတယ်တော့ မဟုတ်ပါဘူး။ IDE တွေလည်း ရွေးချယ်စရာ များပါတယ်။ သေချာတာကတော့ Open Source Web Development တွေအတွက် Windows ဘက်ထက်စာရင် Ubuntu သုံးတာ ပိုအဆင်ပြေ ပိုကောင်းပါလိမ့်မယ်။

အခုနောက်ပိုင်း ကိုကိုရတို့ အပြောများလာတဲ့ Resberry Pi အကြောင်းလဲ မေးချင်တယ်ပျော်။ သူက ဘယ်လိုမျိုးလဲ၊ ဘာတွေ သုံးလို့ရလဲ၊ ဘာ OS နဲ့ သုံးလို့ ရသလဲ။

Raspberry Pi ဆိုတာကတော့ ကျောင်းသားတွေ Programming Language ကို လွယ်ကူ သက်သာစွာ လေ့လာနိုင်ဖို့ လူတစ်ယောက်ကို ကွန်ပျော်တာ တစ်လုံး ပိုင်နိုင်ဖို့ ရည်ရွယ် ထုတ်လုပ်ခဲ့တဲ့ Project ဖြစ်ပါတယ်။ University of Cambridge's Computer Laboratory ကနေ 2006 ခုနှစ်တည်းက စတင်စဉ်းစားခဲ့တဲ့ Project ဖြစ်ပါတယ်။ ကျောင်းသားတွေ ကွန်ပျော်တာ တစ်လုံးတို့ လွယ်ကူစွာ ဝယ်နိုင်ဖို့အတွက်က ဈေးသက်သာဖို့ လိုပါတယ်။ Raspberry Pi ရဲ့ Version A, Rev 1. ကို ဒေါ်လာ ၂၅ နဲ့ ရောင်းချွန်းခဲ့ပါတယ်။ တကယ့်ကို ထူးခြားတဲ့ အောင်မြင်မှုတစ်ခုပါပဲ။ အချို့။ Remote Control တစ်ခုရဲ့ ဈေးနှုန်းထက်တောင် နည်းနေ့ ပါသေးတယ်။ Version A မှာတော့ USB Port ပဲပါဝင်ပြီး RAM က 256 ပဲရှိခဲ့ပါတယ်။ Version B ဆိုပြီး ထပ်မံထပ်လုပ်မှုမှာတော့ အုံမခန်းဖြစ်သွားပါတယ်။ USB Port ၂ ချုပ်သွားပြီး Ethernet Port ပါပါလာလို့ RJ45 ခေါင်းနဲ့ အင်တာနှင် ဖြစ်ဖြစ် Wifi ဖြစ်ဖြစ် ချိတ်ဆက်နိုင်ပါတယ်။ RAM က 512 ဖြစ်သွားပါပြီး Processor ကတော့ ARM11 ဖြစ်ပြီး 700 MHz နဲ့ အလုပ်လုပ်တာဖြစ်ပါတယ်။ Over Clocking လုပ်မယ် ဆိုရင်တော့ 800 MHz အထိ စမ်းသပ်ထားတာ အောင်မြင်ပါတယ်။

Raspberry Pi မှာ Linux OS တော်တော်များများကို Support ပေးနေပါပြီ။ Android တောင် တင်သုံးလို့ရနေပါပြီ။ NOOBS (New Out Of Box Software) ကိုတော့ Main Official အနေနဲ့ ထုတ်ပေး ထားတာပါ။ ကွန်တော်တို့ကတော့ အဓိကအနေနဲ့တော့ Raspbian OS (Raspberry Pi + Debian) ကိုသုံးပါတယ်။ သူဆိုက်မှာ Official အနေနဲ့ NOOBS, Raspbian, RaspBMC (Media), Arch, RISC OS, Open ELEC, Pidora (Fedora + Pi) တို့ကို Download ပေးထားပါတယ်။ Suse Linux । Bodhi Linux । Kali Linux တွေလည်း အသုံးပြုနိုင်ပါတယ်။ Image (OS / Custom OS) ပေါင်း ၉ ခုကို <http://raspberrypidiskimages.com/> ကနေ ရယူနိုင်ပါတယ်။

Raspberry Pi နဲ့ ဘာလုပ်လို့ရလဲ ဆိုတဲ့ မေးခွန်းကို ဖြေစိုက်တော့ တော်တော်ကြီး ခက်ခဲ သွားပါတယ်။ ဘာဖြစ်လို့လဲဆိုတော့ များပြားလွန်းလို့ပါပဲ။ ကော်မီဖျော်စက်၊ ဂိမ်းစက်၊ ပိုးလေဝသ တိုင်းတာရေး၊ Drone အဝေးထိန်းယာဉ်၊ Home Automation System । Ebook Reader । Twitter Radio । FM Transmitter । 3D Printer । Own Cloud (Dropbox ကဲ့သို့) । Web Server (LAMP) । Node.js । Mail Server । File Server । Camera Controller । ZoneAlert । ThinClient । Super Computer । Mini PC (Mac Mini Style) । Home Automation with iPhone Voice (Siri Proxy + Home Automation) စသည်ဖြင့် မရောတွက်နိုင် လောက်အောင် ရှိပါတယ်။ အကယ်၍ ကိုယ်ရှာချင်တဲ့ Pi Project ကို Google ကနေ ရှာမတွေ့ခဲ့ရင် ပြောပေးပါ။ ကွန်တော်တို့ ကူရှာတာပဲဖြစ်ဖြစ် မရှိခဲ့သေးရင် အစပြုတာပဲဖြစ်ဖြစ် အတူတူ စတင်ကြတာပေါ့။

လက်ရှိ နိုင်ငံတကာမှာ စမ်းသပ်အသုံးပြုပြီး အောင်မြင်တဲ့ Project တွေကို <http://pingbin.com/2012/12/30-cool-ideas-raspberry-pi-project/> ဒီမှာ လေ့လာလို့ရပါတယ်။

Resberry Pi က တန်ဖိုးအတော် နည်းတာပဲဖူ။ အဲဒီလို တန်ဖိုးနည်းတဲ့ အနေအထားကြောင့် ရလာနိုင်တဲ့ အကျိုးကျေးဇူးတွေ အများကြီးပဲလို့ ကျွန်တော်မြင်မိတယ်ဖူ။ ကျွန်တော်တို့ နိုင်ငံအတွက် ပညာရေးကဏ္ဍမှာပဲဖြစ်ဖြစ် အခြား စီးပွားရေး လုပ်ငန်းတွေမှာပဲဖြစ်ဖြစ် အသုံးချလို့ ရနိုင်တဲ့ အနေအထားတွေ သိချင်ပါတယ်။

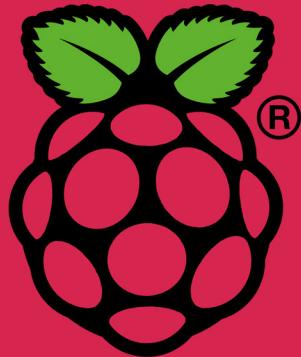
မြန်မာနိုင်ငံအတွက် စဉ်းစားမယ်ဆိုလည်း စဉ်းစားစရာတွေ အများကြီးပါ။ မျက်မမြင် ကလေးတွေ အတွက် eBook Reader တွေ မျက်မမြင်စာ printer တွေ လယ်ယာစိုက်ပျိုးရေးအတွက် ပိုးလေဝသ တိုင်းတွေ့နှင့် ကြောင်းခန့်မျန်းနှင့် ဆေးရုံဆေးခန်းတွေမှာ ဘုံးတဲ့ ဆော့ဝဲတွေအတွက် ကလေးတွေ သိပုံနဲ့ နည်းပညာ သင်ကြားရေးမှာ သင်ထောက်ကုအဖြစ်သုံးဖို့ နယ်မြို့လေးတွေ ရွာလေးတွေမှာ ကွန်ပူတာလို ကုန်ကျားရိတ် အများကြီးမရင်းရပဲ E-Library ထားဖို့ ကလေးတွေကို Programing သင်နဲ့ တစ်ယောက်ကို ကွန်ပူတာ တစ်လုံးပိုင်နိုင်နဲ့ လူကြီး လူငယ်မရွေး အင်တာနှင်း သုံးနိုင်နဲ့ မီးအားလျှပ်စစ်မလိုပဲ နေရာအောင်မရွေး အသုံးပြုနိုင်နဲ့ စသဖြင့် အများကြီး အသုံးပြုနိုင်ပါတယ်။ တန်ဖိုးကြီး Project တွေကို ပြုလုပ်လို့ရသလို အများသုံး တန်ဖိုးနည်း Project တွေကိုလည်း ပြုလုပ်နိုင်ပါတယ်။ နောက်ထပ် ကျွန်တော်တို့ရဲ့ စိတ်ကူးယဉ်အိမ်မက်တွေကို Raspberry Pi နဲ့ အကောင်အထည်ဖော်လို့ရပါသေးတယ်။

အခုလုံး တစ်ယောက်တွေကို ဖော်လုပ်လို့ရသလို အများမြှင့်ဆိုတဲ့ အတွက် Echo က တာဝန်ရှိသူများအားလုံးကို ကျေးဇူးတင်ပါတယ်။ Ubuntu နဲ့ Raspberry Pi ကို အများမြှင့်ဆိုတဲ့ အတွက် တကယ်ပဲ ဝမ်းသာ ဂုဏ်ယူပါတယ်။ ဖတ်မိတဲ့သူတွေအတွက်လည်း Idea တစ်ခုခုတော့ ရသွားဖို့ မော်လင့်မိပါတယ်။

အခုလုံး ဖော်လုပ်လို့ရသလို စိတ်ရှည်လက်ရှည် ဖြေဖော်လို့အတွက်လဲ { echo } က ကိုကိုရောက် ကျေးဇူးတင်ပါတယ်ခင်ဖူ။

■ RPi Introduction

by Ubuntu Loco Team



about

Ubuntu Loco Team



ubuntu-mm.net



[/ubuntumm](#)

Raspberry Pi ဆိတာ ဘာလဲ

RPi ဆိတာကတော့ Raspberry Pi Foundation ကနေပြီးတော့ Project အောင်မြင်ကာ တစ်လုံးကို ၂၅ဒေါ်လာကနေ ၃၅ဒေါ်လာ ရေးနဲ့ ထုတ်လုပ်ရောင်းချနေသော ကလေးများအတွက် ရည်ရွယ်ထားတဲ့ SoC (System on Chip) ကွန်ပျူတာ တစ်ချိုးပဲ ဖြစ်ပါတယ်။ ရေးနှုန်းလဲ သက်သာသလို Credit ကိစ္စနဲ့ ဆေးလိပ်ဘူး တစ်ခုအချယ်ပဲ ရှိလို ပေါ့ပါးသေးငယ်ပါတယ်။ (ငှုံးတို့ အသိအမှတ်ပြု ကျောင်းများသို့ တရားဝင်လှုပါန်းမှသာ ၂၅-၃၅ ဒေါ်လာဖြင့် ရောင်းချပေးခြင်း ဖြစ်ပါသည်။ အပြင်တွင် Tax နှင့် အေား အရောင်းအဝယ်ဝန်ဆောင်မှုများမှ အခွန်များထောင်ပြီး ဝယ်ယူနိုင်သည်။)

A နဲ့ B ရှိတယ်လို့ သိရပါတယ်။ ဘာတွေကွာခြားမလဲ။

Model A ကတော့ 256MB RAM ပဲ ရှိမှာဖြစ်ပြီး USB Port တစ်ခုပါဝင်ပြီး Network Port မပါဝင်ပါဘူး။ Model B ကတော့ 512MB RAM ဖြစ်သွားပါတယ်။ USB Port ၂ခုနဲ့ Ethernet Port ပါဝင်လာလို့ RJ45 နဲ့ ချိတ်ဆက်ပြီး သုံးနိုင်ပါတယ်။

SoC လို့ ပြောသွားပါတယ်။ ဘယ်လို့ Chip မျိုးပါလဲ။

Broadcom BCM2835 ကို သုံးထားပြီး ARM11 Processor ကို သုံးထားပါတယ်။ ရေးနှုန်းချိသာပြီး သင့်တင့်တဲ့ အမျိုးအစားဖြစ်ပါတယ်။ 700MHz နဲ့ အလုပ် လုပ်နိုင်ပါတယ်။ Overclocking လုပ်မယ်ဆိုရင် 800MHz နဲ့ အထိ အဆင့်ပြောပါတယ်။ Video အနေနဲ့ 1080p30 H.264 ကို 40MBits Per Second နဲ့ ပြောပေးနိုင်ပါတယ်။ ရှုပ်ထွက်ဖို့အတွက် HDMI နဲ့ Video Composite ကြိုးကို အသုံးပြန်ပါတယ်။ VGA ကိုတော့ တိုက်ရှိကို အထောက်အပံ့ မပေးတော့ပါဘူး။ Converter တွေနဲ့တော့ သုံးနိုင်ပါတယ်။ Audio အတွက်တော့ HDMI က အသုံးပြန်စွာ အထောက်အပံ့ရသလို Audio Jack လဲ ပါပါတယ်။

Power ပေးစို့အတွက်

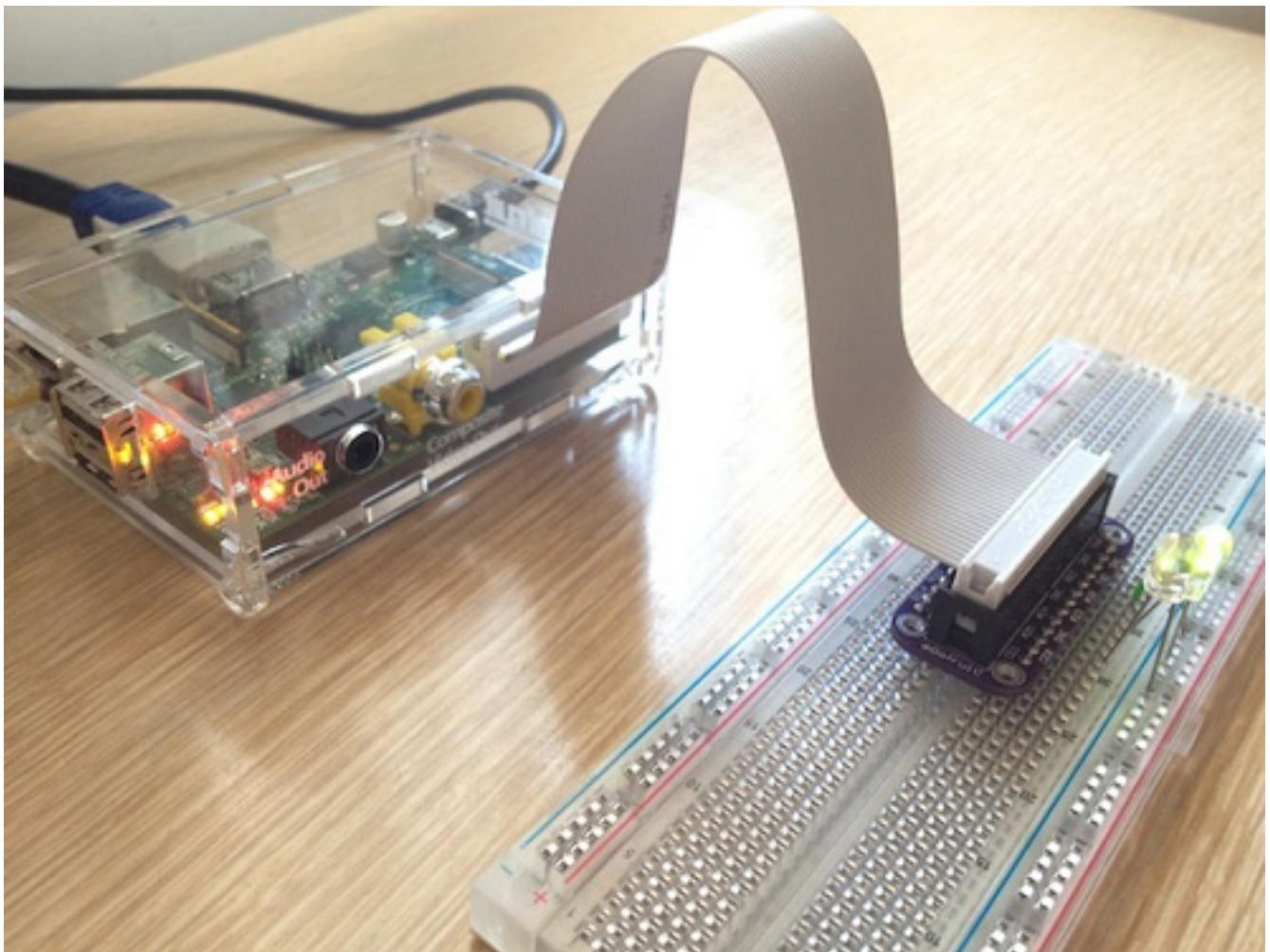
5V 1A ဆိုရင် ရပါတယ်။ သို့မဟုတ် 5V 2A လဲ သုံးနိုင်ပါတယ်။ ကြိုးကတော့ Android ဖုန်းတွေမှာ အားသွင်းတဲ့ Mini-USB ဆိုရင် ရပါတယ်။ ဒါတွေ အကုန်လုံးကို အားသွင်းကြိုးနဲ့ ပါဝါကို တူညီစေလိုတဲ့ EU Law နဲ့ ကိုက်ညီအောင် လုပ်ထားပါတယ်။ သိပ်မကြာခင်မှာပဲ Network ကြိုးထဲကနေ ပါဝါစို့လွှတ်တဲ့ PoE နဲ့ ထပ်ထွက်လာနိုင်ပါသေးတယ်။

OS တွေအနေနဲ့

Linux Distro တော်တော်များများနဲ့ အဆင်ပြောပါတယ်။ အခုခုံး Android တွေတော် စမ်းသပ်ပြီး တင်သုံးနေကြပါပြီ။ လက်ရှိတော့ Debian Base Raspbian OS ကို အသုံးများကြပါတယ်။ <http://www.raspberrypi.org/downloads> ကနေ Download ရယူနိုင်ပါတယ်။

OS ပျက်သွားရင်

လွယ်လွယ်ကူကူပဲ အသစ်ပြန်တင်လို့ ရပါတယ်။



လက်ရှိ နိုင်ငံတကာတွင် အသုံးများနေသော Project များကို လေ့လာရန် -

<http://pingbin.com/2012/12/30-cool-ideas-raspberry-pi-project>

Videos -

iPhone Voice Command - Raspberry Pi Home Automation Control

<http://www.youtube.com/watch?v=PXMmCiaRc9XU>

Kickstarter - BrickPi -

<http://www.youtube.com/watch?v=IFH-dxR6Neg>

R2D2 Robot -

Features:

Voice Control (in English and Chinese, Using PocketSphinx)

Face Recognition (Using OpenCV)

Motion Detection Ultrasonic

Distance Detection

Audio Message Record and Replay

Sound Play and TTS

Rechargeable Battery

Wifi

<http://www.youtube.com/watch?v=znuUm5vbSpl>

Glossary -

BGA: ball grid array. A type of surface mount packaging for electronics.

SoC: system on chip. A computer on a single chip.

GPIO: General purpose input/output. A pin that can be programmed to do stuff.

GPU: graphics processing unit. The hardware handles the graphics.

Distro: a specific package ("flavour") of Linux and associated software.

Brick: to accidentally render a device inert by making changes to software or firmware.

Pxe: preboot execution environment. A way to get a device to boot by via the network.

PoE: power over ethernet. Powering a device via an ethernet cable.

