

கணியம்

கட்டற்ற மென்பொருள் பற்றிய மாத மின் இதழ்

டிசம்பர் 25 2012 - இதழ் 12

<http://www.kaniyam.com>



எழுதியோர்

ஸ்ரீனிவாசன்

சுகந்தி வெங்கடேஷ்

நித்யா

ஜோபின் பிராஞ்சல் ஆன்றனி.

தமிழ்

ஜான் கிறிஸ்டோபர்

ஓஜஸ்

இரா .சுப்ரமணி

தியாகராஜன் சண்முகம்

வடிவமைப்பு: ஸ்ரீனிவாசன்

பிழை திருத்தம்: இரா.சுப்ரமணி

நன்றி:

ஆமாச்சு - வலை தளம்

விக்னேஷ் நந்த குமார் - மக்கள் தொடர்பு

ஊக்கம்:

இந்திய லினக்ஸ் பயனர் குழு,

சென்னை ilugc.in

காஞ்சி லினக்ஸ் பயனர் குழு

kanchilug.wordpress.com

பொருளடக்கம்

ஃபயர்ஃபாக்ஸில் புதிய வசதி! -	4
எளிய செய்முறையில் C/C++ -	5
லைஃபோகிராஃப் (Lifeograph) தனிப்பட்ட மின்னணு நாட்குறிப்பேடு	8
Wallpaper சுழற்சிகள்	10
உபுண்டு முனையச் சிறுகுறிப்பு	12
CSS கற்றுக் கொள்ளுங்கள்!	13
பைதான் -பாகம் 5	17
2012 : லினக்ஸுக்கு என்னே ஒரு வருடம்!	24
MySQL- பாகம்: 4 - தகவல்களை சேமித்தல்	25
தேவாலயமும் சந்தையும்	32
ஓபன் சோர்ஸ் தொடர்பான பணிகளில் ஈடுபடும் நிறுவனங்கள்	37
கணியம் வெளியீட்டு விவரம்	40
கணியம் பற்றி...	41

வணக்கம்.

'கணியம்' இதழ் மூலம் உங்களை மீண்டும் சந்திப்பதில் பெருமகிழ்ச்சி அடைகிறோம்.
அனைவருக்கும் புத்தாண்டு வாழ்த்துக்கள்.

2012 ஆண்டில் கட்டற்ற மென்பொருட்கள் கணிப்பொறியை தாண்டி மொபைல் சாதனங்களை பெரிய அளவில் சென்றடைந்தன. ஆண்ட்ராய்டு இயங்குதளம் முன்னிலையில் இருந்தாலும், அதில் உள்ள பெரும்பான்மையான மென்பொருட்கள் தனியுரிம மென்பொருட்களே. ஆண்ட்ராய்டு இயங்குதளத்திலும் அதிக அளவில் கட்டற்ற மென்பொருட்களை உருவாக்கவும் பயன் படுத்தவும் வேண்டும். **Firefox OS** மற்றும் **ubuntu** போன்றவை மொபைல் சாதனங்களில் முழுதும் கட்டற்ற மென்பொருட்களை கொண்டு வரும் என தெரிவிக்கின்றன.

கணியம் இதழின் படைப்புகள் அனைத்தும் இணைய தளத்திலும் வெளியிடப்படுகின்றன. தளத்தை கண்டு உங்கள் கருத்துகளை பகிரவும். ஓபன் சோர்ஸ் தொடர்பான பணிகளில் ஈடுபடும் நிறுவனங்கள் பற்றிய விவரங்களை சேகரித்து வருகிறோம். உங்களுக்கு தெரிந்த தகவல்களை விக்கியில் பகிரவும்

உங்கள் அனைவரின் பேராதரவோடு கணியம், தனது முதல் ஆண்டை நெருங்க உள்ளது. அடுத்த ஆண்டில் பல்வேறு புதிய முயற்சிகளும் படைப்புகளும் நிகழ உள்ளன. இதற்கு மேலும் பல உதவிக்கரங்கள் தேவை. தமிழும் கட்டற்ற மென்பொருளிலும் ஆர்வம் உள்ள அனைவரது உதவியும் தேவை. வீடியோ பாடங்கள், கேள்வி பதில் தளம், நேரடி பயிற்சி பட்டறைகள், இணைய வழி பயிற்சிகள், அச்ச ஊடக கட்டுரைகள் என பல பணிகள் காத்துள்ளன. வாசகர் அனைவரையும் பங்களிக்க அழைக்கிறோம்.

கணியம் இதழின் படைப்புகள் அனைத்தும், கிரியேடிவ் காமன்ஸ் என்ற உரிமையில் வெளியிடப்படுகின்றன.

இதன் மூலம், நீங்கள் 0~யாருடனும் பகிர்ந்து கொள்ளலாம். ~0~ திருத்தி எழுதி வெளியிடலாம். ~0~ வணிக ரீதியிலும்யன்படுத்தலாம். ஆனால், மூல கட்டுரை, ஆசிரியர் மற்றும் www.kaniyam.com பற்றிய விவரங்களை சேர்த்து தர வேண்டும். இதே உரிமைகளை யாவருக்கும் தர வேண்டும். கிரியேடிவ் காமன்ஸ் என்ற உரிமையில் வெளியிட வேண்டும்.

நன்றி.

புனி ஆசிரியர், கணியம் editor@kaniyam.com



ஃபயர்ஃபாக்ஸில் புதிய வசதி!

ஏறக்குறைய இண்டர்நெட் எக்ஸ்ப்ளோரர் உலவி வழக்கொழிந்து போய் வருகிறது. கூகுளின் க்ரோமும், மொஸில்லாவின் ஃபயர்ஃபாக்ஸ் உலவியும் அனைவரின் கணினிகளையும் அலங்கரித்து வருவது கண்கூடு.

ஆயிரம் காரணம் சொன்னாலும், க்ரோமின் வேகத்திற்கு ஃபயர்ஃபாக்ஸ் ஈடுகொடுக்க முடியவில்லை என்பதும் ஏற்றுக்கொள்ளக் கூடியதே! சிறிது சிறிதாக ஃபயர்ஃபாக்ஸ் தனது வசதிகளை அதிகரித்துக் கொண்டே வருகிறது.

உலவிகளில் **Private Browsing** எனும் வசதி பற்றி எல்லோரும் அறிந்ததே. க்ரோம் உலவியைப் பொறுத்தவரை சாதாரணமாகவும், **Private Browsing**-கிலும் ஒரே நேரத்தில் பணியாற்ற வசதி உள்ளது.



ஃபயர்ஃபாக்ஸில் **Private Browsing Tab**-ஐ தேர்வு செய்தால் (Ctrl+Shift+P) உடனடியாக அப்போதைய **Session** மூடப்பட்டு விடும். இதனால் இரண்டிலும் ஒரே நேரத்தில் பணியாற்ற இயலாமல் போய்விடுகிறது.

இந்தக் குறைபாடு தற்போது ஓரளவு தவிர்க்கப்பட்டுள்ளது. ஃபயர்ஃபாக்ஸ் புதிதாக **Nightly** என்கிற பெயரில் உலவியை அறிமுகம் செய்துள்ளது. இதில் 'New Private Window'-ஐ தேர்வு செய்யும் போது, **Private Window** மற்றும் சாதாரண **window** என இரண்டிலும் ஒரே நேரத்தில் பணியாற்ற முடிகிறது.

இது ஓர் ஆரம்பம்தான். இன்னும் பல புதிய மாற்றங்கள் நிகழும் என நம்புவோம். **Nightly** பதிவிறக்க உரலி:

<http://nightly.mozilla.org/>

தமிழ் நல்ல வாசிப்பாளன். தமிழ் விரும்பி. வலைப்பதிவுகளில் எழுத முயற்சிக்கிறவன்.

எனது தளம்: <http://thamizhg.wordpress.com/> Thamizh G <iamthamizh@gmail.com>

எளிய செய்முறையில் C/C++

C/C++ - கணிப்பொறி மொழி இயல்பிலேயே மிகவும் எளிதான ஒரு மொழியே. இந்த தொடரில் நாம் எளிய செய்முறையில் இந்த மொழியை கற்றுக் கொள்வோம்.

முதலில் C/C++ நிரல்களை(Programs) இயக்க நமக்குத் தேவையான மென்பொருட்களைப் பார்ப்போம்.

1. ஒரு இயங்குதளம் (OS). இங்கு காணப்படும் நிரல்கள் அனைத்தும் சென்ட் ஓ.எஸ்-லினக்ஸ் (CentOS-Linux) என்னும் இயங்குதளத்தில் சரிபார்க்கப்பட்டது.
2. ஒரு தொகுப்பி(Compiler) – நாம் gcc என்னும் ஒரு தொகுப்பியை உபயோகிப்போம். இது எல்லா முழுமையான இயங்குதள நிறுவலிலும் காணப்படும்.

C - நிரலின் கட்டமைப்பு

ஆவணப் பிரிவு (Documentation Section)

இணைப்புப் பிரிவு(Link Section)

வரையறைப் பிரிவு (Definition Section)

முழுமை அளாவிய பிரிவு (Global Declaration Section)

Main செயல்கூறு (Main Function)

{

அறிவிப்புக் கூற்று (Declaration Statements)

செயல்பாட்டுக் கூற்று (Executable Statements)

}

துணை நிரல் பிரிவு (Function Definitions)

துணை நிரல்1() - Function1()

{

}

துணை நிரல்2 - Function2()

{

}

புதிதாக இந்த மொழியை கற்பவர்கள் இதைப் பார்த்து பயந்து போக வேண்டாம். நாம் ஒவ்வொன்றாக கற்க கற்க, இது மிகவும் எளிதாக புரியும்.

இந்தப் பிரிவுகளில் மிக முக்கியமாக இருக்க வேண்டிய இரு பிரிவுகள்

1. இணைப்புப் பிரிவு(Link Section)
2. Main செயல்கூறு(Main Function)

இணைப்புப் பிரிவு:

இந்தப் பிரிவில் இந்த நிரலியில் இணைக்க வேண்டிய தலைப்புக் கோப்புகளை (Header Files) சேர்க்க வேண்டும்.

எ.கா.

```
#include<stdio.h>
```

இங்கு **stdio.h** என்பது தலைப்புக் கோப்பு. இந்தக் கோப்பில் நிலையான உள்ளீடு, வெளியீட்டுக்கான கூற்றுகள் செயல்கூறுகளாக பதியப்பட்டிருக்கும்.

உள்ளீட்டிற்கான கூற்று:

```
scanf
```

வெளியீட்டிற்கான கூற்று:

```
printf
```

முதல் நிரல்

முதலில் நமது பெயரை எப்படி வெளியீடாக திரையில் காண்பிப்பது என்பதை பார்ப்போம்.

```
printf("MY NAME");
```

இங்கு "MY NAME" என்பதில் என்ன உள்ளதோ அது அப்படியே திரையில் தெரியும்.

முழுமையான நிரல்

```
#include<stdio.h>
int main()
{
    printf("MY NAME\n");
}
```

இதனை vi அல்லது emacs போன்ற ஏதேனும் ஒரு உரை திருத்தியில் எழுதி, '.c' எனும் நீட்டிப்பில் சேமிக்க வேண்டும்(எ.கா myname.c)

நிரலை செயல்படுத்த

1. முதலில் நிரலைத் தொகுப்பியில் கொடுத்து, தொகுக்க வேண்டும்

```
$ gcc myname.c
```

2.இது a.out என்னும் ஒரு கோப்பு ஒன்றை புதிதாக நிறுவியிருக்கும்.

3.இப்பொழுது ./a.out என்னும் கட்டளையைக் கொடுப்பதன் மூலம் உங்களது பெயரை திரையில் காணலாம்.

```
$ ./a.out
```

```
MY NAME
```

```
$
```

அடுத்த பதிப்பில்.....

\n என்பது என்ன?

உள்ளீடு (scanf) வாங்குவது எப்படி?

வேறு ஒரு பெயரில் இயங்கக்கூடிய(executable in other names than a.out) கோப்பை நிறுவுவது எப்படி?

இன்னும் சில.....



நான் செ.ஜான் கிறிஸ்டோபர், ஒரு Software Company யில் Team Leader ஆக வேலை செய்கிறேன்.

<http://tamilanjohn.blogspot.in/> <http://ilugdharmapuri.blogspot.in/>

sjchristopher@gmail.com

லைஃபோகிராஃப் (Lifeograph) தனிப்பட்ட மின்னணு நாட்குறிப்பேடு

லைஃபோகிராஃப் என்பது சுய குறிப்பெடுக்க உதவும் செயலி ஆகும். குறிப்பேடு செயலியிடமிருந்து நாம் எதிர்பார்க்கும் அனைத்தையும் நமக்கு இது திருப்திகரமாக வழங்குவதோடு, சில சிறப்பம்சங்களையும், குறைந்த அளவே உள்ள நிறுவும் தொகுப்பாக (installable package) தருகிறது.

சிறப்பம்சங்கள்:

- மறையாக்கம்(encryption) செய்த மற்றும் செய்யாத நாட்குறிப்பேடுகளுக்கு ஆதரவு அளிக்கிறது
- சிறிது நேரத்திற்கு மேல் பயன்படுத்தப்படாமல் இருந்தால் தானாகவே வெளியேற்றி (logs out automatically) விடும் (ஒரு வேளை நீங்கள் வெளியேற மறந்துவிட்டால், உங்கள் நாட்குறிப்பேட்டை பாதுகாக்கவே இந்த ஏற்பாடு) (எனினும் இது கட்டாய தேர்வு இல்லை).
- தலைப்புகள் மற்றும் உட்கலைப்புகளை தானாகவே வடிவமைக்கும் (இந்த யோசனையை வழங்கியதற்காக டாம்பாய்க்கு தான் நன்றி தெரிவிக்க வேண்டும்)
 - பொதுவான தேடல் / வடிகட்டி
 - வார்ப்புருகள் (editor themes)
 - விரும்பிய இடுக்கைகள் (favorite entries)
 - குறியீடுதல் (entry tagging)
 - பிழை சரிபார்த்தல் (spell checking)
 - விக்கி போன்ற மேம்பட்ட எழுத்து வடிவமைத்தல் (**bold**, *italic*, ~~strikeout~~, மேலும் பல)
 - இடுக்கைகளுக்குள் இணைப்பு - URI போன்ற இணைப்புகள் (<http://>, <file://>, <mailto://>, மேலும் பல)
 - தானியங்கி நகலாக்கம் (automatic backup)
 - தனி இடுக்கைகள் அல்லது குறிப்பேடு முழுவதும் உள்ள இடுக்கைகளை மாற்றலாம்