Building an HTML5 Contact Form

by Hein Zaw Htet



about
THIHA

ကျွန်တော်ကတော့ ဟိန်းအောင်ယူပါ။ MyanmarWebDev.com နဲ့ Echo မှာ စာပုံမှန်ရေးပါတယ်။ Myanmar Links မှာ Web Designer/Developer တစ်ယောက်အဖြစ် လုပ်ကိုင်နေပါတယ်။



myanmarwebdev.com



[/heinzawhtet](#)



[@heinzawhtet](#)



[/heinzawhtet](#)

Step 0 - Preparing Files

ကျွန်တော်တို့အခြေခံဆောက်မယ့် Project Folder ထဲမှာ index.html နဲ့ style.css ဆိုပြီး ဆောက်လိုက်ပါမယ်။ Form Validation အတွက် jQuery (<http://www.jquery.com/download>) နောက်ဆုံး Version နဲ့ jQuery Validation Plugin (<http://jqueryvalidation.org/>) ပါ Download ချိပြီး project folder ထဲ ထည့်ထားပါမယ်။

Step 1 - HTML & CSS Boilerplate

HTML အခြေခံ Boilerplate ကို အောက်ကအတိုင်းရေးလိုက်ပါမယ်။ အဲဒီထဲမှာ head ထဲမှာ style.css နဲ့ body အပေါ်ဖက်မှာ js နှစ်ခုကိုပါ ချိတ်ထားပါမယ်။

```
<!DOCTYPE html>
<html>
<head>
    <title>Contact Form</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>
<body>

<script src="jquery-2.0.3.min.js"></script>
<script src="jquery.validate.min.js"></script>
</body>
</html>
```

style.css မှာတော့ body ရဲ့ background, font နဲ့ h1 အတွက် Style ရေးထားပါတယ်။ Font ကိုတော့ Dosis (<http://www.google.com/fonts/specimen/Dosis>) သုံးထားပါတယ်။

```
body {
    background: #2d232c;
    font-family: Dosis
}
h1 {
    color: #fff;
```

Step 2 - HTML Markup for Form

အရင်ဆုံး Body Element ထဲမှာ ရိုဘူးမျှ အားလုံးကို အုပ်ထားဖို့အတွက် container ရေးလိုက်ပါမယ်။ အဲဒီ container ကို width - 400px သတ်မှတ်ပေးပြီး အလယ်ပို့ထားလိုက်ပါမယ်။

```
<div class="container">  
</div>
```

style.css ထဲမှာတော့ အောက်ကအတိုင်းရေးရပါမယ်။

```
.container {  
    width: 400px;  
    margin: 0 auto;  
}
```

ကျွန်တဲ့ Element အားလုံးကို ဒီ Container ထဲမှာပဲ ရှိမှာပါ။ Form အတွက် index.html ထဲမှာ အောက်ကအတိုင်းရေးပါမယ်။ Submit Button ကလွှဲဖြီးတော့ ကျွန်တဲ့ Input တွေကို div.field ထဲမှာ ထည့်ရေးထားပါတယ်။ ဒါလိုပေါ်မှ သက်ဆိုင်ရာ Input ရဲ့ဘေးမှာ Error ထွေ ဖော်ပြုလိုအဆင်ပြေမှာပါ။

```
<form class="contact-form" id="contact-form">  
    <div class="field">  
        <input type="email" id="email" name="email" class="email-field"  
placeholder="Email" required>  
    </div>  
    <div class="field">  
        <input type="text" id="name" name="name" class="name-field" placeholder="Name"  
required>  
    </div>  
    <div class="field">  
        <textarea id="message" name="message" class="message-field"  
placeholder="Message" required></textarea>  
    </div>  
    <input type="submit" class="submit-field" value="Submit">  
</form>
```

Step 3 - Styling Form

Form အတွက် CSS ရေးပါမယ်။ div.field ကိုတော့ Position - Relative ပေးထားပါတယ်။ Error ပြရင်အဆင်ပြအောင်လိုပါ။ Submit Button က သာမန် Input လိုမဟုတ်တဲ့အတွက် .contact-form input[type="submit"] ဆိုပြီး သူ့အတွက် သီးသန့် Style သက်မှတ်ပေးထားပါတယ်။

```
.contact-form .field {
    position: relative;
}
.contact-form input, .contact-form textarea {
    display: block;
    background: #5f4a5d;
    border: none;
    border-radius: 3px;
    resize: none;
    padding: 10px;
    color: #dcabd7;
    font-family: Dosis;
    margin: 15px 0;
    width: 100%;
}
.contact-form input[type="submit"] {
    background: #74ca8e;
    box-shadow: 0 3px 0 #68b57f;
    color: #3e6c4c;
    width: 100px;
    font-size: 16px;
}
```

နောက်ထပ်လိုသေးတာက Error တွေအတွက်ပါ။ Form Submit လုပ်လိုက်လို မှားနေတယ်ဆိုရင် အောက်ကပုံအတိုင်းပြုမှပါ။

Contact us

Email

This field is required.

Name

This field is required.

Message

This field is required.

Submit

Error Message မှ Input ကို ညွှန်ပြထားတဲ့များပုံစံအတွက်ကိုတော့ :before နဲ့သံဃ္ပီးရေးထားပါတယ်။ style.css ထဲမှာ အောက်ကအတိုင်းရေးလိုက်ပါ။

```
.contact-form input.error,.contact-form textarea.error {  
    border: 2px solid #ff836c;  
}  
.contact-form label.error {  
    background: #fd6d52;  
    display: block;  
    padding: 10px;  
    width: 200px;  
    border-radius: 3px;  
    border: 1px solid #ff836c;  
    color: #5e281e;  
    position: absolute;  
    left: 440px;  
    top: 0;  
    cursor: pointer;  
}  
.contact-form .error:before {  
    width: 15px;  
    height: 15px;  
    margin-left: -18px;  
    margin-top: 3px;  
    display: block;  
    position: absolute;  
    -webkit-transform: rotate(-45deg);  
    -moz-transform: rotate(-45deg);  
    -o-transform: rotate(-45deg);  
    transform: rotate(-45deg);  
    display: none\0/;  
    *display: none;  
    background: #fd6d52;  
    content: ' ';
```

ကျွန်ုတ်တို့လိုချင်တဲ့ပုံစံအတိုင်း ထွက်လာဖို့အတွက် jQuery Plugin ကို ခေါ်သံဃ္ပီတော့မယ်။

Step 4 - Initializing Validation Plugin

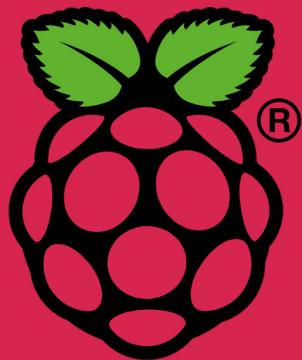
အပေါ်ဖက်မှာ ချိတ်ခဲ့တဲ့ javascript ထွေအောက်ဖက်မှာ အောက်ကအတိုင်း ရေးလိုက်ပါ။

```
<script>  
    $("#contact-form").validate();  
</script>
```

ဒီလိုဆို Form Validation လည်း အလုပ်လုပ်ပါပြီ။

Raspberry Pi Setup

by Ubuntu Loco Team



about

Ubuntu Loco Team



ubuntu-mm.net



[/ubuntumm](#)



RPi အတွက် လိုအပ်သော စာရင်း

USB Keyboard နှင့် Mouse (သို့မဟုတ်) Bluetooth Keyboard နှင့် Mouse

SD Card နှင့် Card Reader

Ethernet Cable နှင့် Network Connection (Optional)

Power Supply နှင့် USB Cable (5V 2A)

Display

HDMI Cable နှင့် HDMI TV/Monitor

သို့မဟုတ်

TV နှင့် AV + Audio to AV Jack

သို့မဟုတ်

HDMI 2 VGA Converter and VGA Monitor

တကယ်လို့ Media Server တွေ File Server တွေဆိုရင်တော့ ကိုဘုတ်တွေ ဖောက်တွေ အမြဲမလိုပါဘူး။ Setup လုပ်တုန်းပဲ လိုပါလိမ့်မယ်။ Media Server တွေဆိုရင်တော့ Control လုပ်ဖို့ Wifi + iPhone / Android စတာတွေကို သုံးနိုင်သလို USB-IR အသုံးပြုတဲ့ Remote တွေလည်း အကျယ်တကူဝယ်ယူ အသုံးပြနိုင်ပါတယ်။



Power Supply

ပါဝါကိုတော့ 5V, 1A ထွက်တဲ့ အအက်ပတာ တစ်ခုနဲ့ဆို အဆင်ပြေပါတယ်။ ကိုယ်က Computer လေးအနားမှာပဲ စမ်းမယ်ဆိုရင် USB ကနေ ပါဝါပေးလို့ ရပါတယ်။ အားသွင်းကြိုးကိုတော့ Universal Micro USB ကြိုးနဲ့ဆို အဆင်ပြေပါပြီ။ မကြာခင်က ထုတ်ပြန်ထားတဲ့ ဥပေါ်များမှာ များပြားလုတဲ့ ပိုဘိုင်းဖုန်းတွေကို အားသွင်းဖို့ တည်တည့်တည်း သတ်မှတ်ထားတဲ့ စံပြစ်ပါတယ်။ ဒီအတွက် EU က လက်ခံတဲ့ သာမဏ် အန်းအရှိက် ဖုန်းအအက်ပတာ တစ်ခုဆို လုံလောက်ပါပြီ။ ဒါပေမယ့် 1A နဲ့ဆိုရင်တော့ များပြားတဲ့ USB Keyboard USB Mouse USB Hub Ext HDD စတာတွေနဲ့ အဆင်ပြေမှာမဟုတ်ပါဘူး။ Bluetooth Keyboard နဲ့ Mouse ကတော့ ပါဝါတဲ့ ပိုသက်သာစေနိုင်ပါတယ်။



SD Card

SD Card ကိုတော့ Class 4 ဖြစ်ဖို့ ဂရူစိုက်ရမှာဖြစ်သလို အနည်းဆုံး 4GB ရှိမှ အဆင်ပြေမှာဖြစ်ပါတယ်။ 4GB ထက်ကျော်လွန်မှသာ 0S အပြင် ကိုယ်ထပ်မံထည့်သွင်းလိုတဲ့ Application တွေနဲ့ Data တွေအတွက် နေရာလွှတ်ရမှာ ဖြစ်ပါတယ်။ ဒီအတွက် SD Card ကော့ Card Reader တွေ USB Hub တွေ RPi ကနေ ထည့်သွင်းပေးထားမှာ မဟုတ်ပါဘူး။ ကိုယ့်ဘာသာ ဝယ်ယူရမှာပဲ ဖြစ်ပါတယ်။ မဝယ်သင့်တဲ့ ကဒ်ကို မဝယ်မိအောင် http://elinux.org/RPi_SD_cards မှာ သွင်းပြီး ကြည့်ရှုသင့်ပါတယ်။ တရာ့ကိုတွေက Boot မတက်နိုင်တာမျိုး ရှိပါတယ်။ SD SDHC 4GB ကို ခန့်မှန်း ရှုပေါ် ကျပ်မှ ပေါ်အထိ ရှိနိုင်ပါတယ်။



HDMI/Composite Cable

ဒီနေရာမှာတော့ နည်းနည်းအကုန်အကျများသွားနိုင်ပါတယ်။ ဘာဖြစ်လိုလဲဆိုတော့ HDMI TV/Monitor တွေ မရှိခဲ့ဘူးဆိုရင် အဆင်မပြေလိုပါပဲ။

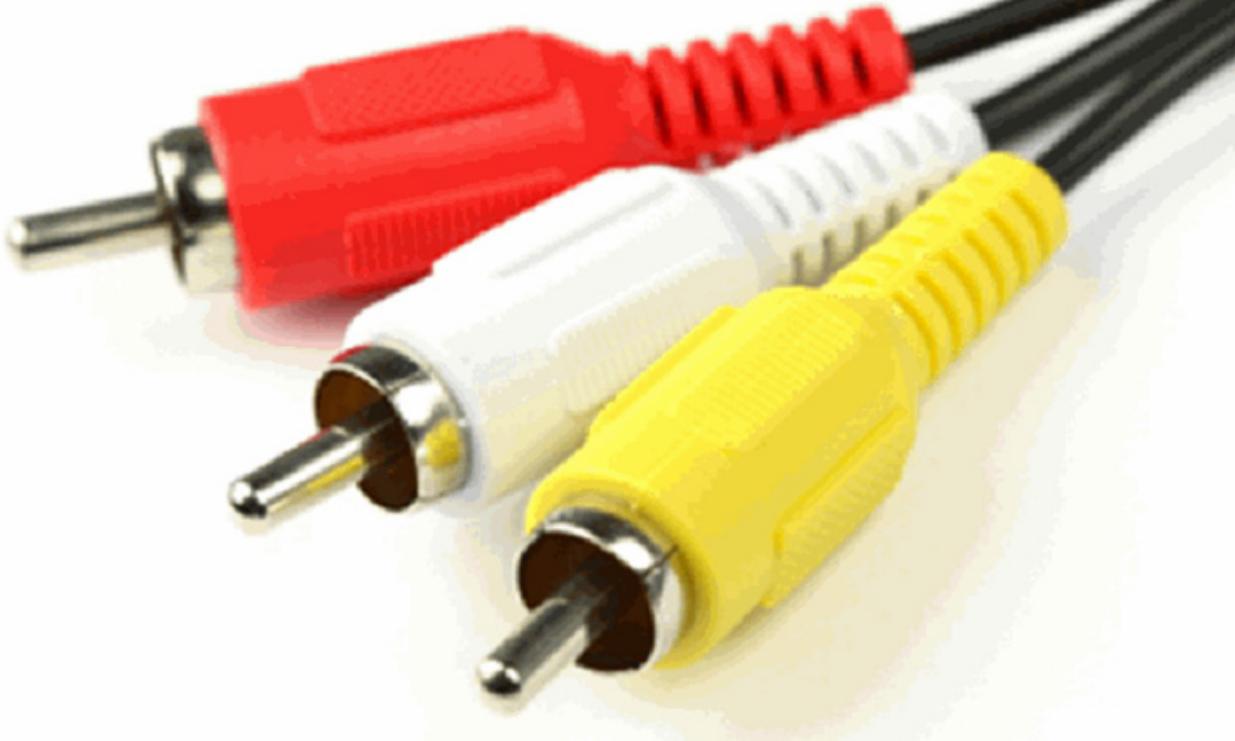
၁ - HDMI TV/Monitor ရှိခဲ့ရင်တော့ အကောင်းဆုံးပါပဲ။ HDMI ကြိုးပဲ လိုပါမယ်။ ခန့်မှန်းခြေ ၁၅၀၀ ကျပ်ကနေ ၃၀၀၀ ဝန်းကျင်နဲ့ ဝယ်ယူနိုင်ပါတယ်။

၂ - DVI Monitor ရှိတယ်ဆိုရင်တော့ HDMI to DVI Converter လေးလိုပါလိမ့်မယ်။ ခန့်မှန်းခြေ ကျပ် ၃၅၀၀ ကနေ ၄၅၀၀ အတွင်း ရှိပါတယ်။

၃ - TV ပဲ ရှိတယ်ဆိုရင်တော့ AV ကြိုးနဲ့ Audio Jack 2 Composite ကြိုး လိုပါလိမ့်မယ်။ တစ်ကြိုးကို ၁၀၀၀ ကျပ် မကော်ပါဘူး။

၄ - VGA Monitor ပဲရှိတယ်ဆိုရင်တော့ HDMI 2 VGA Converter လိုပါလိမ့်မယ်။ ၉၅၀၀ ကျပ်လောက် ပေးရမှာဖြစ်ပြီး အသံလည်းထွက်မှာ မဟုတ်ပါဘူး။ အဆင်လည်း သိပ်မပြေလို့ ဒီလိုသုံးမယ်ဆို သေချာစဉ်းစားသင့်ပါတယ်။





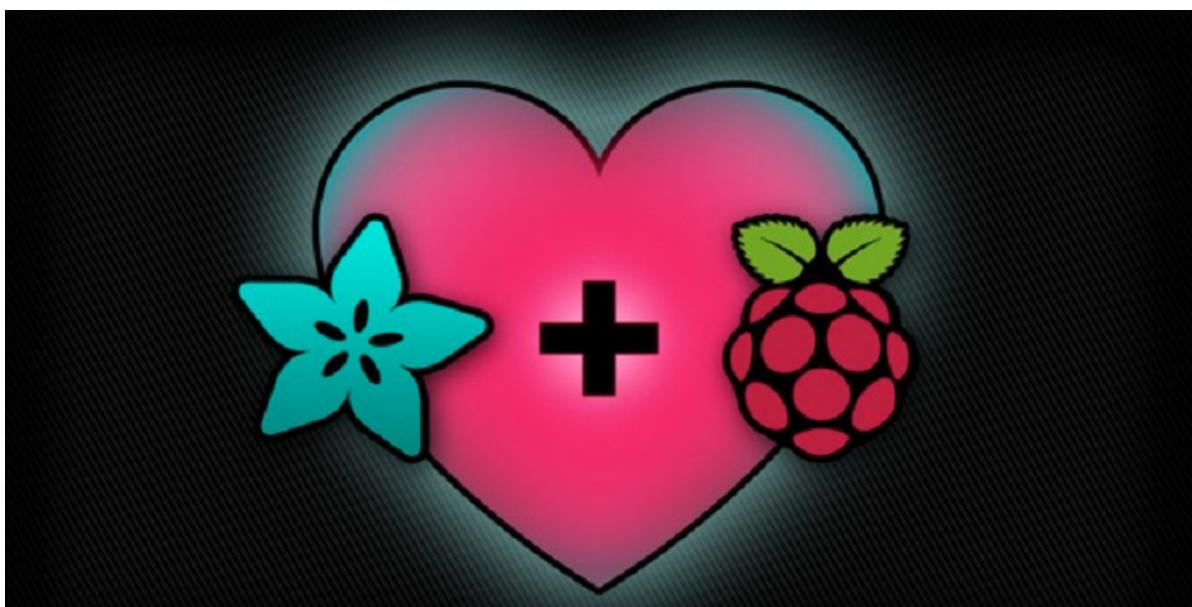
OS (Operating System)

ကျွန်တော်တို့ သုံးနေတဲ့ Windows/Linux/Mac/Android စတာတွေဟာ OS တွေပါပဲ။ OS တွေဟာ စေခိုင်းချက်တွေကို သူ၏ Hardware တွေဆီက တစ်ဆင့် လက်ခံရယူနိုင်ပါတယ်။ Output မျိုးစုံကို ဖို့တွေတိပေးနိုင်ပါတယ်။ Output ဆိုတာ အချက်ပြမိုးတစ်လုံး ဒါမှမဟုတ် စိတ်ကာက အသံတစ်ခု Monitor တစ်ခုဖြစ်နိုင်သလို Network လဲ ဖြစ်နိုင်ပါတယ်။ တကယ်လို့ OS မပါခဲ့ရင် အဲဒီ ကွန်ပူးတာဟာ ဘာမှ လုပ်နိုင်မှာ မဟုတ်ပါဘူး။

သာမန်ကွန်ပူးတာ တစ်ခုဟာ ပုံမှန်အားဖြင့် OS ကို Hard Drive ထဲမှာ သိမ်းဆည်းထားပြီး စက်ဖွင့်ချိန်တိုင်းမှာ RAM ထဲကို OS ကို ဆွဲခေါ်ရပါတယ်။ စဖွင့်ရင် Hard Disk ထဲက ဒေါ်ယူပြီး ပိတ်သိမ်းပြီးဆိုရင်လည်း Hard Disk ထဲမှာ သိမ်းပါတယ်။ ဒါပေမယ့် RPi မှာတော့ Hard Disk မပါလာပါဘူး။ Hard Disk အတား SD Card ကိုသုံးထားပါတယ်။ ကျွန်တဲ့ အလုပ်လုပ်ပုံကတော့ အတူတူပါပဲပဲ။

ဒီနေရာမှာ ကျွန်တော်တို့ဟာ ကွန်ပူးတာအကြီးအသေး ဘာဝယ်ဝယ် OS ပါပြီးသားလို့ သိထားကြပါတယ်။ RPi ကို ဒီတိုင်း သွားဖွင့်မယ်ဆိုရင်တော့ ဘာ OS မှ တက်လာမှာမဟုတ်ပါဘူး။ သိပ်ပြီး အဲ့သွေရရာ မရှိပါဘူး။ ကျွန်တော်တို့ လက်ရှိ သုံးနေတဲ့ Windows/Linux/Mac နဲ့တင် RPi အတွက် OS ထည့်ထားတဲ့ SD Card တစ်ခု ဖန်တီးယူနိုင်ပါတယ်။