- ஒரு தனிப்பட்ட குறிப்பேடு அல்லது அனைத்து குறிப்பேடுகளையும் அச்சிடலாம்
- பொதுவான புள்ளியியல் அட்டவணை (basic statistical chart)

பின்வரும் வெளியீடுகளில் நாம் எதிர்பார்ப்பவை:

- இடுக்கைகளில் புகைப்பட இணைப்பு
- உட்பொருத்திகள் (plug-ins)

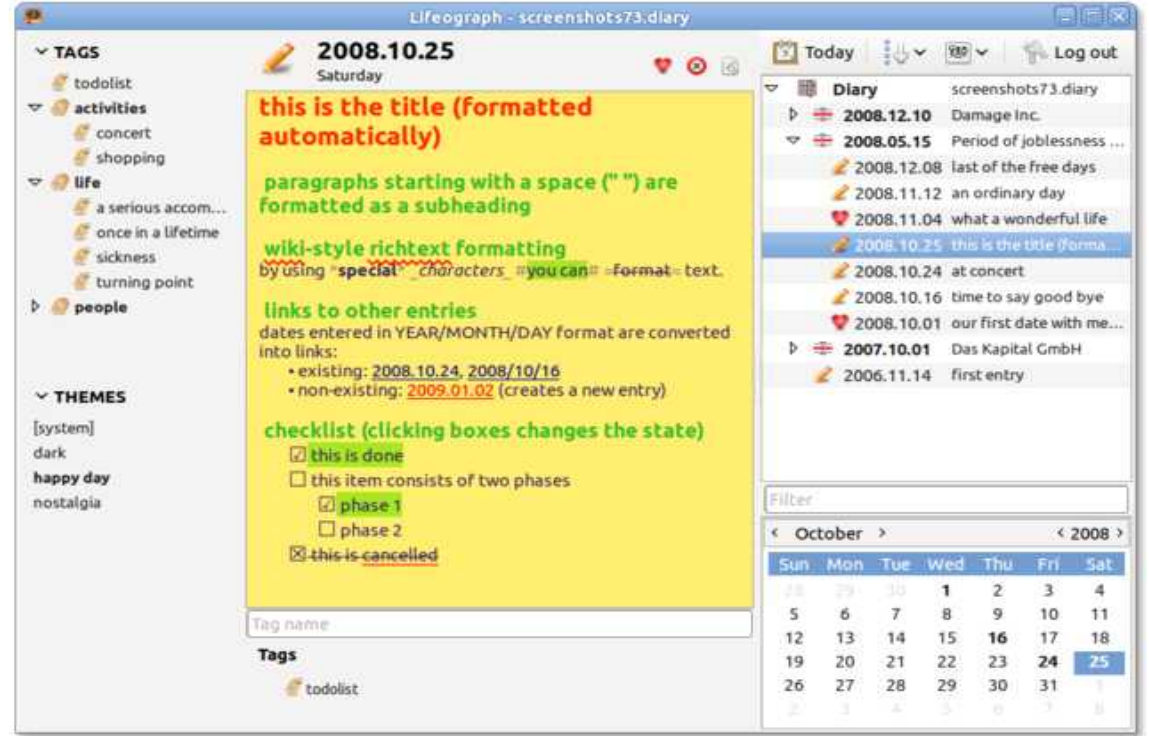
லைஃபோகிராஃப்-ஐ உபுண்டு 12.04 ல் நிறுவ,

முனையத்தில் பின்வரும் கட்டளைகளை தட்டச்சு செய்யவும்:

```
sudo add-apt-repository ppa:dmxe/ppa
```

```
sudo apt-get update
```

```
sudo apt-get install lifeograph
```



நான் ஜோபின் பிராஞ்சல் ஆன்றனி. நான் ஒரு CollabNet மென்பொருள் நிறுவனத்தில் வேலை செய்கிறேன். எனது சொந்த ஊர் நாகர்கோவில். கடந்த 2011 -ம் ஆண்டு கல்லூரி படிப்பை முடித்தேன். கணியம் மூலமாக உங்களை சந்தித்ததில் மகிழ்ச்சி. இந்த வாய்ப்பை கொடுத்த கணியம் ஆசிரியருக்கு நன்றி தெரிவித்து கொள்கிறேன்.

வலை பதிவு : <http://jophinepranjal.blogspot.in/>



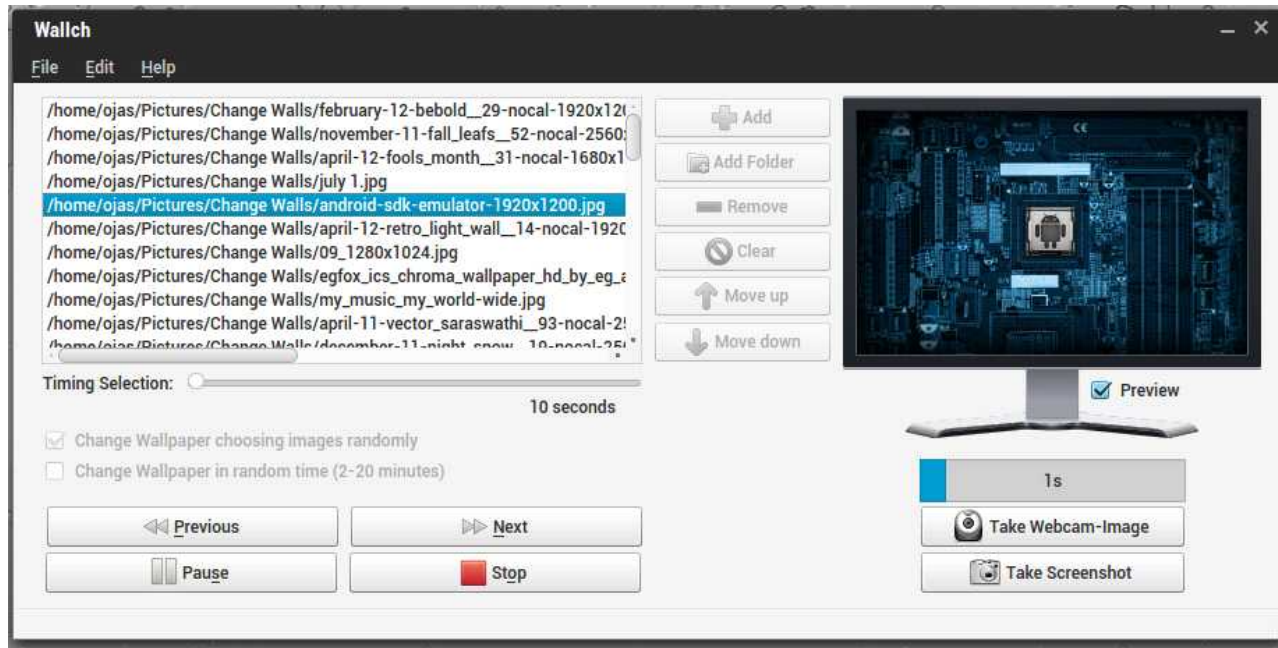
Wallpaper சுழற்சிகள்

விண்டோஸ் பயன்படுத்தி விட்டு, லினக்ஸ் பக்கம் வந்த பின், எனக்கு இருந்த மிகப் பெரும் குறை, **wallpaper**-ஐத் தான்தோன்றித்தனமாக(**random**), ஒரு குறிப்பிட்ட நேரத்துக்கு மாற்றும் வசதி இல்லை. இதற்குத் தகுந்த இலவச லினக்ஸ் மென்பொருள் தான் இந்த **WALLCH**. [<http://wall-changer.sourceforge.net/>]