Adafruit Pi OS



RPi OS

RPi အတွက် OS ကို သူ့ဆိုက်မှာ တရားဝင်အခဲ့ ပေးထားတာတွေ ရှိပါတယ်။ အချို့သော စမ်းသပ်ဗားရှင်းတွေကိုလည်း ဂုဏ်သုံးပြီး ရှာဖွေနိုင်ပါသေးတယ်။ သူနဲ့ အကိုက်ညီဆုံး OS တွေကိုတော့ <http://www.raspberrypi.org/downloads> မှာ သွားရောက်ရသူနဲ့နိုင်ပါတယ်။ သူတို့ အသိအမှတ်ပြု ထုတ်ပေးထားတာတွေထဲက အကောင်းဆုံးကတော့ Wheezy လို့ ပေါ်တဲ့ Debian Distro ပါပဲ။ ဖိုင်ဆိုက်က ငါေဝ မက်ဂါဘိုက် ကျော်မှာဖြစ်လို့ ဆွဲရကူးရ အဆင်မပြောင် ကျွန်တော်တို့ကို ဆက်သွယ်ပြီး လာယူနိုင်ပါတယ်။ ပြီးရင်တော့ 4GB ရှိတဲ့ SCHC SD Card ရယ် SD Card Reader ရယ် လိုအပ်လာပါပြီ။ ပထမဆုံး Download လုပ်ထားတဲ့ ဖိုင်က Zip ဖိုင် ဖြစ်မယ်ဆိုရင်ဖြည့်ချလိုက်လို့ရပါတယ်။ အဲဒါဆိုရင် အထဲက ထုတ်လုပ်ထားတဲ့ ရက်စွဲနဲ့ IMG ဖိုင်တစ်ခုကို တွေ့ရပါလိမ့်မယ်။ အဲဒီ img ဖိုင်ကို ကျွန်တော်တို့ SD Card ထဲကို Write လုပ်ရမှာဖြစ်ပါတယ်။ Copy / Paste တို့ Drag လုပ်ရမှာတို့ မဟုတ်ပါဘူး။

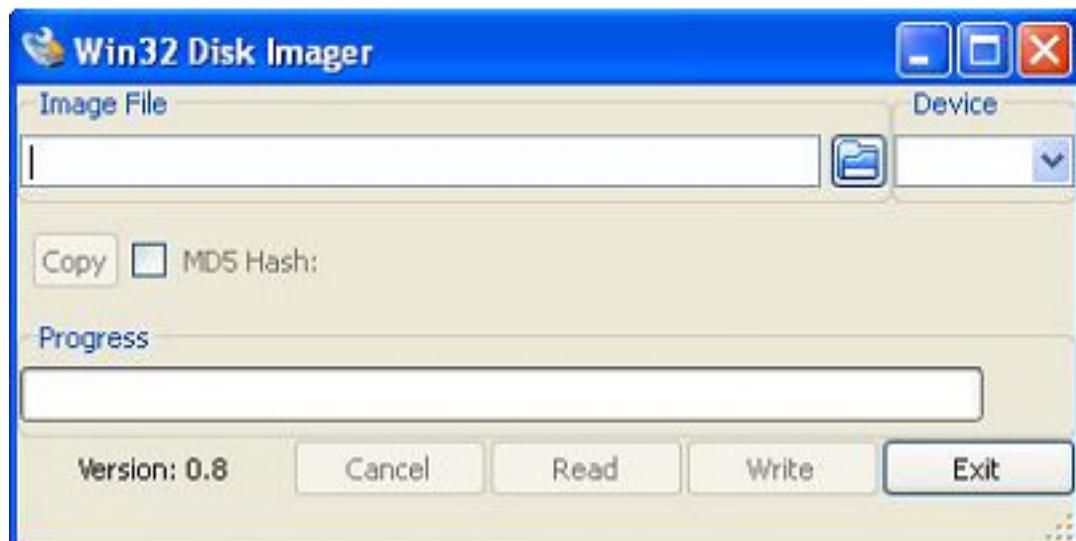
The screenshot shows the official Raspberry Pi Downloads page at www.raspberrypi.org/downloads. The page features the Raspberry Pi logo and navigation links for 'Quick Start' and 'Downloads'. A central text block provides instructions for using raw images, mentioning the use of the 'dd' command or Win32DiskImager. Below this, two sections are shown: 'Raspbian "wheezy"' and 'Soft-float Debian "wheezy"'. Each section lists download links for Torrent, Direct download, SHA-1 hash, and Default login credentials (Username: pi, Password: raspberry).

Torrent	2013-07-26-wheezy-raspbian.zip.torrent
Direct download	2013-07-26-wheezy-raspbian.zip
SHA-1	f072b87a8a832004973db4f5e1edb863ed27507b
Default login	Username: pi Password: raspberry

Soft-float Debian "wheezy"

SD Card Writer

ဒီနေရာမှာ ကျွန်တော်တို့ ဘယ် OS သုံးထားလဲဆိုတာ ပေါ်မှတ်လျှပ်ပြီး ဆော့ဝဲတွေ သုံးရတာ တူမှာမဟုတ်တော့ပါဘူး။ Windows ဆိုရင်တော့ Win32DiskImager ကို သုံးရမှာ ဖြစ်ပါတယ်။ Linux နဲ့ Mac ဆိုရင်တော့ dd ဆိုတဲ့ Command တစ်ခုကို သုံးရုံးနဲ့ အဆင်ပြောပါတယ်။ ဖိုင်ကို Download လုပ်ပြီး ဖိုင်ဆိုက်ကို တိုက်ဆိုင် စစ်ဆေးတဲ့ Sha1sum ဆိုရင်လည်း Windows အတွက် Download ပြုလုပ်ရမှာ ဖြစ်ပြီး Linux (Ubuntu) မှာတော့ တပါတည်းပါပြီးသားပဲ ဖြစ်ပါတယ်။ zip ဖိုင်ကို မဖြေချခေါ်မှာ sha1sum နဲ့ တိုက်စစ်လိုက်လို့ ရပါတယ်။ အဲဒါဆိုရင် ဖိုင်ဆိုင်မပြည့်လို့ OS ကောင်းကောင်း အလုပ်မလုပ်တာကို ရှောင်လွှားနိုင်ပါလိမ့်မယ်။



Windows အတွက် Win32DiskImager ကို <http://bit.ly/Dh2nE> ကနေ Download ရယူနိုင်ပါတယ်။ SD Card ကို SD Card Reader မှာ ထပ်ပြီး Format ချထားသင့်ပါတယ်။ ပြီးခဲ့ရင်တော့ Win32DiskImage ကနေ RaspberryPi SD Card ကို SD Card Reader မှာ ထပ်ပြီး Format ချထားသင့်ပါတယ်။ ပြီးခဲ့ရင်တော့ Win32DiskImager ကနေ RaspberryPi IMG ဖိုင်ကို Write လုပ်ပါယ်။ အရမ်းလွယ်ပါတယ်။ Image ဖိုင်နေရာ မှာ img ဖိုင်ကို ရွေးပေးထားရမှာဖြစ်ပြီး Device နေရာမှာတော့ D: ဖြစ်စေ E: ဖြစ်စေ SD Card ရဲ့ Drive Letter ကို ရွေးပေးရမှာဖြစ်ပါတယ်။ Write ကို နိုဝင်ပြီး ပြီးဆုံးမယ့် အချိန်ကို စောင့်နေရှုပါပဲ။ ပြီးရင်တော့ OS ထည့်သွင်းထားတဲ့ SD Card ကို ရပါပြီ။

Linux အတွက်တော့ ဘာမှ ထပ် Download ရယူစရာမလိုပါဘူး။ တရာ့၏ Root ခွင့်ပြုချက်လိုတဲ့ Command တွေ သုံးမယ်ဆိုရင်တော့ Super User Do ဆိုတဲ့ sudo ကို ရှေ့က ထားသုံးရပါလိမ့်မယ်။ ပထမဆုံး SD Card ကို Reader မှာ တပ်ထားပြီး သိမသိ စမ်းအိုလိုပါတယ်။ အဲဒီအချိန်မှာ ရုပ်တွေးမှု မဖြစ်အောင် အခြား SD Card တွေ USB Stick တွေကို ဖြော်ထားသင့်ပါတယ်။

```
$ sudo fdisk -l
```

sudo fdisk -l နဲ့ ကြည့်ထားရမှာဖြစ်ပြီး အဲဒီမှာ Windows မှာ C: တွေ D: E: တွေ ပေါ်သလိုပဲ sda sdb စသည်ဖြင့် မှတ်သားထားရမှာပါ။

```
Partition table entries are not in disk order

Disk /dev/sdc: 1970 MB, 1970722816 bytes
61 heads, 62 sectors/track, 1017 cylinders, total 3849068 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start        End      Blocks   Id  System
Disk /dev/sdd: 3960 MB, 3960995840 bytes
122 heads, 62 sectors/track, 1022 cylinders, total 7736320 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000
```

ဖိုင်ဆိုဒ်စစ်ဖို့ sha1sum ကော် ဖိုင်ကို SD Card မှာ ရှိက်ဖို့ dd ကော် ပါပြီးသားပါ။ dd ကို သုံးမယ်ဆိုရင် sudo နဲ့ တွဲသုံးရပါမယ်။

```
sudo dd bs=1MB if=~/Download/raspbian.img of=/dev/sdb
```

```
sudo dd bs=1MB if=~/Downloads/raspbian.img of=/dev/sdd
```

sudo ကတော့ super user do ပါ။ dd ကတော့ Disk Drive ကို Copy သို့မဟုတ် Convert လုပ်နိုင်ပြီး ဒီမှာတော့ write လုပ်တဲ့ Command ပါ။ bs ကတော့ dd ရဲ့ byte အတွက် သတ်မှတ်ပေးတာ ဖြစ်ပြီး 1MB မှာ MB ကို အကြီးနဲ့ ရေးပေးရပါမယ်။ if ကတော့ input file ကိုရွေးပေးရမှပါ။ of ကတော့ stdout အတွက် output ပေးမယ့် sd card ကို ရွေးပေးရမှာ ဖြစ်ပါတယ်။

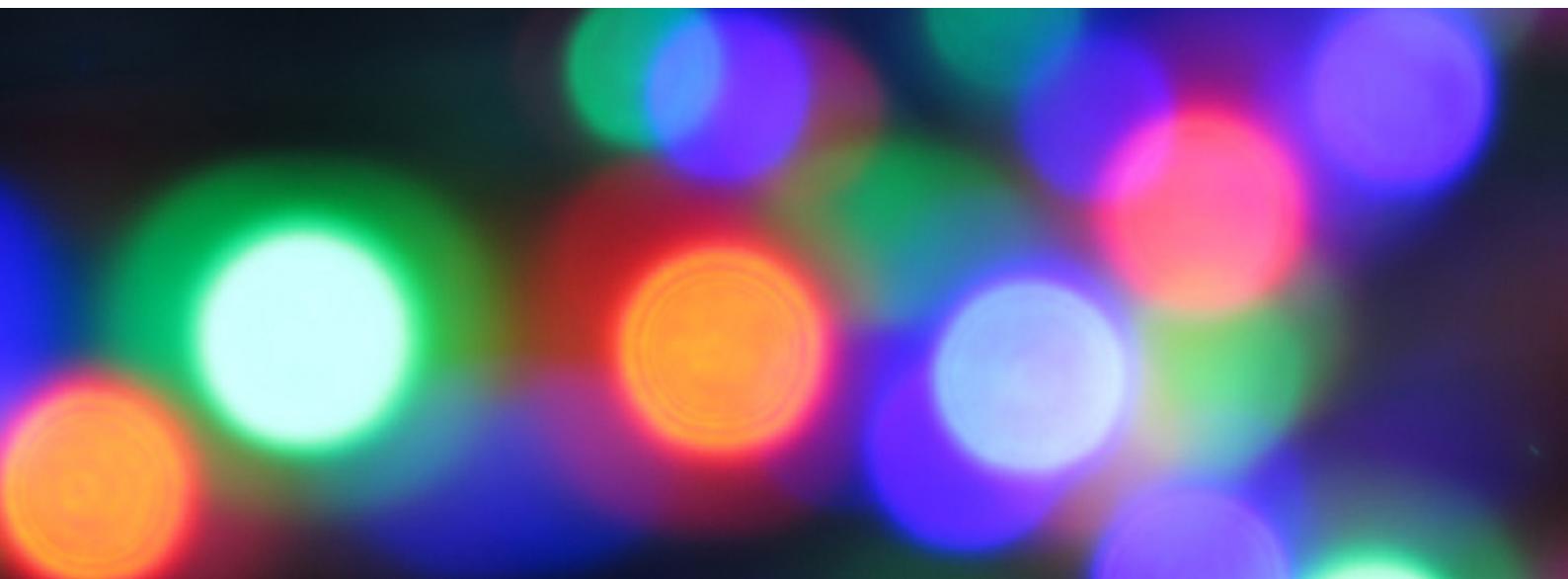
Enter ခေါက်ပြီးတာနဲ့ password တောင်းတဲ့ prompt ကို တွေ့ရမှာပါ။ ဒီနေရာမှာတော့ ကျွန်တော်တို့ Linux မှာလုံခြုံရေးအရ Password ဘယ်နှစ်လုံးဆိုတာ မှတ်လို့မရအောင် လုပ်ထားလို့ *** ဆိုတာမျိုး Asterisk တွေကအစ မပေါ်ပါဘူး။ ကိုယ့်ဘာသာ Password တွေကို မှန်အောင် ရှိက်ပြီး Enter ပဲ ခေါက်လိုက်ပါ။ မကြေခင်မှာပဲ Write Speed/Write Time/records in/out တွေနဲ့ပြီးဆုံးကြောင်း ပြုသပါလိမ့်မယ်။

ဒါဆိုရင်တော့ OS ထည့်ထားတဲ့ SD Card တစ်ခုကို ရရှိပါပြီ။ RPi မှာ SD Card တပ် USB Keyboard Mouse တပ် Display တပ်ပြီးတော့ ပါဝါကြီး ထိုးလိုက်ရင် RPi မှာ OS ကို Setup လုပ်ခိုင်းတဲ့ Screen ကို စမြင်ရပါလိမ့်မယ်။ ဒါဆိုရင် SD Card Setup လုပ်တဲ့အပိုင်းတော့ ပြီးပါပြီ။

မကြေခင် အခြား Advance Step တွေ ရေးပေးပါမယ်။

■ PHP Data Objects (PDO)

by Li Jia Li



about

Li Jia Li

A web addicted who love everything about web technology. Arts and design enthusiast. Core Module developer of RebornCMS.



<http://dragonvirus.com>



/DGVJiaLi



@EyeOfDragonLi



/DragonVirus

PHP Data Objects (PDO) ဆိတ်သာ PHP 5.1 မှာ စပါလာတဲ့ Abstraction Layer တစ်ခုပါ။ အရင် PDO မပါလာခင်ကတော့ mysql သို့မဟုတ် mysqli ကိုသုံးပြီးတော့ mysql_connect() အစရိတ် function တွေကို ခေါ်ပြီး ချက်လေ့ရှိပါတယ်။ ဒါပေမယ့် အဲလိုသုံးတဲ့ အခါမှာ mysql မဟုတ်ဘဲ တဗြား database ကိုသုံးမယ်ဆိုရင် နောက်တဗြား function တစ်ခု ကိုခေါ်ပြီး ပြန်ရေးရတဲ့ အတွက် ပြောင်းလွှာယ်ပြင်လွယ် မရှိပါဘူး။ ဥပမား။ MSSQL ကိုသုံးမယ်ဆိုရင် mssql_connect(), mssql_execute() အစရိသဖို့ ခေါ်သုံးရမှာဖြစ်ပါတယ်။ PDO ကိုသုံးရင်တော့ ဒါမျိုးမလိုတော့ ပါဘူး။ database အားလုံးအတွက်ရေးရမယ့် ပုံစံခြင်းက အတူတူပါပဲ။ တဗြား database ကို သုံးမယ်ဆိုရင် Data source ကို ပြောင်းပေးလိုက်ရှုနဲ့ အားလုံး အဆင်ပြေပါတယ်။ ကိုယ်သုံးမယ့် database အမျိုးအစား အလိုက် php.ini ထဲမှာတော့ driver ကို ကြော်ကြားပေးဖို့လိုပါလိမ့်မယ်။

```
extension=php_pdo_firebird.dll
extension=php_pdo_informix.dll
extension=php_pdo_mssql.dll
extension=php_pdo_mysql.dll
extension=php_pdo_oci.dll
extension=php_pdo_oci8.dll
extension=php_pdo_odbc.dll
extension=php_pdo_pgsql.dll
extension=php_pdo_sqlite.dll
```

အခု ဒီမှာတော့ PDO အသုံးပြုပုံကို MySQL ကို သုံးပြီး ဖော်ပြသွားပါမယ်။

Connecting to the Database

ပထားမည့်အတွက် PDO object ကို create လုပ်ပေးဖို့လိုပါမယ်။ Object ကို create လုပ်တဲ့ အခါမှာ DSN (Data Source Name) လိုခေါ်တဲ့ parameter ကိုထည့်ပေးဖို့လိုပါမယ်။ database username နဲ့ password ဖြစ်တဲ့ ဒုတိယနဲ့ တတိယ parameter ကတော့ optional ပါ။ မထည့်လည့်းရပါတယ်။

```
// connect to mysql database
$conn = new PDO("mysql:host=$host;dbname=$db", $user, $pass);
```

connect လုပ်တဲ့အခါမှာ တကယ်လို့ မရခဲ့ရင် PDO ကို PDOException ကို return ပြန်ပါလိမ့်မယ်။ အဲဒီအတွက် try/catch နဲ့ရေးစေခဲ့ပါတယ်။

```
try {
    $conn = new PDO('mysql:host=localhost;dbname=test', 'username', 'pass');
} catch (PDOException $e) {
    echo "Something wrong. Database connection failed.";
}
```

PDOException အကြောင်းအသေးစိတ်ကိုတော့ နောက်ပိုင်းမှာ ဖော်ပြပေးသွားပါမယ်။ PHP Exception အကြောင်းအသေးစိတ်ကိုတော့ K ရဲ့ PHP Exception မှာသွားဖတ်နိုင်ပါတယ်။

Selecting Data

အခေါ်မှာ PDO Object create လုပ်ခဲ့ ပြီးသွားပြီးဆိုရင်တော့ table ထဲက data select လုပ်ပါမယ်။ သာမန် query ကို လုပ်ဖို့ အတွက် PDO::query() ဆိုတဲ့ method ကို သုံးနိုင်ပါတယ်။ Data ကို select လုပ်လိုက်တဲ့ အချင်မှာ data တွေဆွဲယူဖို့အတွက် PDOStatement ဆိုတဲ့ object တစ်ခုကို create လုပ်ပေးပါလိမ့်မယ်။ ပြီးသွားရင်တော့ PDOStatement::fetch() method ကို သုံးပြီး data တွေကို ပြန်ပြုပါမယ်။ fetch() ကိုသုံးတဲ့ အခါမှာ တစ်ခါခေါ်တိုင်းမှာ row တစ်ခုစီ ထုတ်ပေးပါတယ်။ တကယ်လို့ row အားလုံးကို တစ်ခါတည်း ခေါ်ချင်တယ် ဆိုရင်တော့ fetchAll() ကို သုံးပြီး ခေါ်လို့ရပါတယ်။

```
//connect to database named 'test'  
$conn = new PDO('mysql:host=localhost;dbname=test', 'root', '');  
  
//Query data from table named 'students'  
$query = $conn->query('SELECT name, age FROM students WHERE age=18');  
  
//display results  
while($row = $query->fetch()) {  
    echo $row['name'].' is '.$row['age'].' years old.';  
}  
  
//get all results at once  
$result = $query->fetchAll();
```

fetch() ရော့ fetchAll() ရော့ method နှစ်ခုလုံးမှာ array ဘယ်လိုပုံစံနဲ့ လိုချင်လဲ ဒါမှမဟုတ် object နဲ့လိုချင်လား ကို fetch_style argument ကို ထည့်ပေးပြီး သတ်မှတ်ပေးလို့ရပါတယ်။ fetch_style argument ထည့်ချင်တယ်ဆိုရင် အောက်ကအတိုင်းရေး ပေးရပါမယ်။

```
//get a result with key, value array  
$result = $query->fetch(PDO::FETCH_ASSOC);
```

fetch_style အနေနဲ့က ဝါးမျိုးရှိပါတယ်။ ကိုယ်လိုအပ်သလို ကိုယ့် application နဲ့ အဆင်ပြေတဲ့ ပုံစံကို သုံးနိုင်ပါတယ်။

PDO::FETCH_ASSOC က တော့ key, value array ပုံစံနဲ့ ထုတ်ပြေတာဖြစ်ပါတယ်။

PDO::FETCH_NUM ကလည်း array နဲ့ပြေတာဖြစ်ပါတယ်။ ဒါပေမယ့် သူက key နေရာမှာ field name အစား numeric key အနေနဲ့ပြေတာဖြစ်ပါတယ်။

PDO::FETCH_BOTH ကတော့ field name နဲ့ရော့ numeric key နဲ့ရော့ value တစ်ခုကို နှစ်ခါ ထုတ်ပြေပါတယ်။

PDO::FETCH_CLASS ကတော့ object ပုံစံနဲ့ထုတ်ပြေတာ ဖြစ်ပါတယ်။

Prepared Statements

ဒုန်က query ကို ရေးတဲ့ နေရာမှာ တိုက်ရှိက်ရေးခဲ့ပါတယ်။ အခု တစ်ခါတည်း တန်းပြီး query မလုပ်ပဲ အရင် ပြင်ဆင်ပြီးမှာ query လုပ်ပါမယ်။

```
//connect to database
$conn = new PDO('mysql:host=localhost;dbname=rebornv2b', 'root', '');

//query
$sql = 'SELECT title, body FROM blog WHERE category_id= :category_id';

//prepare query
$prep = $conn->prepare($sql);

//execute query
$prep->execute(array("category_id" => 1));

//get results
$result = $prep->fetch();
```

ဒီမှာ ပထမဆုံး SQL statement ကို prepare() method ထဲကို passing ပေးပြီး PDOStatement ကို အရင်လုပ်လိုက်ပါတယ်။ ပြီးသွားတဲ့ အခါမှ execute() လုပ်ပြီး Query ကို run ပါတယ်။ ပြီးရင်တော့ ထုံးစံအတိုင်း result ကို fetch() နဲ့ယူပါတယ်။ result တစ်ခုထက်ပိုရင်တော့ loop ပတ်ပြီး တစ်ခုခြင်း fetch() လုပ်ပေးဖို့လိုပါမယ်။ ဒီမှာ SQL statement ကို တစ်ချက် ကြည့်ကြည့်လိုက်ပါ။ အဲဒီမှာ ထူးမြားတာ တစ်ခုက category_id = :category_id ဆိုပြီးပေးတားတာ တွေ့ရမှာပါ။ :category_id ဆိုတာက placeholder တစ်ခုပါ။ prepare လုပ်တဲ့ အခါမှာ value ကို တိုက်ရှိက်မပေးဘဲ :category_id ဆိုပြီး placeholder ထည့်ထားပြီး execute လုပ်တဲ့ အခါကျမှာ value ကို array နဲ့ passing ပေးပြီး query လုပ်ပါတယ်။ ဒီလိုလုပ်တဲ့ အတွက် တကယ်လို့ နောက် value တစ်ခုနဲ့ ရှာချင်တယ်ဆိုရင် statement ကို အစအဆုံး ပြန်ရေးစရာ မလိုဘဲ နောက် value တစ်ခုကို passing ပေးပြီး run လိုက်ရှုပါပဲ။

ဒီမှာ (:) semicolon နဲ့ အနောက်က :category_id ဆိုတာ placeholder ကို name တစ်ခုပေးပြီး execute လုပ်တဲ့ အခါမှာလည်း အဲဒီ name နဲ့ပဲ "category_id" => 1 ဆိုပြီး ပြန် assign လုပ်ထားတာတွေ့ရမှာပါ။ တကယ်လို့ နာမည်မပေးချင်ဘူးဆိုရင် (?) ကို သုံးလည်းရပါတယ်။ အောက်က ပုံစံကို တစ်ချက်ကြည့်ကြည့်လိုက်ပါ။

```
//query
$sql = 'SELECT * FROM students WHERE age= ? AND gender= ?';

//prepare query
$prep = $conn->prepare($sql);

//execute query
$prep->execute(array(20, 'female'));

//get results
$result = $prep->fetch();
```

Placeholder နေရာမှာ (?) ထားပြီး execute လုပ်တဲ့ အခါမှာ အစဉ်လိုက်အတိုင်း array နဲ့ passing ပေးရပါတယ်။ ဒီလိုရေးတဲ့ အခါကျရင် ရေးရတာ နည်းပေါ်မယ့် passing ပေးရမယ့် value တွေများလာရင် နည်းနည်းတော့ ရူပ်သွားနိုင်ပါတယ်။ name နဲ့ ပေးတာကတော့ ပိုပြီးကောင်းပါတယ်။ name နဲ့ ပေးတာက ပိုပြီး maintainable ဖြစ်သလို ကိုယ့်ရဲ့ code ကို နောက်လူ ကြည့်မယ်ဆိုလည်း နားလည်လွယ်တာပေါ့။

Binding Values and Parameters

အပေါ်မှာ execute() လုပ်တဲ့ အခါမှာ value တွေကို passing ပေးခဲ့ပါတယ်။ အခု တစ်ခါမှာတော့ PDOStatement မှာ execute မလုပ်ခင် Value ကို bind လုပ်ပေးလိုက်ပါမယ်။ ဒါဆိုရင် bindValue ကို မပြောင်းမခြင်း execute လုပ်တဲ့ အခါမှာ နောက်ဆုံး bind လုပ်ခဲ့တဲ့ value တွေနဲ့ အလုပ်လုပ်မှာ ဖြစ်ပါတယ်။ အဲအတွက်ဘာအားသာချက်ရှိလဲဆိုတော့ တကယ်လို့ ကိုယ်က နောက် တစ်ခါ execute လုပ်တဲ့ အခါမှာ value တစ်ခုပဲပြောင်းဖို့လိုတယ်ဆိုရင် အဲဒီ value ကိုပဲ bind ထပ်လုပ်ပေးလိုက်ရှုပါပဲ။ value အားလုံး ကို ထပ်ပြီး passing ပေးနေစရာ မလိုတော့ပါဘူး။

```
//query
$sql = 'SELECT * FROM students WHERE age= :age AND gender= :gender';

//prepare query
$prep = $conn->prepare($sql);

$prep->bindValue(':age', 20);

$prep->bindValue(':gender', 'female');
$prep->execute(); //get female students who is age of 20.
$femaleStudents = $prep->fetchAll();

$prep->bindValue(':gender', 'male');
$prep->execute(); //get male students who is age of 20.
$maleStudents = $prep->fetchAll();
```

ဒီမှာ value တင်မဟုတ်ပါဘူး parameter ပါ bind လုပ်ပေးလိုရပါတယ်။ parameter ကို placeholder အနေနဲ့ အလုပ်လုပ်တာဖြစ်ပြီး execute မလုပ်ခင်မှာ variable ကို assign လုပ်ပေးလိုက်ရင်ရပါတယ်။ ရင်းသွားအောင် အောက်က example လေးကို ကြည့်ကြည့်လိုက်ပါ။

```
//query
$sql = 'SELECT * FROM students WHERE age= :age AND gender= :gender';

//prepare query
$prep = $conn->prepare($sql);

$prep->bindValue(':age', 20);

$prep->bindParam(':gender', $gender);
```

```

$gender = 'female';
$prep->execute(); //get female students who is age of 20.
$femaleStudents = $prep->fetchAll();

$gender = 'male';
$prep->execute(); //get male students who is age of 20.
$maleStudents = $prep->fetchAll();

```

Inserting, Updating & Deleting Data

Insert, update, delete ကလည်း select နဲ့တူတူပါပဲ။ ဘာလိုလဲဆိုတော့ statement ကို prepare လုပ်တယ်။ ပြီးရင် execute လုပ်တယ်။ ဒီတိုင်းပါပဲ။ ဒါကြောင့် တစ်ခုချင်း အသေးစိတ်ထပ်မပြောတော့ပါဘူး။ ဒေါက်က Insert လုပ်တဲ့ example ကို တစ်ခုက်လောက် ကြည့်ကြည့်လိုက်ပါ။

```

//connect to database
$conn = new PDO('mysql:host=localhost;dbname=rebornv2b', 'root', '');

//SQL Statement
$sql = 'INSERT INTO students (name, age, gender, enrolled_date) VALUES (:name, :age, :gender, NOW())';

$prep = $conn->prepare($sql);

$prep->execute(array(
    ':name'      => 'Mg Mg',
    ':age'        => 20,
    ':gender'     => 'male'
));

echo "A new student ID ". $conn->lastInsertId() .'is just enrolled';

echo $prep->rowCount(). " student(s) is/are just enrolled.";

```

ဒီမှာခါတိုင်းလိုပဲ statement ကို prepare လုပ်တယ်။ ပြီးတော့ execute လုပ်တယ်။ ထူးခြားတာက PDO မှာ insert လုပ်ပြီးရင် insert လုပ်လိုက်တဲ့ ID ကိုသိချင်ရင် PDO Class ရဲ့ lastInsertId() ဆိုတာကို ခေါ်ပြီးသိနိုင်ပါတယ်။ ဒါပေမယ့် ID column က auto_increment ဖြစ်ဖို့တော့လိုပါလိမ့်မယ်။

နောက်ထပ် အသုံးဝင်မယ့် method တစ်ခုကတော့ PDOStatement ရဲ့ rowCount() ပဲဖြစ်ပါတယ်။ rowCount() က insert, update, delete အတွက် ဆိုရင် query လုပ်လိုက်လို့ သက်ရောက်သွားတဲ့ row အရေအတွက်ကို ပြုမှာဖြစ်ပြီး select ဆိုရင်တော့ ထွက်လာတဲ့ row အရေအတွက်ကို ပြုမှာဖြစ်ပါတယ်။ ဒါပေမယ့် select statement အတွက် က database အမျိုးအစားအားလုံးမှာ အလုပ်မလုပ်ပါဘူး။

Error Handling in PDO

ရှေ့မှာ database connection လုပ်တွန်းက Error Handling အတွက် Try Catch ကို သုံးခဲ့တာ မှတ်မီမှာပါ။ အဲလို့
prepare, execute, fetch အဆင့်တွေမှာလည်း error handle လုပ်သင့်ပါတယ်။ အောက်က code ကို အရင် ကြည့်ကြည့်လိုက်ပါ။

```
//connect to database
try {
    $conn = new PDO('mysql:host=localhost;dbname=rebornv2b', 'root', '');
} catch (PDOException $e) {
    echo "Couldn't connect to database";
    exit;
}

//query
$sql = 'SELECT * FROM students WHERE age= :age';

try {
    //prepare query
    $prep = $conn->prepare($sql);

    if ($prep) {
        //Query
        $result = $prep->execute(array('age' => 20));
    }

    if ($result) {
        $students = $prep->fetchAll();
    } else {
        $error = $prep->errorInfo();
        echo "Failed to perform the query with message : ". $error[2];
    }
} catch (PDOException $e) {
    echo "A database problem has occurred : " . $e->getMessage();
}
```

ပထားမည့်အား database connection မှာ error ကို try catch နဲ့ ဖမ်းပြီး error ပြပါတယ်။ ခုတိယ prepare လုပ်တဲ့
အဆင့် မှာလည်း ဒီလိုပဲ try catch နဲ့ ထပ်စစ်ပါတယ်။ ဒါပေမယ့် သူက fail ဖြစ်တဲ့ အခါကျရင် throw Exception လုပ်သလို
false လည်းပြန်လာတတ်ပါတယ်။ ဒါကြောင့် အောက်မှာ if နဲ့ ထပ်စစ် ပြီး မှ execute လုပ်ပါတယ်။ ဒီမှာ အောက် ဆုံးကို နည်းနည်း
ကျကျကြည့်လိုက်ပါ။ အဲဒီမှာ Database problem ဖြစ်ကြောင်း error ပြီး သူ့ရဲ့ Exception မှပါဘာတဲ့ error message ကို \$e->
getMessage() ဆိုပြီးပြပါတယ်။ တတိယ အနေနဲ့ execute လုပ်တဲ့ အဆင့်မှာတော့ သူက ရှင်းပါတယ်။ success ဖြစ်ရင် true ပြန်ပါတယ်။ fail
ဖြစ်ရင် false ပြန်ပါတယ်။ ဒါကြောင့် if နဲ့ ပဲစစ်ပြီး false လာရင် error ပြပေးလိုက်ပါတယ်။ ဒီမှာ \$prep->errorInfo() ကို ခေါ်ထားပါတယ်။
ဒါကတော့ PDOStatement Class ထဲက method တစ်ခုပါ။ သူ့ကို ခေါ်လိုက် ရင် array နဲ့ value ၃ ခုပါလာပါတယ်။

1. SQLSTATE error code (ANSI SQL Standard)

2. Driver Specific error code

3. Driver Specific error message

ဒါကြောင့် message ပြတဲ့နေရာမှာ \$error[2] ဆိုပြီး ခေါ်သုံးထားပါတယ်။

Fetchလုပ်တဲ့ အဆင့် မှာ တော့ တကယ်လို့ fetch() method ကို ခေါ်တဲ့ အခါမှာ တစ်ခုခုမှားရင်တော့ false ပြန်ပေးမှာဖြစ်ပါတယ်။ ဒါပေမယ့် တစ်ခါတစ်ရုံ မှာ Data Fetching Mode ပေါ်မှုတည်ပြီး empty array လည်းပြန်တတ်ပါတယ်။ ဆိုလိုတာတော့ record မရှိဘူးပေါ့။

ဒီလောက်ဆိုရင်တော့ PDOအကြောင်းသိသင့်သလောက်သိသွားပြီလို့ထင်ပါတယ်။ တကယ်လို့အသေးစိတ်သိခင်ရင်တော့ ထုံးစံအတိုင်း PHP Documentation သာ သွားဖတ်လိုက်ပါ။

■ Ajax file upload

by Yan Naing



about

Yan Naing

All I know is I know nothing.



www.myanmarlinks.net



/



@Mr_YanNaing



/Mr-Zer0

ဒီတစ်လတော့ {Echo} ပရိတ်သတ်ကြီးအတွက် ပေါ်ပေါ်ပါးပါး နဲ့ fancy ဖြစ်မယ့် Tutorial လေးတစ်ခုကို ရေးသား တင်ဆက်လိုက်ပါတယ်။ Ajax file upload အကြောင်းလေးပါ။ သိချင်ပြီး မသိသေးတဲ့သူတွေကို အစိကရည်ရွယ်ပါတယ်။

ထုံးစံအတိုင်း Sublime Text ကို ဖွင့်ပြီး Code စရေးပါမယ်။ ပထားဆုံး အဆင့်အနေနဲ့ Upload form ကို အောက်မှာ ပြထားတဲ့အတိုင်း ရေးလိုက်ပြီး index.html ဆိုတဲ့ နာမည်နဲ့ Save လိုက်ပါတယ်။

```
<!doctype html>
<html lang="en">
<head>
    <title>Ajax Upload</title>

    <meta charset="UTF-8">
    <meta name='robots' content='INDEX, FOLLOW' />
    <meta name='viewport' content='width=1024' />
    <meta name='description' content='Ajax upload tutorial' />
    <meta name='keywords' content='Ajax Upload, ajax, upload' />
</head>
<body>
    <form name='upload' id='upload' action='upload.php'>
        <input type="file" multiple name="files[]" id="file">
        <input type="submit" value="Upload" id="submit">
    </form> <!-- end of upload -->
</body>
</html>
```

ပြထားတဲ့ Code ထွေကို localhost ထဲမှာ ajax-upload ဆိုတဲ့ နာမည်နဲ့ folder တစ်ခုထောက်ပြီး သိမ်းထားတာပါ။ ဒါဆိုရင် Browser ကေန <http://localhost/ajax-upload> လို့ခေါ်လိုက်ရင် Upload Field တစ်ခုနဲ့ Upload ဆိုတဲ့ Button တစ်ခုကို တွေ့ရမှာပါ။

Code ကို ကြည့်လိုက်မယ်ဆိုရင် Form ရဲ့ action မှာ upload.php ဆိုပြီး ရေးထားတာကို တွေ့မှာပါ။ ဒါဆိုရင် အဲဒီ index.html ရေးထားတဲ့ directory (ajax-upload) ထဲမှာပဲ upload.php ဆိုတဲ့ file တစ်ခု ရှိမှုရမှာပေါ့။ upload.php ကို တည်ဆောက်လိုက်ပါပြီး အရင်ဆုံး Upload တင်လိုက်ရင် ဒီ upload.php ဆိုတဲ့ file ထဲကို ရောက်မရောက် ကြည့်ပါမယ်။ ဘယ်လိုကြည့်မလဲဆိုတော့ var_dump() ပေါ့များ။

```
<?php
var_dump('Testing');
```

ဒီ code ထွေကို upload.php ထဲမှာ ထည့်ရေးလိုက်ပါတယ်။ ပြီးတော့ ခုနက ခေါ်ထားတဲ့ Browser ထဲက page မှာ upload button ကို နိုင်လိုက်ရင် var_dump() ထုတ်ထားတဲ့ Testing ဆိုတာ ထွက်လာပါတယ်။ ဒါဆိုရင်တော့ စမ်းသပ်မှု အောင်မြင်ပြီးပေါ့။

နောက်တစ်ဆင့် ထပ်တက်ပါမယ်။ ခုနါက html ထဲမှာ ရေးခဲ့တဲ့ form က input type text တွေ textarea တွေအတွက်ဆိုရင် ကောင်းကောင်းအလုပ်လုပ်ပါလိမ့်မယ်။ ဒါပေမယ့် input type file အတွက်တော့ အမျိန်တက်သေးပါဘူး။ ဒီအကြောင်းကို {Echo} ခြောက်လပြည့် {Echo} august 2013 မှာ PHP and Media ဆိုတဲ့ Post ထဲမှာ ဖော်ပြထားတာ ရှိပါတယ်။ ဒီတော့ ဒီ Form ကို Upload form ဖြစ်ဖို့အတွက် အောက်မှာ ပြထားတဲ့အတိုင်း enctype ကို ထပ်ဖြည့်ပေးလိုက်ပါတယ်။

```
<form name="upload" id="upload" action="upload.php" enctype="multipart/form-data"
method='POST'>
```

အခုခံရင်တော့ ဒီ Form က မျိန်မျိုး ကန်ကန် အလုပ်လုပ်သွားပြီလို ထင်ပါတယ်။ ဟုတ်မဟုတ်သိရအောင် ခါတိုင်း လုပ်နေကြအတိုင်း var_dump () ထုတ်ကြည့်လိုက်ပါမယ်။ ဒီတစ်ခါ ဘာကို var_dump() ထုတ်မလဲဆိုတော့ \$_FILES ဆိုတာကို ထုတ်ကြည့်ပါမယ်။ \$_FILES ဆိုတာက HTTP POST method နဲ့လာတဲ့ input type file ထဲက data တွေကို store လုပ်ထားတဲ့ PHP ရဲ့ Global variable ပါ။ သူတဲ့မှာ ဘာတွေရှိလဲဆိုတာကို ကြည့်လိုက်ရင် ပိုပြီး နားလည် သဘောပေါက်သွားမှာပါ။

upload.php ထဲမှာ ဒါမျိုးလေး ရေးပြီး var_dump() ထုတ်လိုက်ပါတယ်။

```
var_dump($_FILES);
```

ပြီးတော့ Browser မှာ ခုနါကအတိုင်း <http://localhost/ajax-upload> ဆိုတာကို ခေါ်လိုက်ပါတယ်။ ပြီးတော့ File field ကို နိုပ်ပြီး ကွန်ပျိုးတာထဲမှာ ရှိတဲ့ file တစ်ခုကိုရွေးလိုက်ပါတယ်။ Upload Button ကို နိုပ်လိုက်ပါတယ်။ မျိန်တယ်ဆိုရင်တော့ အောက်မှာ ပြထားတဲ့အတိုင်း ပေါ်လာပါမယ်။

```
array (size=1)
  'files' =>
    array (size=5)
      'name' => string '7822615_d279_625x1000.jpg' (length=25)
      'type' => string 'image/jpeg' (length=10)
      'tmp_name' => string 'C:\xampp\tmp\phpDDB4.tmp' (length=24)
      'error' => int 0
      'size' => int 193147
```