இதனைப் பதிவிறக்க : <http://sourceforge.net/projects/wall-changer/files/latest/download?source=files>

பதிவிறக்கியக் கோப்பை நீங்கள் அழுத்தும் பட்சத்தில், நாம் **Ubunut Software Center**-க்கு அழைத்துச் செல்லப்படுவோம். அங்கு **Install** என்னும் பட்டனை அழுக்கினால் போதும்.

இதனை நாம் நிறுவிய பின், நமக்குத் தேவையான **wallpaper**-களை இதனுள் சேர்த்து விட்டு, சுழற்சிக்கான நேரத்தையும் மாற்றி விட வேண்டும். பின் இதனை இயக்கினால், தானாக நாம் சேர்த்த **wallpaper**-களெல்லாம் மாறி மாறி வரும்.



சிறப்பம்சங்கள் :

- **wallpaper**-களை நாம் ஆல்பமாக சேமித்து வைத்துக் கொள்ளலாம். அதனை நாம் எப்பொழுது வேண்டுமானாலும் இதில் பயன்படுத்திக் கொள்ளலாம்.
- படக் கோப்புகள் உள்ள **folder**-ஐ நேராக சேர்க்கலாம்.
- **wallpaper**-ஐ வரிசையாக அல்லது மாற்றி மாற்றி சுழல வைக்கலாம்
- சுழற்சி நேரத்தையும் நாம் தேர்வு செய்யலாம், அல்லது அதுவே மாற்றி மாற்றி அமைத்துக் கொள்ளும்.
- இந்த மென்பொருள் மூலம் **web cam** அல்லது **snapshot** எடுத்து, **wallpaper** சுழற்சியில் சேர்க்கலாம்
- வரலாறு ரொம்ப முக்கியம். இந்த மென்பொருள் நாம் **wallpaper**-களை மாற்றி அமைக்கும் (சேர்க்கும், நீக்கும்) பொழுது, தேதியையும் நேரத்தையும் குறித்து வைத்துக் கொள்கிறது. இதுவரை எவ்வளவு நேரம் இந்த மென்பொருள் இயங்கிக் கொண்டுள்ளது, எத்தனை முறை **wallpaper**-ஐ மாற்றி உள்ளது என்பதையும் கச்சிதமாக கணக்கிட்டுச் சொல்கிறது.
-
- எந்நேரமும் இணைய வசதி உள்ளவர்களுக்கு, உலகத்தின் செயற்கைக்கோள் படத்தை அல்லது விக்கி பீடியா 'photo of the day' படத்தை, அதுவாகவே எடுத்து **wallpaper**-ஆக வைக்கும். நம் கணினி இயங்க ஆரம்பித்த உடன், இதுவும் அதன் வேலையை ஆரம்பித்து விடும். இதனை நாம் கட்டுப்படுத்தலாம். நாம் விரும்பினால், ஒவ்வொரு **wallpaper** மாறும் போதும் ஒலியைச் சேர்த்துக் கொள்ளலாம். இதைத் தவிர , இந்த **WallCh** நிறுவனம் நமக்கென 1000 **wallpaper**-களை இலவசமாகத் தருகிறது. (சொடுக்கவும்) <http://wall-changer.sourceforge.net/downloads/desktop-wallpapers.html>
- நீங்களும் என்னைப் போல ஒரு **wallpaper** விரும்பியா? நீங்கள் நிச்சயம் பார்த்து பயன்படுத்த வேண்டிய தளம் -> wallbase.cc .
- இந்தத் தளம் **Creatvie Commons**-ல் உள்ளது! வண்ண வண்ண வால்பேப்பர்கள், மாற்றி மகிழுங்கள் !

பெயர்: ஓஜஸ் aoojass@gmail.com ட்விட்டர் : @oojass

வலைப்பூ : நாற்சந்தி - naarchanthi.wordpress.com

நான் அ.ஓஜஸ். உங்களுள் ஒருவன். உங்களைப் போல் ஒருவன். உலகத்தையும், இணையத்தையும், படித்து வருகிறேன்.

உபுண்டு முனையச் சிறுகுறிப்பு

கீழ்கண்டவற்றை உபுண்டு ஜினோம் முனையத்தில் தட்டச்சு செய்யுங்கள்:

```
sudo apt-get install fortunes-ubuntu-server -y
```

இனி உபயோகமான குறிப்புகள் பெற, பின்வரும் கட்டளையைப் பல முறை பயன்படுத்துங்கள்:

```
ubuntu-server-tip
```

இதோ சில எடுத்துக்காட்டுகள்:

lsdf பயன்படுத்தி எந்தெந்த கோப்புகளுக்கு ஓப்பன் ஹாண்டில் (**open handle**) உள்ளது என்பதை கண்டுபிடித்து விடலாம்.

'lsdf +D /path' பயன்படுத்தி கொடுக்கப்பட்டுள்ள தடத்திற்கான பிராசஸ்களை (**process**) கண்டுபிடித்து விடலாம்

நீங்கள் ஏதாவது தவறான கட்டளை அல்லது கடவுச்சொல் தட்டச்சு செய்துவிட்டால், பயன்பட வேண்டாம், **ctrl + u** பயன்படுத்தி முழு வரியையோ, **ctrl + w** பயன்படுத்தி ஒரு வார்த்தையையோ அழித்து விடலாம்.

நீங்கள் PostgreSQL தரவுத்தளம் (**database**) பயன் படுத்துபவராக இருந்தால், **ptop** பயன்படுத்தி நிகழ் நேர (**real time**) பயன்பாட்டை கண்காணிக்கலாம். அது போலவே //PostgreSQL-லில் இருந்து கோப்பிற்கு ஏற்றுமதி செய்வதும் எளிது.

முனையக் கட்டளை வரலாற்றை **ctrl+r** பயன்படுத்தி எளிமையாகத் தேடலாம், **ctrl+r** பயன்படுத்தியபின், கட்டளையில் இடம் பெற்றிருந்தது என நீங்கள் கருதும் சில வார்த்தைகளை பயன்படுத்தினால் உங்களுக்கான பதில் கிடைத்துவிடும்.