ဒီ Tutorial ကို လိုက်လုပ်ကြည့်ရင် တချို့ ကွန်ပျိုးတာတွေမှာက ပေါ်တဲ့ ပုံစံ နည်းနည်း ကွဲနိုင်ပါတယ်။ အောက်မှာ ပြထားတဲ့ ပုံစံအတိုင်း ထွက်လာရင်လည်း မှန်ပါတယ်။

```
array(1) { ["files"]=> array(5) { ["name"]=> string(25) "7822615_d279_625x1000.jpg" ["type"]=> string(10) "image/jpeg" ["tmp_name"]=> string(24) "C:\xampp\tmp\php5581.tmp"
["error"]=> int(0) ["size"]=> int(193147) } }
```

ဒီနေရာမှာ မှာနိုင်တဲ့ အခြေအနေသိပ်မရှိပါဘူး။ ဒါပေမယ့် enctype ကိုတော့ ဂရိစိုက်ပြီး ထည့်ပေးဖို့ လိုပါမယ်။ ဒီလိမ့်မဟုတ်ရင် var_dump() မှာ ဘာမှ ပေါ်လာမှာ မဟုတ်ပါဘူး။

var_dump() ထုတ်ထားတဲ့ဟာကို ကြည့်မယ်ဆိုရင် အပေါ်ဆုံးက files ဆိုတဲ့ array key က input field ရဲ့ name ပါ။ နောက် အဲဒီ array ထဲမှာ ပါတဲ့ name, type နဲ့ size ကတော့ သိသာပါတယ်၊ ကိုယ့်ရွေးလိုက်တဲ့ file ရဲ့ name, type နဲ့ size ပါ။ tmp_name ဆိုတာကိုတော့ နည်းနည်းရှင်းပြန့် လိုမယ်ထင်ပါတယ်။ PHP က file တွေ upload တင်တဲ့အခါမှာ အဲဒီ file ကို temp file အနေနဲ့ သူရဲ့ temp file ထားဖို့ သတ်မှတ်ထားတဲ့ နေရာမှာ အရင် တင်လိုက်ပါတယ်။ ပြီးမှ အဲဒီ temp file ကို ကိုယ့်ရွေးချင်တဲ့နေရာကို ပြန်ရွေးပေးရပါတယ်။ error ကတော့ file နဲ့ပတ်သက်တဲ့ error ဖြစ်ပါတယ်။ ဒီနေရာမှာ ပြောမယ့် error က လေးခုရှိပါတယ်။

- UPLOAD_ERR_OK = 0
- UPLOAD_ERR_INI_SIZE = 1
- UPLOAD_ERR_FORM_SIZE = 2
- UPLOAD_ERR_PARTIAL = 3
- UPLOAD_ERR_NO_FILE = 4

ပထမဆုံး တစ်ခုကတော့ error မရှိတဲ့အကြောင်း ပြောတာပါ။ UPLOAD_ERR_INI_SIZE ကတော့ php.ini ထဲမှာ ကြော်ထားတဲ့ maximum upload file size ထက်ကြီးတဲ့ file ကို တင်လိုက်ရင် တက်မယ့် error ပါ။ နောက်တစ်ခု UPLOAD_ERR_FORM_SIZE ကတော့ form မှာ တင်ခွင့်ပေးမယ့် file size ကို သတ်မှတ်ပေးထားရင် အဲဒီထက် file size ကြီးတဲ့ file တွေကို ဒီ error ပြောပါ။ file တွေကို upload တင်ရင်းနဲ့ ပျက်သွားတာမျိုး file တစ်ဝက်တပျက်ပဲ upload ဖြစ်သွားတာမျိုးဆိုရင် UPLOAD_ERR_PARTIAL ဆိုတဲ့ error ကို ပြပါလိမ့်မယ်။ နောက်ဆုံးတစ်ခုကတော့ file upload မဖြစ်သွားရင် ပြောမယ့် error ပါ။

ဒီလောက်သိရှိပါဆိုတော့ Upload တင်လိုပါပြီ။ upload တင်မယ်ဆိုတော့ ပထမဆုံး လုပ်ဖို့ လိုအပ်တာက ကိုယ်တင်မယ့် file ကို ထားမယ့် နေရာကို သတ်မှတ်ဖို့ပါ။ ဒါတော့ file ဆိုတဲ့ directory တစ်ခုကို အဲဒီ ajax-upload ထဲမှာ ဆောက်လိုက်ပါတယ်။ upload file တွေကို အဲဒီထဲမှာ တင်ပါမယ်။ Upload တင်ဖို့က ဘာမှသိပ်ပြီး ခက်ခက်ခဲ့ လုပ်စရာမလိုပါဘူး။ move_uploaded_file() ဆိုတဲ့ function တစ်ခုသုံးရှုံး အလုပ်ဖြစ်ပါတယ်။ Code ကို တစ်ချက်လောက် ကြည့်လိုက်ပါပြီး။

```
$path = __DIR__ . '/file/';  
  
$tmp_name = $_FILES['files']['tmp_name'];  
$name = $_FILES['files']['name'];  
  
move_uploaded_file($tmp_name, $path.$name);
```

upload.php ထဲမှာ ဒီ code တွေရေးတော့ upload တင်လိုက်ပါတယ်။ အဆင်ပြေပြန့် ကိုယ်တင်ချင်တဲ့ file ကို file ဆိုတဲ့ directory ထဲကို တင်လိုက်နိုင်ပြီ။ ဒါဆိုရင် code ထဲမှာ ဘာတွေရေးထားသလဲ ဆိုတာကို ကြည့်လိုက်ပါမယ်။ အပေါ်မှာ ပြောခဲ့သလိုပဲ \$path ဆိုတဲ့ variable ထဲမှာ uploaded file တွေကို ထားပယ့် နေရာကို ကြော်ပေးလိုက်ပါတယ်။ move_uploaded_file() ဆိုတဲ့ function ကို php documentation ထဲမှာ ကြည့်လိုက်မယ်ဆိုရင် parameter နှစ်ခု ထည့်ပေးရပါတယ်။ ပထမ Parameter က tmp name ပါ။ နောက်တစ်ခုကတော့ ကိုယ့်ရဲ့ filename ကို ထားချင်တဲ့နေရာ နေရာ အပြည့်အစုံနဲ့ ထည့်ပေးရမှာပါ။ သဘောကတော့ php ရဲ့ tmp file ကို ကိုယ်ကြိုက်တဲ့နာမည်နဲ့ ရွှေ့ချင်တဲ့နေရာကို ပြန်ရွေ့လိုက်တဲ့ သဘောပါပဲ။

ဒီလောက်ဆိုရင် Upload တင်လိုက်သွားပြီး။ ဒါပေမယ့် error တက်လာရင် ဘယ်လိုလုပ်မလဲ။ ဟုတ်တာဖော့၊ error ကိစ္စကို ဖြေရှင်းရနိုင်းမယ်။ လွယ်ပါတယ်။ Switch case ကိုသုံးပြီး စစ်လိုက်မယ်။ အောက်မှာရေးထားတဲ့အတိုင်းလေးပေါ့ဖြူ။

```
$file = $_FILES['files'];

switch ($file['error']) {
    case 1:
        echo 'File size is larger than limitation of php.ini -_-';

        break;

    case 2:
        echo 'File size is larget than limitation of form. -_-';

        break;

    case 3:
        echo 'File is uploaded partially. -_-';

        break;

    case 4:
        echo 'There is no file to upload. -_-';

        break;

    default:
        $path = __DIR__.'/file/';

        $tmp_name = $file['tmp_name'];
        $name = $file['name'];

        move_uploaded_file($tmp_name, $path.$name);

        echo 'Successfully uploaded \\\o/ <br/><br/>';
        echo 'Name : ' . $file['name'] . '<br/>';
        echo 'Size : ' . $file['size'] . ' bites <br/>';
        echo 'Type : ' . $file['type'] . '<br/>';

        break;
}
```

ဒဲ error တက်လောက်မယ့် file လေးတွေ တင်ကြည့်ပါ့။ case ထဲမှာ ရေးထားတဲ့ 1, 2, 3, 4 တွေက အပေါ်မှာ ပြောထားတဲ့ error code နံပတ်တွေပါ။ default မှာက error မရှိရင် upload ဖြစ်သွားအောင် ရေးထားတာပါ။ upload ဖြစ်သွားတဲ့ file ရဲ့ name ရယ် size ရယ် type ရယ်ကို echo ထုတ်ပြီး ပြန်ပြထားပါတယ်။ ကြည့်ရတာ နည်းနည်း အဆင်ပြေအောင် br tag တွေနဲ့ ချားထားတာပါ။

Upload တင်တဲ့ Process က ပြီးသွားပြီ ဆိုပေမယ့ သိပ်တော့ မကျေနပ်သေးဘူး။ နည်းနည်းလောက် အထွန် တက်လိုက်ချင်သေးတယ်။ Multi upload တင်ချင်ပါပြီတဲ့ ဘယ်လိုလုပ်ကြမလဲ။ ဒါဆိုရင် Multi upload ရဖို့အတွက် ဆက်ပြီး ရေးကြတာပေါ့။

Multi upload ရဖို့ Form ထဲက input type file ကနေ အရင်ဆုံး စပြီးပြင်ရပါမယ်။ နည်းနည်း သိသာသွားအောင် မပြင်ခင်မှာ browser က file တင်တဲ့ box ကို နိုင်ပြီး ကွန်ပျူးတာထဲက file ကို ရွေးကြည့်ပါမယ်။ အဲလိုရွေးတဲ့အခါမှာ ctrl + click နဲ့ multi ရွေးကြည့်လိုက်ပါမယ်။ ရွေးလိုမရဘူးဟု။ နောက်တစ်ခုရွေးလိုက်ရင် အရင် တစ်ခုက deselect ဖြစ်သွားတယ်။ ဒါတော့ multi upload တင်လိုမရတာမဟုတ်ဘူး ရွေးလိုကို မရတာပေါ့။ ဒါဆို ရွေးလိုရအောင် လုပ်ကြည့်မယ်။ index.html က input type file ရဲ့ name ကို [] ဒါလေးပိုထည့်ပေးလိုက်ပါတယ်။ ပြီးတော့ multiple ဆိုတာလေးပါထည့်ပေးလိုက်တယ်။

```
<input type="file" multiple name="files[]" id="file">
```

ပြီးရင် save ပြီးတော့ ခုနအတိုင်းပဲ file ပြန်ရွေးကြည့်ပါတယ်။ ရသွားပြီ။ Multi ရွေးလိုရသွားပြီဗျာ။ ထည့်ရေးပေးလိုက်တဲ့ multiple ဆိုတာက file ရွေးတဲ့အခါမှာ multiple ရဖို့ပါ။ name မှာ လေးထောင့်ကွင်း ထည့်ပေးတာကတော့ multiple ရွေးလိုက်တဲ့ file ထွေကို array အနေနဲ့ POST လုပ်လိုရအောင် ဖြစ်ပါတယ်။ အခု multiple ရွေးလိုရပြီဆိုပြီး upload button ကို နှိပ်ပစ်လိုက်ရင်တော့ php ရဲ့ချစ်ဖွယ်သော error ထွေကို မြင်ရမှာ ဖြစ်ပါတယ်။

ဘာလိုပဲဆိုတာကို သိရအောင် \$_FILES ဆိုတဲ့ variable ကို ပြန်ပြီး var_dump() လုပ်ကြည့်ပါမယ်။

```
array (size=1)
  'files' =>
    array (size=5)
      'name' =>
        array (size=2)
          0 => string '4.jpg' (length=5)
          1 => string '53.386.jpg' (length=10)
      'type' =>
        array (size=2)
          0 => string 'image/jpeg' (length=10)
          1 => string 'image/jpeg' (length=10)
      'tmp_name' =>
        array (size=2)
          0 => string 'C:\xampp\tmp\phpDB2A.tmp' (length=24)
          1 => string 'C:\xampp\tmp\phpDB3A.tmp' (length=24)
      'error' =>
        array (size=2)
          0 => int 0
          1 => int 0
      'size' =>
        array (size=2)
          0 => int 372891
          1 => int 346990
```

မြင်တဲ့အတိုင်းပါပဲ။ name ထွေ size ထွေ type ထွေ အားလုံးက Array ထွေ ဖြစ်နေပါတယ်။ ဒါဆိုရင် ဒီ array အတိုင်း အဆင်ပြေအောင် ပြန်လိုက်ရေးပေးရမယ့် သဘောပြေ့။ ကိုယ်က ဒီလို လိုချင်တယ်ဆိုမှတော့ ဒီလောက်တော့ လုပ်ရမှာပြေ့။ ဒါပေမယ့် သိပ်ခက်ခဲတဲ့ ကိစ္စတော့မဟုတ်ပါဘူး။ အောက်မှာ ရေးထားတဲ့အတိုင်း foreach နဲ့ loop ပတ်ပြီး ရေးလိုက်တာပြေ့။

```
<?php

$file = $_FILES['files'];

foreach ($file['error'] as $key => $value) {

    switch ($file['error'][$key]) {
        case 1:
            echo 'File size is larger than limitation of php.ini -_-';
            break;

        case 2:
            echo 'File size is largert than limitation of form. -_-';
            break;

        case 3:
            echo 'File is uploaded partially. -_-';
            break;

        case 4:
            echo 'There is no file to upload. -_-';
            break;

        default:
            $path = __DIR__.'/file/';

            $tmp_name = $file['tmp_name'][$key];
            $name = $file['name'][$key];

            move_uploaded_file($tmp_name, $path.$name);

            echo 'Successfully uploaded \\\o/ <br/><br/>';
            echo 'Name : ' . $file['name'][$key] . '<br/>';
            echo 'Size : ' . $file['size'][$key] . ' bites <br/>';
            echo 'Type : ' . $file['type'][$key] . '<br/>';
            echo '======<br/>';

            break;
    }
}
```

ဒါပါပဲ။ file ကို multiple ရွေးပြီး upload တင်ကြည့်လိုက်ပါဉိုး။ {Echo} ပရိတ်သတ်ကြီး အနေနဲ့ ဒီလိုစဉ်းစားမိချင်စဉ်းစားမိမယ်ဖူ။ ဒါက multi ရွေးပြီး upload တင်လိုက်လိုကို။ တစ် file တည်း ရွေးပြီး upload တင်ရင်ရော ခန်ကအတိုင်း ပြန်ပြင်ရေးမှ အဆင် ပြောမှ မဟုတ်ဘူးလားလို့လေ။ တစ် file တည်းတင်လည်း အဆင်ပြောပါတယ်ဖူ။

သိပ်ပြီး အရေးမပါလှပေမယ့် သိသင့်တယ် ထင်လို့ ပြောရည်းမယ်ဖူ။ အခါ multi upload တင်တဲ့အတွက် foreach loop ပတ်ထားတာကို ကြည့်လိုက်မယ်ဆိုရင် \$file ထဲက error ကို loop ပတ်ထားပါတယ်။ \$file['error'] ကို loop ပတ်လို့ရသလို \$file['name'] ကိုလည်း loop ပတ်လို့ရပါတယ်။ စဉ်းစားကြည့်ရင် ဒါကြီးက သိပ်တော့ သဘာဝ မကျေဘူးဖြစ်နေတယ်။ တော်း developer တွေကတော့ အဲဒီ array ပုံစံကို ဖုန်္ဂြိုက်လို့ ကိုယ့်ပိုင် ပုံစံ ပြန်ပြောင်းကြလေ့ရှုပါတယ်။ အောက်မှာ ပြထားတဲ့ ပုံစံအတိုင်းပေါ့။

```
array (size=1)
'files' =>
array (size=2)
0 =>
array (size=5)
  'name' => string 'localhost testings phpandmedia upload.php.png' (length=45)
  'type' => string 'image/png' (length=9)
  'tmp_name' => string 'C:\xampp\tmp\phpBCA2.tmp' (length=24)
  'error' => int 0
  'size' => int 10856
1 =>
array (size=5)
  'name' => string 'localhost testings phpandmedia upload.php3.png' (length=46)
  'type' => string 'image/png' (length=9)
  'tmp_name' => string 'C:\xampp\tmp\phpBCA3.tmp' (length=24)
  'error' => int 0
  'size' => int 23501
```

ဒါဆိုရင် files ကို foreach ပတ်လိုက်တာနဲ့ အဆင်ပြောပြီး၊ ဒီအတိုင်းတော့ ပြန်ပြောင်းပြီး မရေးတော့ပါဘူး။ လက်ရှုပုံစံကလည်း အဆင်ပြောနေတာကိုး။

Php ကိုသုံးပြီး multi file upload လုပ်တာတော့ ကောင်းကောင်း ရေးတတ်သွားပြီပေါ့။ ဒါဆို နောက်တစ်ဆင့် တက်ကြမယ်လေး။ Ajax ကိုသုံးပြီး upload မယ်။ plain javascript မှာ ajax ကို ရေးလို့ရပါတယ်။ တော်း javascript framework တွေ javascript library တွေမှာလည်း ajax ကို ပိုပြီး လွယ်လွယ်ကူကူ သုံးလို့ရအောင် လုပ်ပေးထားတာတွေ ရှုပါတယ်။ ဒီမှာတော့ လူသီများတဲ့ jQuery library ကိုပဲသုံးပြီး ရေးပါမယ်။

ရေးနေကျအတိုင်းပဲ html ထဲမှာ jQuery ကို ချိတ်ပါမယ်။ ပြီးတော့ script နည်းနည်းလည်း ရေးရည်းမယ်။ Upload button ကို နိုင်လိုက်ရင် Ajax နဲ့ Upload ဖြစ်သွားဖို့လေး။ Upload button ကိုလည်း jQuery ကနေ handle လုပ်ရတာ လွယ်ကူအောင် id ပေးလိုက်ပါမယ်။

```
<input type="submit" value="Upload" id="submit">
```

class တွေ id တွေ မပါဘဲလည်း handle လုပ်လိုရပေမယ့် class ဒါမှမဟုတ် id ပေးထားတာက ပိုပြီး handle လုပ်ရတာလွယ်ကူတာပေါ့။ ပြီးတော့ ဆက်လုပ်ရမှာက Upload Button ကို နိုင်လိုက်ရင် အခါစိုး upload.php ကိုသွားတယ်။ ajax နဲ့ upload တင်ချင်တာဆိုတော့ Upload button ကို နိုင်လိုက်ရင် အဲလို Link သွားလိုမဖြစ်ဘူး။ ဒီတော့ အဲဒီ Upload button ကို သူ့အလုပ် သူမလုပ်နိုင်အောင် တားရမယ်။ အဲလို တားဖို့အတွက် အောက်မှာ ပြထားတဲ့အတိုင်း Script နည်းနည်း ရေးလိုက်ပါတယ်။

```
<script src="http://localhost/ajax-upload/jquery-1.10.2.min.js"></script>
<script>
    $('#submit').bind('click', function(e){
        e.preventDefault();

        alert('Click tel ha');
    });
</script>
```

ဒီအတိုင်း ရေးပြီးတော့ Upload button ကို နိုင်ကြည့်လိုက်ရင် Click tel ha ဆိုတဲ့ Alert box တစ်ခုတက်လာပါမယ်။ upload.php ကို ရောက်မသွားဘူး။ ဒါဆိုရင် Button ရဲ့ လုပ်ငန်းစဉ်ကို ဖျက်ဆီးတဲ့ ကိစ္စအောင်မြင်သွားပြီ။ Script မှာ preventDefault() ဆိုတဲ့ function က button ကို သူ့အလုပ်သူမလုပ်နိုင်အောင် တားလိုက်တာပါ။ e ကတော့ event ကိုဆိုလိုတာပါ။ Button ကို နိုင်လိုက်တယ်ဆိုတဲ့ Event ပါ။ နောက်တစ်ကြောင်းက alert ကတော့ Button ကို နိုင်လိုက်တဲ့အခါမှာ ဒီ Script ထဲကို ရောက်မရောက် သိသာအောင် ရေးလိုက်တာပါ။ ခုနှင့် Click tel ha ဆိုတဲ့ Alert box က သူ့ကြောင့် ပေါ်တယ်ဆိုတာ ရိုပ်မိမှာပါ။

နောက်တစ်ဆင့်အနေနဲ့ Upload button ကိုနိုင်လိုက်ရင် input type file ထဲမှာ ရွေးထားတဲ့ file တွေကို ပြန်ယူကြည့်မယ်။ အဲဒါတွေကို Ajax နဲ့ upload တင်မှာကို။

```
var myFile = document.getElementById('file');
console.log(myFile.files);
```

ခုနှင့် alert function ကို ဖျက်ပြီး အပေါ်ကအတိုင်း ရေးလိုက်ပါတယ်။ ပြီးတော့ Browser မှာ javascript console ကို ဖွင့်တယ်။ Form မှာ file နှစ်ခုလောက်ရွေးပြီး Upload button ကို နိုင်လိုက်တဲ့အခါ

```

▼ FileList {0: File, 1: File, length: 2, item: function} ⓘ
  ▼ 0: File
    ► lastModifiedDate: Mon May 06 2013 18:21:42 GMT+0630 (Myanmar Standard Time)
      name: "21103_563875986966354_1537063148_n.jpg"
      size: 18231
      type: "image/jpeg"
      webkitRelativePath: ""
    ► __proto__: File
  ▼ 1: File
    ► lastModifiedDate: Thu Oct 04 2012 16:10:03 GMT+0630 (Myanmar Standard Time)
      name: "54154.jpg"
      size: 327106
      type: "image/jpeg"
      webkitRelativePath: ""
    ► __proto__: File
  length: 2
  ► __proto__: FileList

```

Chrome browser ရဲ့ console မှာ အဲလို ပေါ်ပါတယ်။ ရွေးလိုက်တဲ့ file နှစ် file ရဲ့ အချက်အလက်တွေပါ။ code ကို တစ်ချက် ဖြန်ကြည့်လိုက်ပါမယ်။ အဲဒီမှာ ပထမရေးလိုက်တဲ့ တစ်ကြောင်းက input type file ဆိုတဲ့ Element ကို myFile ဆိုတဲ့ variable ထဲကို assign လုပ်လိုက်တာပါ။ ပြီးတော့ အောက်မှာရေးထားတဲ့ တစ်ကြောင်းကတော့ အဲဒီ Element ထဲမှာ ရွေးထားတဲ့ file တွေကို ပြန်ထုတ်ကြည့်တာပါ။ Debug လုပ်ကြည့်တဲ့သဘောပါပဲ။

ဒါဆို file တွေကို handle လုပ်လိုရန်ပြီပဲ။ အဲဒီ file တွေကိုသာ ajax အတွက် နည်းနည်းပြင်ဆင်ပေးလိုက်ရင် အားလုံးအဆင်ပြေပြီ။ ဒီတော့ ပြင်ဆင်လိုက်တာပေါ့များ။ ဒါမျိုးလေးရေးပြီးတော့။

```

<script>

var datas = new FormData();

$('#submit').bind('click', function(e){
  e.preventDefault();

  var count = myFile.files.length;

  for (var i = 0; i < count; i++) {
    var f = myFile.files[i];
    datas.append('files[]', f);
  }
});

</script>

```

ဒါဆို Code ထဲမှာ သာတွေရေးထားလဲဆိုတာ ကြည့်ရအောင်။ ပထမဆုံးအကြောင်းက FormData object တစ်ခု တည်ဆောက်လိုက်တာပါ။ တစ်ခု ရှိတာက FormData ဆိုတဲ့ object က တရာ့။ Browser တွေက Support မလုပ်ပါဘူး။ ဥပမာ - IE8 နဲ့ သူရဲ့ရှေ့က IE version တွေပေါ့။ ဒါကို စစ်ထားချင်တယ်ဆိုရင်လည်း စစ်လို့ရပါတယ်။ if (window.FormData) ဆိုပြီး စစ်ထားလို့ရပါတယ်။

Multiple file upload ဆိုတော့ ရွေးထားတဲ့ file တွေက array တွေနဲ့လာပါလိမ့်မယ်။ အဲဒီ array ထဲက file တွေကို ajax request ထဲကနေ Server ကို ပို့နိုင်ဖို့ ပြင်ဆင်ရပါမယ်။ ဒါကြောင့် ရှုံးရှုံးရှင်းရှင်း for loop နဲ့ပတ်ပြီတော့ datas object ထဲကို ပေါင်းထည့်ပေးလိုက်ပါတယ်။ append ဆိုတဲ့ function ထဲက ပထား parameter ဖြစ်တဲ့ files[] ဆိုတဲ့ array ထဲကို ရွေးထားတဲ့ file တွေကို ပေါင်းထည့်ပေးလိုက်ပါတယ်။ php ဖက်ကနေ ပြန်ယူထားတာက \$_FILES ထဲက files ဆိုတဲ့ array ထဲက ယူထားတာမို့လိုပါ။ ဒါတွေ အကုန် ပြင်ဆင်ပြီးသွားပြီဆိုရင်တော့ ajax နဲ့ upload လုပ်ဖို့ အဆင်သင့် ဖြစ်သွားပါပြီ။

```
$.ajax({
    url: 'http://localhost/ajax-upload/upload.php',
    type: 'POST',
    data: datas,
    processData: false,
    contentType: false
});
```

ဒါ Code လေးကိုသာ for loop ခဲ့အောက်မှာ ထည့်ရေးပေးလိုက်ရင် ajax upload လုပ်ငန်းစဉ်ပြီးဆုံးသွားပြီလို ထင်ပါတယ်။ Upload တင်ကြည့်လိုက်ပါတယ်။ ဟာ.. ဝမ်းသာစရာပဲ။ File ရွေးပြီး upload button ကို နိုင်လိုက်တဲ့အခါမှာ upload.php ကို မသွားဘူး၊ reload မလုပ်ဘူး ဒါပေမယ့် file ဆိုတဲ့ directory ထဲကိုသွားကြည့်လိုက်တော့ တင်လိုက်တဲ့ file တွေရောက်နေပြီ။ ဒီလိုပဲဖြစ်ရမှာပေါ့။

jQuery ခဲ့ ajax ကို သုံးသွားပုံလေးကို ပြန်ကြည့်လိုက်ပါပြီး။ ပထားဆုံး option (url) က ကိုယ်ခဲ့၍ file တွေကို upload တင်မယ့် link ပါ။ type မှာတော့ POST method နဲ့ သွားမယ်လို ပြောလိုက်ပါတယ်။ data ထဲမှာတော့ ခုနက ပြင်ဆင်ခဲ့တဲ့ datas ဆိုတာကို ထည့်ပေးလိုက်ပါတယ်။ Default အနေနဲ့ဆိုရင် ပို့လိုက်တဲ့ data ကို queryString အဖြစ်ပြောင်းပြီး ပိုပါတယ်။ အဲဒီကို မပြောင်းချင်တဲ့အတွက် processData ကို false ပေးလိုက်ပါတယ်။ နောက်ထပ် တစ်ခုကဗျာည်း အဲလိုပါပဲ။ jQuery က default content type ကို application/x-www.form-urlencoded; charset=UTF-8 အဖြစ်ထားပါတယ်။ ဒါပေမယ့် အဲလိုမထားချင်ဘူး ကိုယ်ပို့လိုက်တဲ့ file နဲ့ဆိုင်တာ (multipart/form-data) ပဲ ထားချင်တာဆိုတော့ false ပေးလိုက်ပါတယ်။

ကဲ Ajax file upload ပြီးသွားပြီဆိုတော့ ဒီ tutorial လေးကို ကြည့်ပြီး ကိုယ်လိုအပ်တဲ့နေရာမှာ ဥာက်ရှိသလို အသုံးပြုနိုင်ပါသည်ပေါ့များ။ ဒီလောက်ဆိုရင် Php file upload with ajax ကို ကောင်းကောင်း ရေးတတ်သွားပြီဟုတ်။

■ PHP Exceptions

by K



ARE YOU AN EXCEPTION?

BECAUSE I CAN'T WAIT TO CATCH YOU.

 Image Credit - <http://tvtropes.org>

[about](#)

K

Khay | Khay Usaki | Anime and Game Enthusiast.

RebornCMS core module developer.

 khay.me

 [/khaylay](https://www.facebook.com/khaylay)

 [@khayusaki](https://twitter.com/khayusaki)

 [/khay](https://github.com/khay)

Errors ဆိတ် programming နဲ့ ပတ်သက်လာရင် developer တွေ အမြတ်များ တွေ့ကြံးနေရတဲ့ အရာတွေပဲ ဖြစ်ပါတယ်။ အဲဒီ Errors တွေကို အမျိုးအစား ခွဲခြား ကြည့်လိုက်မယ်ဆိုရင် စာလုံးပေါင်းမှားယွင်းနေတတ်တဲ့ syntactical error, run-time ဝါမှမဟုတ် logical error တစ်ခုရှုပါ။ Syntax error အနေနဲ့ developer အများစုံ အဖြစ်များတာကတော့ statement တွေအဆုံးမှာ semicolon တွေမပါတာတွေ အဖြစ်များတာပါပဲ။ Database ကိုချိတ်ဆက်တဲ့အခါမျိုးမှာ server down နေတာမျိုးကိုတော့ run-time error လို့ခေါ်ပါတယ်။ Logic error ကတော့ variable နဲ့ မသက်ဆိုင်တဲ့ data တွေ အသုံးပြုတာမျိုးကို ဆိုလိုပါတယ်။ ကျွန်တော်တို့ ရေးနေတဲ့ code တွေမှာ error တွေ နည်းနိုင်သမျှ နည်းပါးနိုင်အောင် ကျွန်တော်တို့ရဲ့ web application တွေမှာ သင့်တော်တဲ့ error handling တွေ လိုအပ်ပါတယ်။

Exceptions တွေကတော့ program ကနေပြီးတော့ errors တွေရှိလာတဲ့အခါ ပြောပြတဲ့ special condition တွေပဲ ဖြစ်ပါတယ်။ Exception တွေက objects တွေ ဖြစ်သလို PHP ရဲ့ built-in class တစ်ခုလည်း ဖြစ်ပါတယ်။ Exception object တစ်ခုမှာဆိုရင် ဘယ်နေရာမှာ error ရှိနေသလဲဆိုတာကို filename နဲ့ line number နဲ့ ဖော်ပြပေးသလို၊ error message အပြင် error code နဲ့ ပတ်သက်တဲ့ အချက်လက်တွေ ပါဝင်ပါတယ်။

Handling Exceptions

အခု exceptions throw ဖြစ်နိုင်တဲ့ function တွေကို ဘယ်လို ထိန်းချုပ်ရမလဲ။ PDO (PHP Data Objects) extensions တွေကလည်း exceptions throw ပြုလုပ်တဲ့အတွက်ကြောင့် ဥပမာအနေနဲ့ PDO ကို အသုံးပြုလိုက်ပါမယ်။ အောက်က code ကို ကြည့်လိုက်မယ်ဆိုရင် database ကို ချိတ်ထားတယ်၊ ဒါပေမယ့် ကျွန်တော်တို့မှာ 'mydb' ဆိုတဲ့ host ရှိမနေပါဘူး။

```
$database = new PDO ('mysql:host=mydb');
```

အဲဒီ code ကိုသာ run လိုက်မယ်ဆိုရင် fatal error ကိုတွေ့ရပါလိမ့်မယ်။ Connection ချိတ်မရတဲ့အတွက်ကြောင့် PDO class ကနေပြီးတော့ exception တစ်ခုကို throw လုပ်လိုက်ပါလိမ့်မယ်။ အဲဒီ error ကို မမြင်ချင်တဲ့အတွက်ကြောင့် ကျွန်တော်တို့တွေ try/catch ကို အသုံးပြုလိုက်ပါမယ်။

```
try {
    $database = new PDO('mysql:host=mydb');
    echo "Connected to database";
} catch (Exception $e) {
    echo "Huh? " . $e->getMessage();
}
```

ဒါက try/catch ပုံစံကို ဖော်ပြထားတဲ့ နမူနာ code ပုံစံပါ။ try block ထဲမှာဆိုရင် ကျွန်တော်တိ။ application မှာ run နိုင်းမယ့် code တွေကို ရေးထားမယ်၊ ဒါပေမယ့်လည်း exception throw ဖြစ်ပြီး error တက်လာနိုင်တာကို သိထားရမယ်။ အဲဒီအတွက် catch block ထဲမှာ ဖြစ်လာမယ့် error တွေနဲ့ ပတ်သက်ပြီး ထိန်းချုပ်ဖော်ပြပေးမယ့် code တွေကို ရေးရပါမယ်။

သိထားရမှာ try/catch နဲ့ ရေးပြီဆိုတာနဲ့ အကယ်၍ ကျွန်တော်တိ။ code မှာ database ကို ချိတ်တဲ့အခါ exception ဖြစ်ခဲ့မယ်ရင် PHP အနေနဲ့ try ထဲမှာရှိတဲ့ ကျွန်တဲ့ code တွေကို အလုပ်လုပ်မှုမဟုတ်တော့ပဲ catch ကို တန်းရောက်သွားမှာပဲ ဖြစ်ပါတယ်။ ဒီညာမှာဆိုရင် exception ရှိနေတဲ့အတွက်ကြောင့် ကျွန်တော်တိတွေအနေနဲ့ "Connected to database" ဆိုတဲ့ message ကိုတွေ့ရမှာမဟုတ်ပါဘူး။

တွေား programming language တွေမှာဆိုရင် ကျွန်တော်တိတွေ အနေနဲ့ try/catch/finally ဆိုပြီးအသုံးပြုရပေမယ့် PHP မှာတော့ finally ဆိုတာ မလိုအပ်ပါဘူး။

Why Exceptions?

အရင်ကျွန်တော်တိ။ အသုံးပြုခဲ့တဲ့ Error ပြဿနာ ပုံစံတွေနဲ့ နှင့်ယူဉ်လည်းမယ်ဆိုရင် Exceptions တွေက သပ်ရပ်လုပ်လွှန်းပါတယ်။ ကျွန်တော်တိတွေအနေနဲ့ ဘယ်လို ပြဿနာမျိုးလဲဆိုတဲ့ အပေါ်မှုတည်ပြီး exceptions တွေကနေ application ကို ဆက်လက် လုပ်ဆောင်သင့်လား? ဒါမှမဟုတ် လုလေပေနဲ့ ရင်တန်းလိုက်မလား ဆိုတာကို သတ်မှတ် ပြုလုပ်ပေးနိုင်ပါတယ်။

အစိုင်းမှာတုန်းက exceptions တွေက objects တွေ ဖြစ်တယ်လို့ ပြောခဲ့ပါတယ်။ အဲဒီလို objects တွေ ဖြစ်တဲ့အတွက်ကြောင့် ကျွန်တော်တိတွေအနေနဲ့ exceptions အသစ်တွေကို ထပ်မံထည့်သွင်းနိုင်သလို data နဲ့ behavior တွေကိုလည်းပြုပြန်နိုင်ပါတယ်။ အကယ်၍ objects တွေနဲ့ ပတ်သက်ပြီး ဘယ်လိုလုပ်ဆောင်နိုင်သလဲဆိုတာကို နားလည်ပြီးသားဆိုရင်တော့ exception တွေနဲ့ ပတ်သက်ပြီးတော့ အခက်အခဲရှိမှာ မဟုတ်သလို ကျွန်တော်တိတွဲ လိုအပ်တဲ့အတိုင်း error handling တွေကို လွှာယ်လွယ်ကူကူနဲ့ လုပ်ဆောင်နိုင်ပါလိမ့်မယ်။

Throwing Exceptions

အခုဆိုရင် PHP မှာ built-in ပါဝင်တဲ့ exceptions throw နဲ့ပတ်သက်ပြီး ဘယ်လို ပြုလုပ်ရမလဲဆိုတာကို အပေါ်က ဥပမာမှာ တွေ့ပြီးသွားပါပြီ။ ဆက်ပြီးတော့ exceptions throw တွေကို ကျွန်တော်တိတွဲအနေနဲ့ ဘယ်လိုမျိုး ကိုယ်တိုင် လုပ်ဆောင်နိုင်မလဲ?

```
//something is wrong  
throw new Exception ('Error is occurred, something is wrong');
```

Exception throw ပြုလုပ်ဖို့အတွက် ကျွန်တော်တို့, throw ဆိုတဲ့ keyword ကို အသုံးပြုလိုက်ပါတယ်။ ပြီးတော့ thrown ပြုလုပ်ဖို့အတွက် Exception object တစ်ခုကို သတ်မှတ်ပေးတယ်။ Exception သတ်မှတ်ပြီးပြီ ဆိုတာဘူး။ constructor ကို parameter အနေနဲ့ error message ပြန်ပေးလိုက်တယ်။ အဲဒီ constructor မှာပဲ error code ကိုလည်း ပြန်ပေးလိုတယ်ဆိုရင် optional အနေနဲ့ second parameter ကို အသုံးပြုနိုင်ပါတယ်။

Extending Exceptions

ကျွန်တော်တို့အနေနဲ့ ကိုယ်ပိုင် classes တွေနဲ့ သက်ဆိုင်ရင် exception ပုံစံတွေကို ပြုလုပ်ဖို့အတွက် Exception object ကို extend ပြုလုပ်နိုင်ပါတယ်။ PDO extension အနေနဲ့ exception thrown လုပ်တော့မယ်ဆိုရင် PDOException ပုံစံနဲ့ လာတဲ့အတွက်ကြောင့် ကျွန်တော်တို့အနေနဲ့ database error လား၊ တွော်း ဘယ်လို exception မျိုးလဲဆိုတာကို အလွယ်တကူ ခွဲခြား သိနိုင်ပါတယ်။ Exception ကို extend လုပ်မယ်ဆိုရင် object inheritance လုပ်ပြီး အလွယ်တကူနဲ့ အသုံးပြုနိုင်ပါတယ်။

```
class CustomException extends Exception {}
```

ဘယ်လို properties မျိုးမဆို methods မျိုးမဆို ကျွန်တော်တို့ ထည့်သွင်းလိုသမျှကို ဒီ Exception class ထဲကို ထည့်သွင်းနိုင်ပါတယ်။ တစ်ခုသတိထားရမှာက အကောင်းရှိ base ဖြစ်တဲ့ Exception class ကို extend လုပ်ထားရပါမယ် မဟုတ်ရင် PHP အနေနဲ့ အဲဒီကို thrown ပြုလုပ်ခွင့် ပြုမှာ မဟုတ်ပါဘူး။ အဲဒီလိမ့်မျိုး အသုံးပြုခြင်းကနေ ကျွန်တော်တို့အနေနဲ့ application ရဲ့ ဘယ်နားမှာ ဘယ်လို ဖြစ်နေတယ်ဆိုတာမျိုးကို program ကနေ ပြန်လာပေးတဲ့ error messages တွေကို ဖတ်နေစရာ မလိုတော့ပဲ exception တွေကနေ တစ်ဆင့် သိနိုင်မှာပဲ ဖြစ်ပါတယ်။ မတူညီတဲ့ exception classes တွေကို အသုံးပြုခြင်းကနေ မတူညီတဲ့ exception type တွေကို catch ပြုလုပ်နိုင်မှာပဲ ဖြစ်ပါတယ်။

Catching Specific Types of Exception

အောက်က Code ကို ကြည့်လိုက်ပါ။ Exception thrown တွေအများကြေား ထည့်သွင်းထားတာကို တွေ့ရပါမယ်။

```

class StudentException extends Exception {}

class School {
    public function classroom(Student $student) {
        //check student has a name
        if(empty($student->name)) {
            throw new Exception('Student name is not specified.');
        }

        // check age
        if($student->age < 5) {
            throw new StudentException('A student must be at least 5 years old.');
        }

        //working okay
        return true;
    }
}

```

ဥပမာကို ကြည့်လိုက်မယ်ဆိုရင် StudentException ဆိုတဲ့ exception ကို empty အနေနဲ့ထွေးရပါမယ်။ School ဆိုတဲ့ class မှာတော့ classroom() ဆိုပြီး method ပါဝင်ပြီး Exception ကော့ StudentException နှစ်ခုလုံးကို throw လုပ်ထားတာ ထွေးရပါတယ်။

အောက်က code မှာတော့ catch block နှစ်ခုသုံးထားပါတယ်။

```

$iSchool = new School();
$student = new Student();

//add students randomly
$student->age = rand(3,9);

try {
    $iSchool->classroom($student);
    echo "Student registered!";
} catch (StudentException $e) {
    echo "Student age error : " . $e->getMessage();
    // redirect to register new student
} catch (Exception $e) {
    echo "Have error here : " . $e->getMessage();
    // error found, stop every process
    exit;
}

```

ဥပမာထဲမှာ ကျွန်တော်တို့ အဖော်၏ School ကော Student objects နှစ်ခုလုံးကို သတ်မှတ်ကြောထားပါတယ်။ Student object မှာဆိုရင် name နဲ့ age နှစ်ခုလုံး ပါဝင်ရပါမယ်။ အဲဒီတော့ ကျွန်တော်တို့ Student တွေကို သတ်မှတ်ပါမယ်။ အခုတော့ PHP ရဲ့ rand() ဆိုတဲ့ function ကို အသုံးပြုပြီး ကျောင်းသားတွေရဲ့ age ကို random ပြုလုပ်လိုက်ပါတယ်။

Try block ထဲမှာဆိုရင် ကျွန်တော်တို့တွေက ကျောင်းသားကို classroom (ကျောင်းခန်း) သတ်မှတ်ပေးဖို့ ပြုလုပ်ထားပါတယ်။ အကယ်၍ အားလုံးသာ အဆင်ပြေချောမွေ့စွာနဲ့ ပြီးဆုံးသွားမယ်ဆိုရင်တော့ "Student registered!" ဆိုတဲ့ message ကိုမြင်တွေ့ရမှာ ဖြစ်ပါတယ်။ အဲဒီနောက်မှာတော့ ပြဿနာတွေ့ရင် သပ်သပ်ရပ်ရပ်ဖြေရှင်းနိုင်ဖို့ အတွက် catch block နှစ်ခုကို တွေ့ရပါလိမ့်မယ်။ ပထမ catch block ကတော့ StudentException နဲ့ ပတ်သက်သမျှကို ဖမ်းပေးထားပါတယ်။ အကယ်၍ ပထမ catch နဲ့ ကိုက်ညီတဲ့ ပြဿနာမဟုတ်ဘူးဆိုရင် နောက်ဆုံး ကျွန်ရှိနေတဲ့ ဘယ်လို exception မျိုးကိုမဆို catch လုပ်ပေးထားမယ့် Exception ကနေပြီးတော့ ဖမ်းထားပေးမှာပဲ ဖြစ်ပါတယ်။

Program ကို run ကြည့်မယ်ဆိုရင် အရင်ဆုံး Student ရဲ့ age က 3 နဲ့ စလာမှာ ဖြစ်ပါတယ်။ try လုပ်တဲ့အခါမှာတော့ classroom ဆိုတဲ့ function ထဲကို ဝင်သွားပါမယ်။ အဲဒီထဲမှာ အနည်းဆုံး age က 5 ထက်ကြီးမှ လက်ခံမှာ ဖြစ်တဲ့အတွက်ကြောင့် StudentException နဲ့ catch ဖြစ်သွားတဲ့အတွက်ကြောင့် အောက်ကအတိုင်း Message ကိုတွေ့ရပါလိမ့်မယ်။

Student age error : Student name is not specified.