பாஷ்(**bash**) நிரல் தொடரியில் (**syntax**) வட்டமடித்தலை (**loop**) உருவாக்க:

```
"for i in ★; do echo $i; done"
```

தடத்தை இருமுறை தட்டச்சு செய்யாமல் நகலாக்கம் செய்ய:

"cp /long/path/to/file/name{,.orig}" இது **suffix.orig** உடன் ஒரு நகலை ஏற்படுத்தும். **ஜோபின் பிராஞ்சல் ஆன்றனி**

CSS கற்றுக் கொள்ளுங்கள்!

சுகந்தி வெங்கடேஷ்

இணைய வடிவமைப்பாளர் ஒவ்வொருவரும் கேள்விப்பட்டிருக்கும் ஒரு சொல் CSS. அப்படி என்றால் என்ன? CSS எப்படி செயல்படுகின்றது என்று பலருக்கு எவ்வளவு ஆழமாகத் தெரியும், என்பது எனக்குத் தெரியவில்லை. ஆனால் தமிழில் அதற்கான வழிகாட்டி இல்லை என்பதால் இதை எழுதுகிறேன்.

இணைய தளத்தை வடிவமைக்கப் பயன்படுத்தப்படும் மொழி HTML என்று அழைக்கப்படுகிறது. அதில் செயல்படுத்தப்படும் இழைகளை உலாவிகள் எவ்வாறு அமைக்க வேண்டும் என்று இணைய வடிவமைப்பாளர் எழுதிக் கொடுக்கும் மொழியே CSS. ஆங்கிலத்தில் அது Cascading Style Sheets என்று அழைக்கப்படுகிறது. தமிழில் 'விழுதொடர் வடிவ அறிக்கை' என்று அறியப்படுகிறது. Cascading Style Sheets-ல் எழுதப்படும் குறியீடுகளை செயல்படுத்தும் உலாவிகள் முதலில் எழுதப்பட்ட வரியை செயல்படுத்தி ஒன்றன் பின் ஒன்றாக, அடுத்தடுத்து வரும் வரிகளை செயல்படுத்தத் தொடங்குவதால் இக்குறியீடுகள் விழுதொடர் (Cascading) என்ற பெயரைப் பெறுகின்றன.

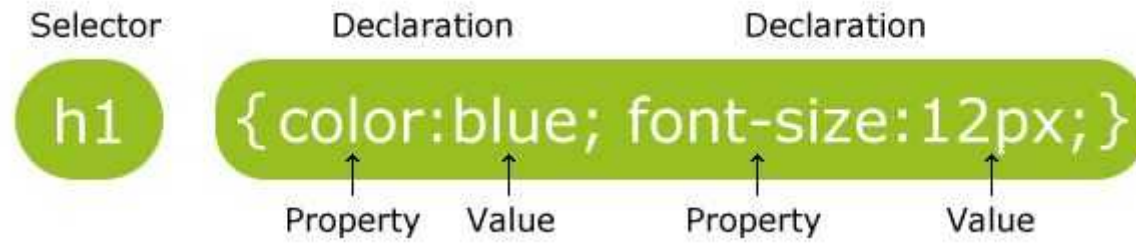
<http://w3schools.com> என்ற இணைய தளத்தில் சொல்லப்படுவதாவது HTML இழைகள் ஒரு இணையப் பக்கத்தின் உள்ளடக்கம்(content) என்னவாக இருக்க வேண்டும் என்று சொல்கிறதே தவிர, அவை எப்படி அமைய வேண்டும் என்று சொல்வதில்லை. உள்ளடக்கங்கள் எவ்வாறு அமைய வேண்டும் என்று சொல்லும் வேலையை விழுதொடர் வடிவ அறிக்கை(Cascading Style Sheets) செய்கிறது.

விழுதொடர் வடிவ அறிக்கை எப்படி செயல்படுகிறது என்று பார்ப்போம்.

இதை எழுதுவதற்கு இலவச உரைப்பான் notepad++ போதுமானது. குறிப்புகள் எழுத எழுத, அவை வண்ணங்கள் மூலம் நமக்குப் பிரித்துக் காட்டப்படும் ஆவணத்தைச் சேமிக்கும் போது CSS என்ற ஆவணமாக சேமித்தால் போதுமானது.

உலாவிகள் புரிந்து கொள்ளும் படி விழுதொடர் வடிவ அறிக்கையை எப்படி எழுதுவது என்று பார்ப்போம்.

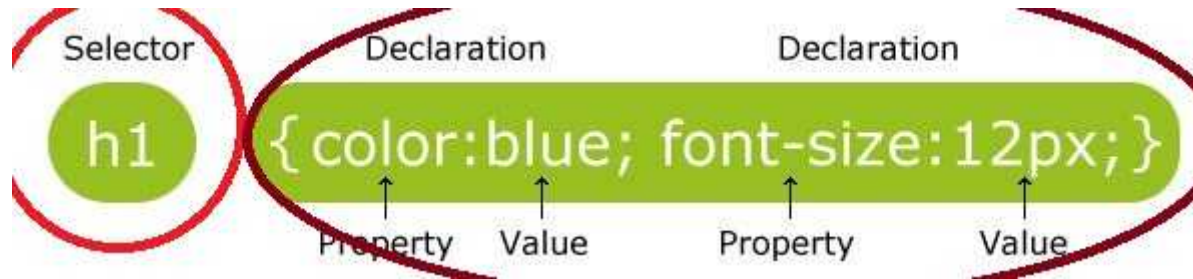
கீழே உள்ள படத்தைப் பாருங்கள்.



இதில் சொல்லியிருப்பது HTML-ல் <h1> என்று வரும் தலைப்பின் வண்ணம் நீலம்; எழுத்துருவின் அளவு 12px.

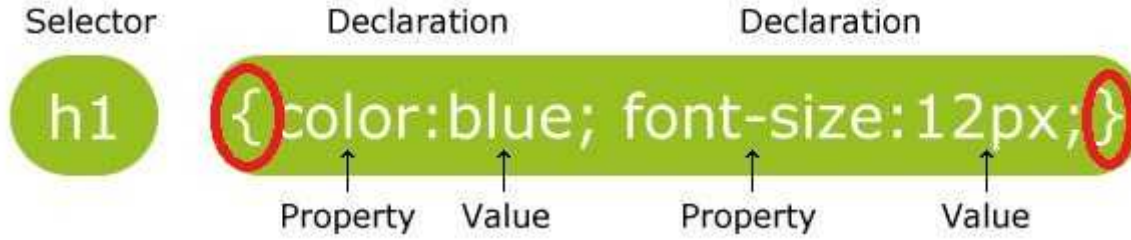
இந்த அறிவிப்பைக் கண்டதும், உலாவி இணையப் பக்கத்தில் தலைப்பு <h1> எனக் குறிக்கப்பட்டிருக்கும் எல்லா இழைகளுக்கும் முதலில் நீல வண்ணம் இடும். அதன் பின் அவற்றின் எழுத்துருவின் அளவை 12px ஆக மாற்றும். அதாவது தேர்ந்தெடுக்கப்பட்ட ஒரு இணைய இழையின் பண்புத் தன்மையை விவரிப்பது தான் விழுதொடர் வடிவ அறிக்கை (Cascading Style Sheets)யின் வேலை.