နောက်ထပ်တစ်ခေါက် age ကို 9 နဲ့ လာတဲ့အခါမှာတော့ သတ်မှတ်ထားတဲ့ age limit မှာ ရှိတဲ့အတွက်ကြောင့် Student-Exception catch မမိပေမယ့် student မှာ name မပါလာတဲ့အတွက်ကြောင့် (empty ဖြစ်နေတဲ့အတွက်ကြောင့်) ကျွန်တဲ့ Exception က catch ဖြစ်ပြီး အောက်က error message ကို တွေ့ရပါလိမ့်မယ်။

Have error here : Student name is not specified.

ဥပမာထဲမှာတော့ exceptions တွေကို class အထဲမှာ thrown လုပ်ထားပေမယ့် နောက်ပိုင်းမှာ objects ရဲ့ method အဖော်၏ catch ပြန်လုပ်ထားပါတယ်။ အကယ်၍ Exceptions တွေအဖော်၏ catch မဖြစ်သွားဘူးဆိုရင် Program run တဲ့အခါမှာ Un-caught Exception ဆိုပြီး fatal error တွေကိုတွေ့ရတတ်ပါတယ်။

Setting a Global Exception Handler

Exceptions တွေကို throw လုပ်တဲ့အခါ ကျွန်တော်တို့ရဲ့ code တွေက catch မလုပ်နိုင်တာတွေကို မဖြစ်စေဖို့အတွက် ကျွန်တော်တို့ရဲ့ application မှာ default behavior တွေ သတ်မှတ်နိုင်ပါတယ်။ အဲဒီအတွက် set_exception_handler() ဆိုတဲ့ function ကို အသုံးပြုပါမယ်။ အဲဒီ function ကို အသုံးပြုပြီးတော့ အကယ်၍ exception က try/catch block မှာ catch မဖြစ်တဲ့အခါမျိုးအတွက်

default exception handler သတ်မှတ်ပေးတာပဲ ဖြစ်ပါတယ်။ Exception handler ကလည်း user ကို error ပြတာပဲ ဆိုပေမယ့် fatal error message ထက်စာရင်တော့ ကြည့်လို့ ကောင်းပါတယ်။ exception_handler ခေါ်လိုက်ပြီ ဆိုတာနဲ့ လုပ်ဆောင်ချက်တွေ အားလုံး ရပ်တန်းသွားမှာပဲ ဖြစ်ပါတယ်။

သာမန် exception hander တစ်ခုအနေနဲ့ အောက်က code တွေနဲ့ ဆင်တူပါတယ်။

```
function exceptionHandler($e) {  
    echo "Uncaught exception : ", $e->getMessage();  
}  
  
set_exception_handler('exceptionHandler');  
throw new Exception('Uncaught Exception!');
```

Exception handler အနေနဲ့ function တစ်ခုကို သတ်မှတ်ပြီး set_exception_handler နဲ့ uncaught exceptions တွေအတွက် အဲဒီ function ကို သတ်မှတ်ပေးလိုက်တာပဲ ဖြစ်ပါတယ်။ Program တစ်ခုမှာ အဲဒီလို function မျိုး ပါဝင်ခဲ့မယ်ဆိုရင် အမြဲလိုလိုကတော့ bootstrap ဖိုင်တွေ ဒါမှုမဟုတ် script တွေရဲ့ အစိုင်းလောက်မှာ သတ်မှတ်ကြောပေးထားတာကို တွေ့ရတတ်ပါတယ်။

Exception တွေကို ထိန်းချုပ်ဖို့ အတွက် set_exception_handler() ကို အသုံးပြုသလို PHP မှာပဲ ထပ်ပြီးတော့ errors တွေအတွက် set_error_handler ဆိုတာလည်း ရှိပါသေးတယ်။

အခုလောက်ဆိုရင်တော့ PHP Exception အပိုင်းနဲ့ ပတ်သက်ပြီး try/catch နဲ့ exception တွေကို အကြမ်းရှင်းလောက်ပဲ ဖြစ်ဖစ် နားလည်သဘောပေါက်ပြီလို့ ထင်ပါတယ်။ Exception တွေနဲ့ ပတ်သက်ပြီး သိချင်သေးတယ်ဆိုရင်တော့ Brandon Savage ရေးထားတဲ့ [Exceptional PHP : Introduction to Exceptions](#) ဆိုတဲ့ ဆောင်းပါးအပြင် Nettuts ရဲ့ tutorial တစ်ခုဖြစ်တဲ့ [The Ins and Outs of PHP Exceptions](#) တွေကို ဖတ်ကြည့်သင့်ပါတယ်။

■ New Features from PHP 5.4

by Nyan Lynn Htut

php 5.4

about

Nyan Lynn Htut

Myanmar Links Professional Web Development Team
မှာ Web Developer တစ်ယောက် အဖော်၏ လုပ်ကိုင်နေသူ
တစ်ယောက်၊ PHP Open Source CMS တစ်ခြုံဖြစ်တဲ့ Re-
born CMS ရဲ့ Core Developer တစ်ယောက်၊ REST API
နဲ့ Frontend Javascript Application ထွေအကြောင်းကို
စိတ်ဝင်စားသူ။



www.reborncms.com



[/nyanlynnhtut](#)



[@nyanlynnhtut](#)



[/nyanlynnhtut](#)

Traits

PHP မှာ interface ကိုတော့ multiple implement ရပေမယ့် class တစ်ခုကို inheritance (extend) လုပ်တဲ့အခါမှာ multiple မရပါဘူး။ ဒီအတွက် code တွေကို reuse လုပ်တဲ့နေရာမှာ အခက်အခဲတစ်ချို့ရှုပါတယ်။ ဒီအတွက် အဆင်ပြေအောင် 5.4 မှာ traits ဆိုပြီး PHP က feature တစ်ခုကို မိတ်ဆက်ပေးခဲ့ပါတယ်။

```
<?php
class Base {
    public function sayHello() {
        echo 'Hello ';
    }
}

trait SayWorld {
    public function sayHello() {
        parent::sayHello();
        echo 'World!';
    }
}

class MyHelloWorld extends Base {
    use SayWorld;
}

$o = new MyHelloWorld();
$o->sayHello();
```

ဒီအတွက် ကျွန်ုတ်တို့ အသုံးပြုတဲ့ code တွေက ပိုပြီးတော့ reuse ဖြစ်သွားပါတယ်။ အရင်က single inheritance ပဲရတဲ့အတွက် multiple လိုချင်ရင် ကြံးရတဲ့ အခက်အခဲတွေကို ပြောလည်သွားစေပါတယ်။

Short Array Syntax

PHP မှာ 5.4 ကနေစပြီးတော့ array ကို short syntax ကို သုံးပြီးရေးလို့ရပါတယ်။ အရင်က PHP မှာ array ကို ရေးတော့မယ်ဆိုရင် array(...) ဆိုပြီးရေးမှုရပေမယ့် အခါ5.4 ကနေစပြီးတော့ ဒီလိုရေးလို့ရလာပါတယ်။

```
$array = [1, 2, 3, 4];
$free = [];
```

Function array dereferencing

5.4 မဝိုင်ခင် PHP version တွေမှာ array() ပြန်လာတဲ့ function က value ကို တန်းပြီး ၏လိုမပါဘူး။ အဲဒီအတွက် variable တစ်ခုထဲကို assign လုပ်ပြီးမှသာ ၏လိုမပါဘူး။ ဥပမာ ဒီလို function မျိုးဆိုပါတယ့်

```
function test()
{
    return array('hello', 'world', 'myanmar');
}
```

ကျွန်တော်တို့အနေနဲ့ ဒီ function က result ရဲ့ array ထဲက ရှေ့ဆုံးတစ်ခုကိုပဲ လိုချင်တယ်ဆိုရင် ဒီလိုမျိုးသုံးပေးပါတယ်။

```
$result = test();
echo $result[0];
```

5.4 မှာတော့ ကျွန်တော်တို့၊ တွေ တိုက်ရှိက်သုံးလို့ရလာပါပြီ။

```
echo test()[0];
```

Closures now support \$this

အရင်က PHP ရဲ့ Closure မှာ \$this ကို သုံးလို့မပါဘူး။ အဲဒီအတွက် Object ထဲမှာ Closure ကို သုံးလို့ Object ထဲက variable တစ်ခုရှုကို လိုချင်ရင် use ဆိုတဲ့ keyword နဲ့ တွေပြီးသုံးပေးပါတယ်။ အခုတော့ ဒီလိုမျိုး လုပ်စရာမလိုတော့ပဲ \$this ဆိုတဲ့ keyword ကို Closure ထဲမှာ အသုံးချလို့ရပါတယ်။

ဒါကတော့ 5.3 မှာရေးတဲ့ ပုံစံပါ။

```
class Support
{
    protected $name = 'Foo';

    public function __construct()
    {
        $ins = $this;
        $func = function() use ($ins) {
            return $ins->name;
        };
    }
}
```

```

        echo $func();
    }
}

$s = new Support();

```

အခ 5.4 က စပြီးတော့ အခလို သုံးလို ရပါတယ်။

```

class Support
{
    protected $name = 'Foo';

    public function __construct()
    {
        $func = function() {
            return $this->name;
        };

        echo $func();
    }
}

$s = new Support();

```

ဒီတစ်ချက်ကတော်တော်လေးကို အဆင်ပြေစေပါတယ်။ အရင်ကဆို use နဲ့ ပြန်ညွှန်းရတာနဲ့ public variable မဟုတ်တော့ အပြင်ကနေ ခေါ်လိုမရတာနဲ့ (Closure ကို Object ရဲ့ scope ပြင်ပလို့ သတ်မှတ်ထားတဲ့အတွက်) အစရိတ္တဲ့ ပြဿနာတွေ ရှိပါတယ်။ အခတော့ အဲဒီလို ပြဿနာတွေ ကင်းဝေးသွားပါပြီ။

Class member access on instantiation

ဒီ feature ကလဲတော်တော်လေးကို ကောင်းမွန်တယ်လို့ ပြောလို့ရပါတယ်။ ဘာကြောင့်လဲဆိုတော့ ကျွန်တော်တို့မှာ ဒီလိုမျိုး get() ဆိုရင် name ကို return ပြန်ပေးမယ့် method တစ်ခုရှိတယ်ဆိုပါတော့။ လိုချင်တာကလဲ ဒီ get() တစ်ခုတည်းပေါ့။

```

class Hello
{
    protected $name = 'Hello World';

    public function get()
    {
        return $this->name;
    }
}

```

Object ကို instantiate လုပ်ပြီး get() ကို ၏၏ဖို့ရာ အရင် version တွေမှာဆို တစ်ကြောင်းထဲနဲ့ မပြီးပါဘူး။ ပထမဆုံး instantiate လုပ်ပြီး variable ထဲကို assign အရင်လုပ်ရပါတယ်။ ပြီးမှ get method ကို ၏၏လို့ရပါတယ်။

```
$obj = new Hello();
echo $obj->get();
```

5.4 ကနေ စီးပွားရေးလုပ်လုပ်မယ်။

```
echo (new Hello())->get();
```

တစ်ကြောင်းထဲနဲ့တင် ကျွန်တော်တို့သုံးလို့ရသွားပါတယ်။ variable ကို assign လုပ်စရာမလိုလို။ performance ပိုင်းမှာလဲ ပိုပြီးကောင်းမွန်သွားပါတယ်။

Web server in CLI mode

Web အတွက်ရေးလို့ရတဲ့ Language တော်တော်များများက Web Server အတွက် ကိုယ်တိုင် လက်ရှိရေးနေတဲ့ folder ထဲမှာတင် run လို့ရနေတဲ့အခိုန်မှာ PHP အတွက်ကတော့ Apache လိုမျိုး service တစ်ခုကို အားကိုးနောပါတယ်။ Project Folder ထဲမှာ run လို့ရတဲ့အတွက် development stage မှာအများကြီး အသုံးဝင်ပါတယ်။ Server area ထဲကို file တွေရွေ့နေစရာမလိုပဲ ကြိုက်တဲ့ နေရာမှာ run လို့ရတဲ့အတွက် တော်တော်လေးအဆင်ပြေပါတယ်။ ဒါအတွက် PHP သမားတွေလဲ 5.4 ကနေစလို့ Development အတွက် သူများတွေလို ကိုယ်ပိုင် Server နဲ့ run လို့ရပါပြီ။ ဒါ feature ကတော့ PHP ကို တစ်ခြား language တွေနဲ့ယဉ်ရင် နောက်ကျမကျန်နေစေတဲ့ feature တစ်ခုပါ။ Development အတွက် တော်တော်ကို အဆင်ပြေသွားစေပါတယ်။

Callable Typehint

Typehinting အတွက် callable ဆိုတာကိုပါ PHP 5.4 မှာ သုံးလို့ရလာပါတယ်။ အရင်က Object ကွဲ Array တွေပဲ သုံးလို့ရတာကနေ callable ဖြစ်တာကိုပဲ passing ပေးရမယ်ဆိုပြီ။ သတ်မှတ်ပေးလို့ ရလာတာဖြစ်ပါတယ်။

■jQuery User Interface

by Thet Paing



Image Credit - <https://www.crossdistinction.com>

about
Thet Paing

 /mgthetpaing

 @mgthetpaing

 /ThetPaing

ယခု တစ်လ Echo အတွက် ရေးမယ့် အကြောင်း အရာကတော့ jQuery JavaScript Library တစ်ခု အကြောင်းပါ။ အဲဒါကတော့ jQuery UI လိုချော့တဲ့ jQuery User Interface ပါ။ jQuery UI သည် user interface အတွက်လိုအပ်တဲ့ interactions တွေ Effects တွေ Widget တွေနဲ့ Themes အတွက် တည်ဆောက်ထားတဲ့ jQuery JavaScript Library တစ်ခုပါ။ သူ့ကို Website တွေ Web application တွေတည်ဆောက်တဲ့ နေရာမှာ အသုံးပြန်ပါတယ်။ jQuery UI ကို Download လုပ်ဖို့ အတွက်ကတော့ သူရဲ့ Website ဖော်တဲ့ <http://jqueryui.com> မှာ ရှိပါတယ်။ အခုခေါ်နေတဲ့ အချင့်မှာတော့ jQuery UI သည် Stable version တစ်ခုဖြစ်တဲ့ (v1.10.3 jQuery 1.6+) ကိုရောက်နေပါပြီ။

The screenshot shows the official [jQuery UI](http://jqueryui.com) website. At the top, there's a dark header bar with social media links (Facebook, Twitter, LinkedIn, YouTube, GitHub) and a navigation menu with links to Plugins, Contribute, Events, Support, and the jQuery Foundation. Below the header is the [jQuery UI](http://jqueryui.com) logo.

The main content area has a dark background with light-colored boxes for different sections:

- Interactions:** A list of interaction types: Draggable, Droppable, Resizable, Selectable, Sortable.
- Widgets:** A list of widget types: Accordion, Autocomplete, Button, Datepicker, Dialog, Menu, Progressbar, Slider, Spinner, Tabs, Tooltip.
- Effects:** A list of effect types: Add Class, Color Animation, Effect, Hide, Remove Class, Show, Switch Class, Toggle, Toggle Class.
- Utilities:** A list of utility types: Position, Widget Factory.
- What's New in jQuery UI 1.10?** A section describing the changes in version 1.10, mentioning bug fixes and API redesigns. It includes links to upgrade guides and changelogs for 1.10.0, 1.10.1, 1.10.2, and 1.10.3.
- Dive In!** A section encouraging users to explore the library, with a note about staying informed through the blog and Twitter.
- Download jQuery UI 1.10.3** (highlighted in blue): A section for downloading the library, featuring "Custom Download" and "Quick Downloads" for "Stable" (v1.10.3, jQuery 1.6+) and "Legacy" (v1.9.2, jQuery 1.6+).
- Developer Links:** A list of developer resources: Source Code (GitHub), jQuery UI Git (WIP Build), Theme (WIP Build), Bug Tracker, Submit a New Bug Report, Discussion Forum, Using jQuery UI, Developing jQuery UI, Development Planning Wiki, Roadmap, Previous Releases, Changelogs, Upgrade Guides.

jQuery UI မှာပါတဲ့ Feature တွေက Interactions, Widgets, Effects နဲ့ Utilities တို့ဖြစ်ပါတယ်။ Interactions ဆိတဲ့ အတိုင်းသူ ထဲမှာ လျှပ်ရှားနေတာ တွေပါပါတယ်။ Widgets တွေကတော့ Website ထဲမှာ သို့မဟုတ် Web application တွေထဲမှာ အသုံးပြုတဲ့ button တွေ menu တွေစတာတွေပါဝင်ပါတယ်။ အဲဒီထဲမှာပါတဲ့ Datepicker ကတော်တော်လေး အသုံးဝင်ပြီး ကောင်းလည်းကောင်းပါတယ်။ Effects တွေနဲ့ Utilities တွေကလည်း အသုံးဝင်ပါတယ်။ jQuery UI မှာပါတဲ့ Function တွေထဲက ကိုယ့်နဲ့ အသုံးတည့်မယ့် Function တွေကို ယူသုံးလိုက်ရှုပါပဲ။ ဒါဆိုရင်တော့ ကိုယ့်ရဲ့ Development time ကို လျော့ကျ စေမှာပါ။ Website တွေမှာ အတော်တော်လေး အသုံးများနေတဲ့ Draggable တို့၊ Sortable တို့ကို အစမ်းအနေနဲ့ သုံးပြုပါမယ်။ အရင်ဆုံးတော့ စမ်းစိုးအတွက် Folder တွေ File တွေ အရင်ဆောက်ပါမယ်။