இந்தப் படத்தையே இன்னும் ஆழமாகப் பார்த்தால் விழுதொடர் வடிவ அறிக்கை அமைப்பு எப்படி இருக்க வேண்டும் என்று புரியும். ஒவ்வொரு விழுதொடர் வடிவ அறிக்கையும் இரண்டு பாகங்களைக் கொண்டது. முதல் பகுதி தேர்வி(selector), இரண்டாவது பகுதி பிரகடனம்(declaration). எடுத்துக் கொண்ட <h1> தான் இங்கு தேர்வி. அதன் பண்புகளான எழுத்தின் வண்ணமும், அதன் அளவும் அடுத்தடுத்து 'பிரகடனம்' பகுதியில் அறிவிக்கப்படுகின்றன.

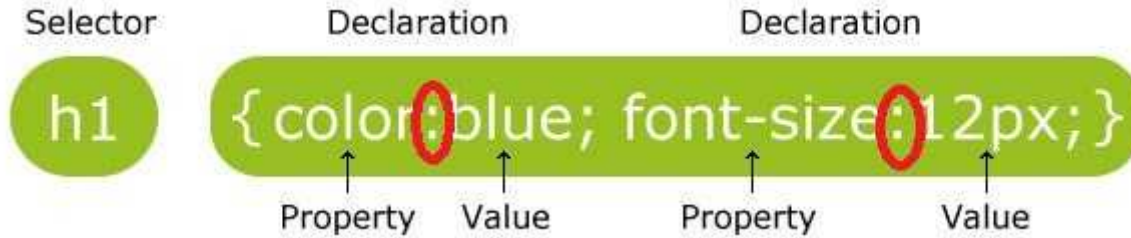


```
h1 {
color:blue;
font:12px;
}
```

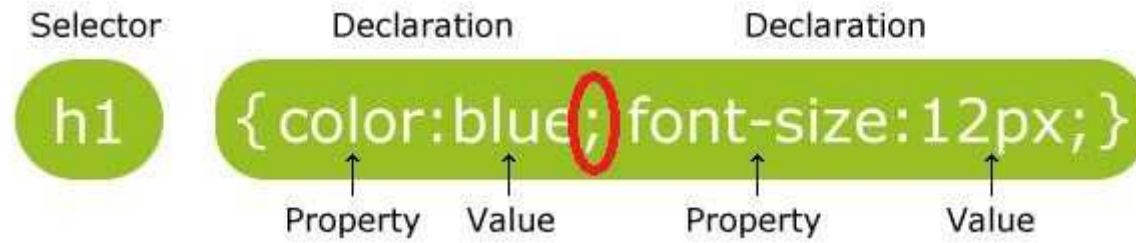

என்ற அறிவிப்பை எடுத்துக் கொண்டால் h1 என்ற தேர்வியையும், அதற்குரிய பிரகடனத்தையும் பிரிப்பது { }சுருண்ட அடைகுறிகள்.



அடைகுறிகளுக்கு உள்ளே வருவது தான் இழையின் பண்புத் தன்மையை அறிவிக்கும் பிரகடனம். இவை இரண்டு பிரிவுகளாக வருகின்றன. ஒரு பிரகடனத்தின் முதல் பகுதி இழையின் பண்புத் தன்மை. அடுத்து வருவது அதன் மதிப்பு இரண்டையும் பிரிப்பது முக்காற் புள்ளி :



ஒரு தேர்வியைக் குறிக்கும் போது, அதன் பல பண்புத் தன்மைகளை நாம் அறிவிக்க நேரிடலாம். ஒரு பண்புத் தன்மையையும், அடுத்த பண்புத் தன்மையையும் பிரித்து எழுத வேண்டும். அப்படி எழுதவில்லை என்றால் உலாவிகளால் புரிந்து கொள்ள முடியாது. பண்புத் தன்மைகளைப் பிரிக்க உதவுவது ; (அரைப்புள்ளி).



இது வரை உலாவிகளுக்குப் புரியும் படி எப்படி ஒரு விழுதொடர் அறிக்கை இருக்க வேண்டும் என்று பார்த்தோம். அப்படியானால் மனிதர்கள் தங்களுக்குத் தேவையான குறிப்பை எழுத முடியாதா? என்ற சந்தேகம் எழலாம்.

/★ ★/ என்ற அடைப்புக்குள் நாம் எழுதும் குறிப்புக்களை உலாவிகள் உதாசீனப்படுத்தி விடும். அவற்றை இணைய வடிவமைப்பாளர் தன் குறிப்புகளுக்காக பயன்படுத்திக் கொள்ளலாம்.

```

/* தலைப்பை அலங்கரிக்கும் விதம் */
h1 {
  color:blue;
  font:12px;
}

```

விழுதொடர் வடிவ அறிக்கையை எப்படி எழுதுவது என்று கண்டு கொண்டோம். அதை இணையப் பக்கத்தோடு இணைத்தால் தான் அது வேலை செய்யும்.

அடுத்த இதழில், எப்படி ஒரு விழுதொடர் வடிவ அறிக்கையை, ஓர் இணையப் பக்கத்தோடு இணைப்பது என்று பார்ப்போம்.

பைதான் - 5

(சென்ற வார தொடர்ச்சி...)

4.7 function – Arguments:

function-களை define செய்யும் போது பல்வேறு argument-களையும் பயன்படுத்தலாம். அதை மூன்று விதமாக செய்யலாம். அவை:

4.7.1 Default Argument Values:

ஒன்று அல்லது அதற்கு மேற்பட்ட argument-களுக்கு, நாமாக, default value-ஐ தருவது இந்த முறை. இதனால், அந்த function-ஐ பயன்படுத்தும் போது குறைவான argument-களுக்கு மட்டுமே, value தந்தால் போதுமானது.

```
def ask_ok(prompt, retries=4, complaint='Yes or no, please!'):
    while True:
        ok = raw_input(prompt)
        if ok in ('y', 'ye', 'yes'):
            return True
        if ok in ('n', 'no', 'nop', 'nope'):
            return False
        retries = retries - 1
        if retries < 0:
            raise IOError('refusenik user')
        print complaint
```

இந்த function ஐ கீழ்க்கண்டவாறு பயன்படுத்தலாம். ask_ok('Do you really want to quit?') அல்லது ask_ok('Delete file', 2)

இந்த உதாரணம் in என்ற keyword ஐ அறிமுகம் செய்கிறது. இது தரப்பட்ட ஒரு sequence ல் ஒரு குறிப்பிட்ட value இருக்கிறதா? இல்லையா? என்பதை சொல்கிறது.

Default value கள் ஒரு function அழைக்கும் போது மதிப்பு பெற்று இயக்கப் படுகின்றன. இதை விளக்கும் உதாரணம் இதோ:

```

i = 5

def f(arg=i):
    print arg

i = 6
f()

```

கவனிக்க: default value ஆனது ஒரே ஒரு முறை மட்டுமே மதிப்பிடப் படுகிறது. இது மாறும் தன்மை கொண்ட list , dictionary, class களின் instance போன்றவற்றில் பயன்படுகிறது. உதாரணமாக, கீழ்வரும் உதாரணம், தரப்படும் argument களை, அடுத்தடுத்த call களின் போது, அவற்றை ஒரே list ல் இடுகிறது.

```

def f(a, L=[]):
    L.append(a)
    return L

print f(1)
print f(2)
print f(3)

```

இதன் output

```

[1]
[1, 2]
[1, 2, 3]

```

default value ஆனது அடுத்தடுத்த call களில் பயன்பட வேண்டாம் என்றால், பின்வருமாறு எழுதவும்

```
def f(a, L=None):
    if L is None:
        L = []
    L.append(a)
    return L
```

4.7.2 Keyword Arguments:

keyword=value என்ற முறையால், குறிப்பிட்ட keyword க்கான value ஐ default ஆக function argument ல் தரலாம். கீழ்வரும் உதாரணத்தை காண்க.

```
def parrot(voltage, state='a stiff', action='vroom', type='Norwegian Blue'):
    print "-- This parrot wouldn't", action,
    print "if you put", voltage, "volts through it."
    print "-- Lovely plumage, the", type
    print "-- It's", state, "!"
```

இதை கீழ்வரும் பல்வேறு முறைகளில் பயன்படுத்தலாம்.

```
parrot(1000) # 1 positional argument
parrot(voltage=1000) # 1 keyword argument
parrot(voltage=1000000, action='VOOOOOM') # 2 keyword arguments
parrot(action='VOOOOOM', voltage=1000000) # 2 keyword arguments
parrot('a million', 'bereft of life', 'jump') # 3 positional arguments
parrot('a thousand', state='pushing up the daisies') # 1 positional, 1 keyword
```

ஆனால், கீழ்வரும் முறைகளில் பயன் படுத்த முடியாது.

```
parrot() # required argument missing
parrot(voltage=5.0, 'dead') # non-keyword argument after a keyword argument
parrot(110, voltage=220) # duplicate value for the same argument
parrot(actor='John Cleese') # unknown keyword argument
```

keyword மற்றும் value களை எந்த வரிசையில் வேண்டுமானாலும் தரலாம். function ஐ define செய்த அதே வரிசையில் தால் தர வேண்டும் என்பதில்லை.

ஒரே argument க்கு பல முறை value தர முடியாது. ஒரே ஒரு முறை மட்டும் தர முடியும். சாதாரண parameter ன் பெயரையே positional parameter ஆகவும் பயன் படுத்த முடியாது. இப்படி செய்தால் பின்வருமாறு பிழை ஏற்படும்.

```
>>> def function(a):
...     pass
...
>>> function(0, a=0)
Traceback (most recent call last):
  File "<stdin>", line 1, in ?
TypeError: function() got multiple values for keyword argument 'a'
```

**name என்ற முறையில் parameter களை பெறும் போது. **name ஐ தவிர, பிற keyword argument ஒரு dictionary ஆக பெறப் படுகின்றன.

*name என்று பயன்படுத்தும் போது, positional arguments ஒரு tuple ஆக பெறப் படுகின்றன . *name ஆனது **name க்கு முன்பாக வரவேண்டும்.

பின் வரும் function ஐ பாக்கவும்.

```
def cheeseshop(kind, *arguments, **keywords):
    print "-- Do you have any", kind, "?"
    print "-- I'm sorry, we're all out of", kind
    for arg in arguments:
        print arg
    print "-" * 40
    keys = sorted(keywords.keys())
    for kw in keys:
        print kw, ":", keywords[kw]
```


இதை பின்வருமாறு பயன்படுத்தலாம்

```
cheeseshop("Limburger", "It's very runny, sir.",
           "It's really very, VERY runny, sir.",
           shopkeeper='Michael Palin',
           client="John Cleese",
           sketch="Cheese Shop Sketch")
```

இதன் output

```
-- Do you have any Limburger ?
-- I'm sorry, we're all out of Limburger
It's very runny, sir.
It's really very, VERY runny, sir.
-----
client : John Cleese
shopkeeper : Michael Palin
sketch : Cheese Shop Sketch
```

keyword argument ன் பெயர்களை sort() மூலம் ஒழுங்கு படுத்தியதை கவனிக்கவும். இது keyword dictionary ஐ print செய்வதற்கு முன்பே sort செய்கிறது. இப்படி செய்யவில்லை என்றால், argument களை print செய்யும் order ஐ வரையறுக்க முடியாது.

4.7.3 எண்ணற்ற argument கள்:

சில நேரங்களில் function களுக்கு எண்ணற்ற argument களை தர நேரிடலாம். runtime ல் இந்த argument களை பெறலாம். இவை ஒரு type ஆக பெறப்படும். இதே functional ல் சாதாரண argument ஐ கூட பயன்படுத்தலாம்.

```
def write_multiple_items(file, separator, *args):
    file.write(separator.join(args))
```

4.7.4 Argument List ஐ பிரித்தல்:

சில நேரங்களில் மேலே சொன்னதற்கு எதிரான நிலையும் ஏற்படலாம். அதாவது argument கள் ஒரு tuple ஆகவோ list ஆகவோ இருக்கலாம். அவற்றை unpack செய்து ஒரு function க்கு தரலாம். உதாரணமாக range() என்ற inbuilt function க்கு start மற்றும் stop என்ற argument களை தரவேண்டும். அவற்றை தனித்தனியாக தரவில்லை என்றால், அந்த function ஐ * என்ற operator உடன் தந்து. argument களை list அல்லது tuple ல் இருந்து பிரித்தெடுக்கலாம்.

```
>>> range(3, 6)           # normal call with separate arguments
[3, 4, 5]
>>> args = [3, 6]
>>> range(*args)         # call with arguments unpacked from a list
[3, 4, 5]
```

இதே போல, dictionary ல் இருந்து keyword argument களை பிரித்து எடுக்க '**' என்ற operator ஐ பயன்படுத்தவும்.

```
>>> def parrot(voltage, state='a stiff', action='vroom'):
...     print "-- This parrot wouldn't", action,
...     print "if you put", voltage, "volts through it.",
...     print "E's", state, "!"
...
>>> d = {"voltage": "four million", "state": "bleedin' demised", "action": "VOOM"}
>>> parrot(**d)
-- This parrot wouldn't VOOM if you put four million volts through it. E's bleedin' demised !
```

4.7.5 lambda அமைப்பு:

functional programming என்ற ஒரு முறை Lisp போன்ற மொழிகளில் மிகவும் பிரபலம் அடந்து வருகிறது. அதன் பயன்கள் காரணமாக பைதான் மொழியிலும் functional programming தன்மைகளை இணைத்துள்ளனர். lambda என்ற keyword மூலம் சிறிய பெயரில்லாத function களை உருவாக்க முடியும். இரு argument களின் கூட்டுத் தொகையை return செய்யும் ஒரு function இதோ.

'lambda a, b : a+b'

function object கள் தேவைப்படும் இடங்களில் இந்த lambda function களை பயன்படுத்தலாம். syntax ன் படி அவை ஒரே ஒரு expression மட்டுமே. ஆனால், சாதாரண பிற function கள் போலவே செயல் படுகின்றன. Nested function போலவே lambda function களும் தமது scope க்கு உட்பட்ட variable களை பயன்படுத்துகின்றன.

```
>>> def make_incrementor(n):
...     return lambda x: x + n
...
>>> f = make_incrementor(42)
>>> f(0)
42
>>> f(1)
43
```

4.7.6 Documentation Strings:

Function மற்றும் class எழுதும் போது. அதன் பண்பு மற்றும் பயன்களை விவரிப்பது அவசியம். இது பிறர் படிப்பதற்கும், சிறிது காலம் கழித்து நாமே படித்து புரிந்து கொள்வதற்கும் மிகவும் உதவும்.