ပုံထဲ ကအတိုင်း ဆောက်လိုက်ပါတယ်။ ပြီးရင် အရင်ပြီးဆုံး Draggable ကိုစမ်းပါမယ်။ ဒါကြောင့် index.html ထဲမှာ

```

<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <title>Echo jQuery UI Draggable + Sortable</title>
    <script src="js/jquery-1.9.1.js"></script>
    <script src="js/jquery-ui-1.10.3.custom.js"></script>
    <link rel="stylesheet" href="css/style.css">
    <script>
        $(function() {
            $("#echo-draggable").draggable({
            });
        });
    </script>
</head>
<body>

<ul>
    <li id="echo-draggable" class="test-echo-drag">Drag me</li>
</ul>
</body>
</html>

```

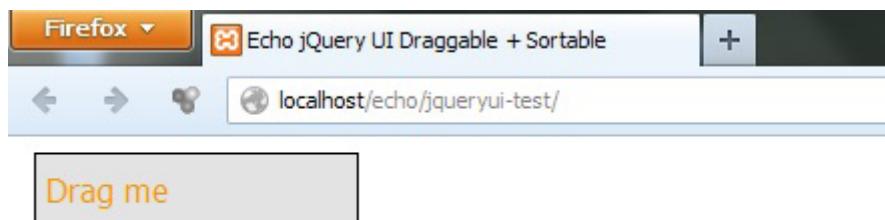
ရေးထားတဲ့ ပုံစံကတော့ echo-draggable ကို drag လုပ်လို့ရအောင်လုပ်မှာ ဖြစ်ပါတယ်။ ပြီးရင် အမြင်လေးလှသွားအောင် style.css ထဲမှာ

```

ul {
    list-style-type: none;
    margin: 0;
    padding: 0;
    margin-bottom: 10px;
}
li {
    margin: 5px;
    padding: 5px;
    width: 150px;
    background: #e3e3e3;
    border: 1px solid #000000;
}
.test-echo-drag {
    color: #FF9A07;
}

```

လိုပေးလိုက်ပြီး run ကြည့်မယ်ဆိုရင်



Draggable လုပ်လို ရပါပြီ။ နောက်တစ်ခု အနေနဲ့ Sortable ကိုပေးကြည့်ပါမယ်။ head ထဲက script ထဲမှာ

```

$( "#echo-sortable" ).sortable({
    revert: true
});

```

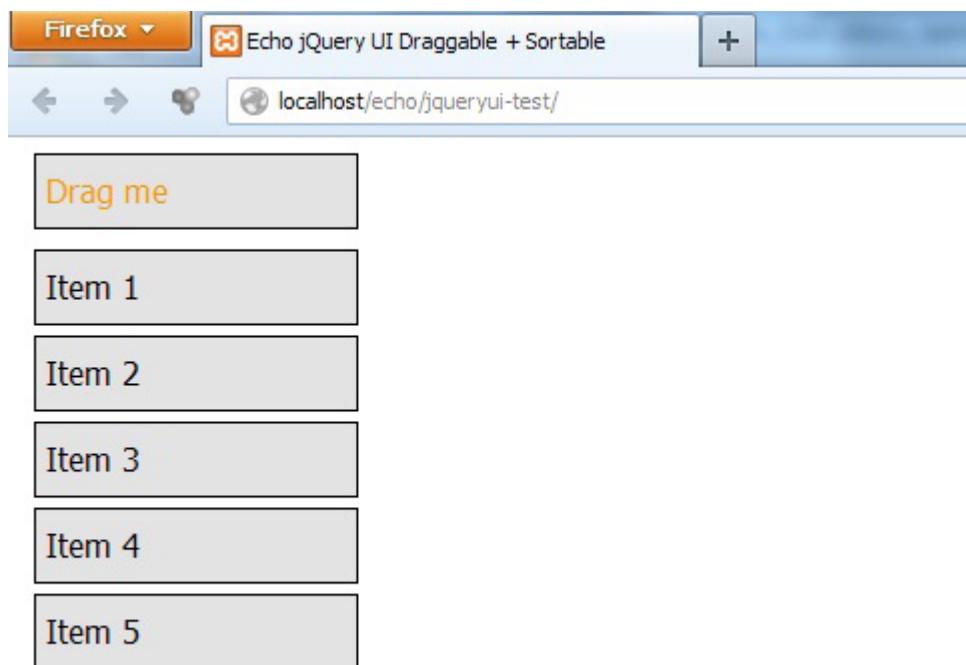
ပြီးတော့ body ထဲမှာ

```

<ul id="echo-sortable">
    <li class="test-echo">Item 1</li>
    <li class="test-echo">Item 2</li>
    <li class="test-echo">Item 3</li>
    <li class="test-echo">Item 4</li>
    <li class="test-echo">Item 5</li>
</ul>

```

ဒါလေးထည့်ရေးပြီး run ကြည့်မယ်ဆိုရင်



အပေါ်က ul > li က Drag လုပ်လို့ရပြီး အောက်က ul > li က Sort လုပ်လို့ရမှာပါ။ sortable ထဲမှာ ထည့်ရေးထားတဲ့ revert: true ကတော့ ကြည့်တဲ့အခါ အဆင်ရောအောင်လိုပါ။ နောက်ဆုံးအနေနဲ့ Drag နဲ့ Sort ကိုပေါင်းရေးကြည့်ပါမယ်။

```

$( "#echo-draggable" ).draggable({
    connectToSortable: "#echo-sortable",
    helper: "clone",
});

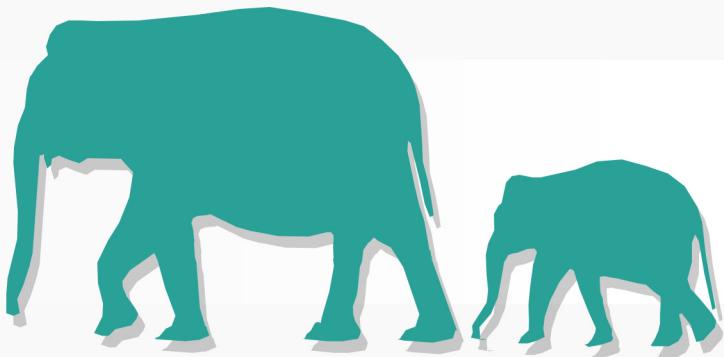
```

Script ထဲမှာရေးထားတဲ့ draggable ထဲမှာ code အနည်းငယ်ထည့်ရေး လိုက်ပါမယ်။ ဒါဆိုရင် အပေါ်က ul > li ကို အောက်က ul > li ထဲကိုဆွဲချုပြီး Sort လုပ်လို့ရပါပြီ။

jQuery UI က ရိုးရှင်ပြီး သုံးရလွယ်ကူပါ။ အထေက်ကရေးနဲ့ တာတွေကလည်း သူ Documentation ကိုကြည့်ပြီး ရေးထားထားပါ။ တစ်ခြား Function တွေက လည်း ထိုအတူပါပဲ။ အသုံးပြုရလွယ်ကူပါတယ်။ ဒါကြောင့် jQuery UI (jQuery User Interface) ကို Echo ပရီသတ် တွေနဲ့ မိတ်ဆက်ပေး ရွင်းဖြစ်ပါတယ်။ jQuery UI ကို အသုံးပြုပြီး Echo ပရီသတ်များ အကျိုးကော်မူး တစ်စုံတစ်ရာ ရရှိပါစေလို့ မျှော်လင့် ရင်း။

■ PHP Inheritance

by Thet Paing

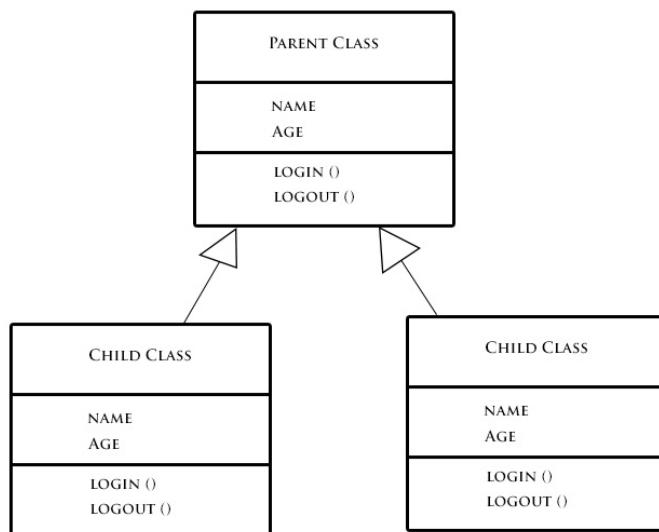


| PHP
Inheritance

about
Thet Paing

/mgthetpaing
 @mgthetpaing
 /ThetPaing

Inheritance ကို မြန်မာလိပ်စီရေမယ်ဆိုရင်တော့ အမွှေဆက်ခံတာပါ။ PHP မှာပါတဲ့ Object Inheritance ကလည်းထိုအတူပါပဲ။ Class တစ်ခုနဲ့ တစ်ခုကို Inheritance ကိုအသုံးပြုပြီးချိတ်ဆက်လိုက်တာပါ။ လာပြီးချိတ်ဆက်လိုက်တဲ့ Class က Child ဖြစ်ပြီး အချိတ်ခံလိုက် တဲ့ Class က Parent ဖြစ်ပါတယ်။ Parent နဲ့ Child က အပြင်လောကနဲ့ အတူတူပါပဲ။ မိဘ က သား၊ သမီးအများကို ရှိနိုင်ပေမဲ့ သား၊ သမီးမှာတော့ သူတို့ကို မွေးတဲ့ မိဘ က တစ်ယောက်တည်းဖြစ်မှာပါ။ ဆိုလိုတာက Parent Class ကတစ်ခုတည်းဖြစ်နိုင်ပေမယ့် Child Class ကတော့ အမြောက်အများဖြစ်နိုင်ပါတယ်။ Parent Class ကို base class ဒါမှမဟုတ် superclass လိုပေါ်တတ်ပြီး Child Class တွေကိုတော့ derived class ဆိုမဟုတ် subclass လိုပေါ်လေ့ရှိပါတယ်။ Inheritance ကိုအသုံးပြုလိုက်တဲ့ အတွက် Parent Class က properties တွေနဲ့ method တွေက Child Class မှာ အသုံးပြုလိုရနိုင်တာကြောင့် Development time ကိုသက်သာစေပါတယ်။ ပြီးတော့ Code တွေက ပြန်လည်အသုံးပြုလိုရတဲ့ အတွက် ညစ်ညမ်း Code (ရှုပ်ပြီး ထပ်ခါထပ်ခါရေးနေရသော Code) တွေကို လျော့ချုပြီးသားဖြစ်ပါတယ်။ သူရဲ့ ပုံစံကတော့



How to use Object Inheritance

Class တစ်ခုကို တစ်ခြား Class တစ်ခုက inherit လုပ်ချင်ရင်တော့ extends ဆိုပြီး အသုံးပြုရပါတယ်။ ဒါကြောင့် Code အနည်းငယ် ရေးပြီးရှင်းပါမယ်။ PHP Manual ထဲက Code ကိုပဲ အစ်းရေးပါမယ်။ ဒီနေရာမှာတစ်ခု အကြံပေးချင်တာက PHP ကိုလေ့လာမယ် ဆိုရင်တော့ PHP Manual ကို Offline Download လုပ်ထားသင့်ပါတယ်။ အဲဒီထဲမှာက ကိုယ်ရှာချင်တဲ့ PHP Function တွေ Code တွေ အသုံးအနှစ်းတွေကို ပြည့်ပြည့်စုစုပါတယ်။ Download လုပ်ရမယ့်နေရာကတော့ ဒီမှာ ပါ။

အခုအရင်ဆုံး PHP ကို run လိုရတဲ့ ကိုယ့်ရဲ့ local မှာ သွေးထားသော XAMPP ဒါမှမဟုတ် WAMP ဒါမှမဟုတ် အေားတစ်ခုခု ထဲမှာ Folder ဆောက်ထားရပါမယ်။ ကျွန်တော်ကတော့ XAMPP သုံးတဲ့ အတွက် XAMPP ထဲက htdocs ထဲမှာ Folder တစ်ခုဆောက်ပါတယ်။ အဲဒီ အထဲမှာ index.php file တစ်ခုဆောက်ပါတယ်။ ပြီးတော့ အဲဒီ File ထဲမှာ Class တစ်ခုရေးပါမယ်။

```

<?php

class Foo
{
    public function printItem($string)
    {
        echo 'Foo: ' . $string;
    }

    public function printPHP()
    {
        echo 'PHP is great.<br>';
    }
}

?>

```

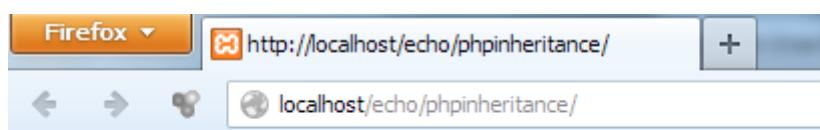
အဲဒီ အထဲမှာ printItem ဆိတ္တနဲ့ printPHP ဆိတ္တ Function နှစ်ခုရှိပါတယ်။ printItem ဆိတ္တဲ့ function ကဲ string တစ်ခုကို လက်ခံပြီး အဲဒီ string ကို echo ထုတ်ပေးပါတယ်။ အခု function ကိုပဲ အရင် run ကြည့်မယ်။ ဒါကြောင့် အဲဒီ Class ရဲ့ အပြင်ဖက်မှာ

```

$foo = new Foo();
$foo->printItem('baz.');
$foo->printPHP();

```

ဒါ Code တွေရေးပြီး run လိုက်မယ်ဆိုရင်



Foo: baz.PHP is great.

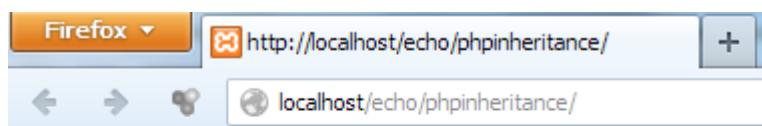
ဒီလိုပေါ်မှာပါ။ \$foo ထဲကို new ဆိုပြီး Foo() object တစ်ခုအောက်လိုက်ကာ Foo class ထဲက Function ကိုခေါ်သုံးလိုက်တာပါ။ နောက်တစ်ဆင့်အနေနဲ့ inheritance ကိုစပ်ပေးကြည့်ပါမယ်။ class Foo ရဲ့အောက်မှာ နောက်ထပ် class တစ်ခုထပ်အောက်ပါမယ်။

```
class Bar extends Foo
{
    public function printItem($string)
    {
        echo 'Bar: ' . $string;
    }
}
```

အဲဒီမှာ ရေးထားတဲ့ extends foo ဆိုတာက အပေါ်မှာရှိတဲ့ class Foo ကို inherit လုပ်လိုက်တာပါ။ ဒါဆိုရင်တော့ Foo မှာပါတဲ့ Function တွေကို Bar မှာ ပြန်သုံးလိုပါပြီ။ နောက်တစ်ခုရှိသေးတာ inheritance ဆိုတဲ့ သဘောက အမွှေရတာဖြစ်တဲ့ အတွက် inherit လုပ်တဲ့ class က သူ့ပြောင်းချင်တဲ့ ပုံစံ ပြန်ပြောင်းလိုပါတယ်။ ဆိုလိုတာက အဖော် သားကို ပိုက်ဆံအိတ် တစ်လုံးပေးလိုက်တယ်။ ပိုက်ဆံအိတ်ထဲမှာ ပိုက်ဆံတွေ ထည့်ပေးလိုက်တယ်။ ဒါကို သားက ပိုက်ဆံအိတ်ကို ပိုက်ဆံမထည့်ဘဲ စဉ်၍ထည့်ချင်လဲထည့်မယ်။ ဒါမှမဟုတ် ငှင့်ပိုက်ဆံအိတ်မှာ keychain လဲ ချိတ်ချင် ချိတ်ထားမှာပဲ။ အဲဒီ သဘောပါ။ ဒါကြောင့် အခုံ bar ဆိုတဲ့ class က foo ဆိုတဲ့ class မှာပါတဲ့ printItem ဆိုတဲ့ Function ကို အနည်းငယ် ပြောင်းလိုက်ပါတယ်။ echo 'Foo: ' ဆိုတာကို echo 'Bar: ' ဆိုပြီးပြောင်းလိုက်တာပါ။ ဒါတွေကို စမ်းပြီး run မယ်ဆိုရင်

```
$bar = new Bar();
$bar->printItem('baz.');
$bar->printPHP();
```

ဒီလို Code တွေရေးရပါမယ်။ Bar ဆိုတဲ့ Class အတွက်လည်း object တစ်ခုအောက်ရမှာပါ။ ဒါတွေရေးပြီးရင်တော့ run လိုပါပြီ။



Foo: baz.PHP is great.

Bar: baz.PHP is great.

ဒီလိုမျိုးတွေရမှာပါ။ နောက်တစ်မျိုးနေနဲ့ property တစ်ခုကို Foo class ထဲမှာ ထည့်လိုက်ပါမယ်။ ပြီးတော့ Function အသစ်လည်း ထပ်ရေးပါမယ်။

```
class Foo
{
    public $test = "";

    public function printItem($string)
    {
        echo 'Foo: ' . $string ;
    }

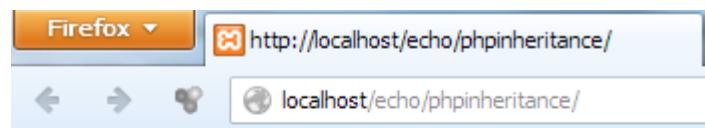
    public function printPHP()
    {
        echo 'PHP is great.</br>' ;
    }

    public function echoTest()
    {
        echo $this->test;
    }
}
```

အဲဒီ နှစ်ခုကိုရေးပြီးရင်တော့ run ဖို့ အတွက် Code အသစ်အနည်းငယ်ရေးပါမယ်။

```
$foo->test = 'Echo Test<br>';
$foo->echoTest();
$bar->test = 'Test Echo<br>';
$bar->echoTest();
```

ဒါဆိုရင်တော့ brower မှာ



Foo: baz.PHP is great.

Echo Test

Bar: baz.PHP is great.

Test Echo

ဒါဆိုရင် တော့ PHP ရဲ့ Object Inheritance အကြောင်းကို တစ်စွဲတစ်စွဲတောင်းလောက်သိပြီးထင်ပါတယ်။

Thank You

ယနေ့စွဲတွေမှာ Web Industry ရဲ့ ပြောင်းလဲမှုတွေဟာ အရမ်းကို မြန်ဆန်လွန်းလှပါတယ်။ {echo} အနေနဲ့ Web Design, Web Development နဲ့ပတ်သက်တဲ့ ဆောင်းပါးတွေ၊ Tutorial တွေကို ရူတောင့်ပေါင်းစုံကနေ အစွမ်းကုန်ကြီးဘား ဖော်ပြပေးသွားမှာဖြစ်ပါတယ်။ Web Design, Web Development နဲ့ပတ်သက်ပြီး အသက်မွေးဝမ်းကျောင်း ပြရလိုက်တွေအတွက် တစ်ဖက်တစ်လမ်းက အထောက်အကျော်စွဲ ရည်ရွယ်ပါတယ်။

သင့်ရဲ့ တန်ဖိုးရှိတဲ့ အခါန် ခကဗောဓာတ်ဖြတ်ကို {echo} နဲ့ ဖြတ်သန်းပေးလို့ ကျေးဇူးတင်ပါတယ်။

{echo} Magazine ၏ စာမျက်နှား၏ မူပိုင်ခွင့်များသည် မူရင်းစာရေးသူများသာ ပိုင်ဆိုင်ပါသည်။ စာရေးသူ၏ ခွင့်ပြုချက်မရရှိပဲ ထိုစာမျက်နှား (သို့မဟုတ်) ထိုစာမျက်နှား၏ တစိတ်တပိုင်းကို Printed Document အနေဖြင့် ဖြစ်စေ၊ နှစ်မှုလ {echo} ၏ PDF ကို ဖုန်းပြောင်းလဲ၍ ဖြစ်စေ၊ တစိတ်တပိုင်းအား ဖြတ်တောက်၍ ဖြစ်စေ၊ အေား Magazine များအတွင်း ပေါင်းစပ်၍ ဖြစ်စေ၊ ပြန်လည်ဝေမျှခွင့်၊ ပြန်လည်ထုတ်ဝေခွင့် မရှိပါ။ အွန်လိုင်းပေါ်တွင် ပြန်လည်ဝေမျှမည် ဆိုပါက နှစ်မှုလ Download Link အတိုင်း ဝေယှဉ်ပါရန် မေတ္တာရပ်ခံအပ်ပါသည်။