சாதாரணமாக comment போல இல்லாமல், இந்த document களுக்கு தனியாக formatitng முறைகள் உள்ளன.

முதல் வரி சுருக்கமாக இருக்க வேண்டும். அந்த function அல்லது object ன் பயனை சொல்ல வேண்டும். பெயர் மற்றும் type ஐ சொல்ல தேவையில்லை. முதல் எழுத்து capital letter ஆக இருக்க வேண்டும். கடைசியில் முற்றுப்புள்ளி வைக்க வேண்டும்.

இதில் பல வரிகள் இருக்கும் என்றால் இரண்டாவது வரி வெற்று வரியாக, காலியாக இருக்க வேண்டும். சுருக்கமான வரியையும் விளக்க உரையையும் பிரித்து காட்ட வேண்டும். விளக்க உரையில் விரிவாக object ன் பயன்கள், பயன்படுத்தும் முறைகள் பற்றி பல பத்திகளில் எழுதலாம்.

multiline doc string க்கு ஒரு உதாரணம் கீழே தரப்பட்டுள்ளது.

```
>>> def my_function():
...     """Do nothing, but document it.
...     No, really, it doesn't do anything.
...     """
...     pass
...
>>> print my_function.__doc__
Do nothing, but document it.

    No, really, it doesn't do anything.
```

தொடரும்...
(அடுத்த இதழில் டேட்டா ஸ்ட்ரக்சர்)

2012 : லினக்ஸுக்கு என்னை ஒரு வருடம்!

2012 ஆம் ஆண்டில் லினக்ஸின் சாதனைகளை விளக்கும் காணொளி ஒன்றை, லினக்ஸ் நிறுவனம் (The Linux Foundation) இணையத்தில் வெளியிட்டுள்ளது. 2:38 நிமிடங்கள் ஓடக்கூடிய இந்தக் காணொளி, கடந்த 12 மாதங்களில் லினக்ஸ் கடந்து வந்த மைல்கற்களையும், முக்கிய நிகழ்வுகளையும் நம் பார்வைக்குக் கொண்டு வருகிறது. அவற்றில் சில:-

- ஆன்ட்ராய்டின் அபரிதமான வளர்ச்சி
- கூகுள் அறிமுகம் செய்த லினக்ஸ் மடிக்கணினி 'Chromebook'
- லினக்ஸ் நிறுவனத்தில் சாம்சங் (Samsung) மற்றும் HP உறுப்பினர்களானது

காணொளியைக் காண:- <http://www.youtube.com/watch?v=Unfx2qCj6Ao>

இரா .சுப்பிரமணி subramani95@gmail.com

MySQL- பாகம்: 4 - தகவல்களை சேமித்தல்

Library எனும் மாதிரி **Databases**-ல் **data**-வை செலுத்துதல்

book எனும் **table**-ல் **data**-வை செலுத்துதல்:-

ஒரு **table**-ல் **data**-வை செலுத்துவதற்கு பல்வேறு வழிமுறைகள் இருந்தாலும், பொதுவாக **insert statement** மூலம் செலுத்துவது நடைமுறையில் அதிகமாகப் பயன்படுத்தப்படுகிறது. இதற்கான **syntax** பின்வருமாறு.

```
INSERT [INTO] [db_name.]table_name
(list, of, columns, ...)
VALUES (list, of, values, ...)
```

இந்த **insert statement** ஆனது **table**-ன் இறுதியில் ஒரு **row**-வை **insert** செய்யும். அதன் பின்னர் கொடுக்கப்படும் மதிப்புகளை அந்த **row**-வில் செலுத்திவிடும்.

book எனும் **table**-ல் ஒரு **row** எவ்வாறு செலுத்தப்படுகிறது என்பதை பின்வரும் உதாரணத்தின் மூலம் காணலாம்.

```
INSERT book (author, title, cond)
VALUES ('Stephen R. Covey',
'The 7 Habits Of Highly Effective People ',
'good');
```

இதில் **Insert** எனும் **keyword**-ஐத் தொடர்ந்து நேரடியாக “**book**” எனும் **tablename** கொடுக்கப்பட்டுவிட்டது. இடையில் இருக்கும் **INTO** எனும் **optional keyword** நீக்கப்பட்டிருப்பதைக் காணலாம். அடுத்ததாக **book** எனும் **tablename**-ஐத் தொடர்ந்து, எந்தெந்த **column**-ல் மதிப்புகள் செலுத்தப்பட வேண்டும் எனும் **column list** கொடுக்கப்பட்டுள்ளது. ஒரு **row**-ஆனது **insert** செய்யப்படும்போது இந்த **list**-க்குள் இல்லாத **columns**-க்கு **default** மதிப்புகள் **insert** செய்யப்படும்.

இந்த **list**-க்குள் “**book_id**” எனும் **primary key column**-ஐக் கொடுக்கத்தேவையில்லை. ஏனெனில் ஒவ்வொரு **row**, **insert** செய்யப்படும்போதும் இந்த **column**-ஆனது **auto-increment** மூலம் மதிப்புகளைப் பெற்றுவிடும் எனவே இந்த **column**-ஐ வெளிப்படையாக **list**-க்குள் கொடுக்கத் தேவையில்லை.

இந்த **list**-ம் **optional**-ஆகவே பயன்படுத்தப்படுகிறது. இந்த **list** கொடுக்கப்படவில்லை என்றாலும், மதிப்புகள் அனைத்தும் **table**-ல் இருக்கும் **columns**-ல் தொடர்ச்சியாக செலுத்தப்பட்டுவிடும். அனால் எந்த மதிப்புகள் எந்த **column**-ல் செலுத்தப்படுகின்றன எனும் விஷயத்தில் பெரும் குழப்பம் ஏற்பட நேரிடும். எனவே இதனைத் தவிர்ப்பதற்காக இவ்வாறு வெளிப்படையாக கொடுத்து விடுவது நல்லது.

கடைசியாக **VALUES** எனும் **keyword**-ஐத் தொடர்ந்து எந்தெந்த மதிப்புகள் **table**-ல் செலுத்தப்பட வேண்டுமோ அந்த மதிப்புகள் அனைத்தும் ஒரு **parenthesis**-க்குள் கொடுக்கப்படும். இந்த **parenthesis**-க்குள் இருக்கும் ஒவ்வொரு மதிப்பும் ஒரு **comma**-ஆல் வேறுபடுத்தப்பட்டிருக்கும். மேலும் இந்த மதிப்புகள் எழுத்துக்களாக இருந்தால் அதனை **quotes**-க்குள் கொடுக்க வேண்டும். என்களாக இருந்தால் அது தேவையில்லை.

அடுத்ததாக ஒன்றுக்கும் மேற்பட்ட **rows**-ஐ **table**-ல் செலுத்துவதற்கு, மதிப்புகளைக் கொண்டுள்ள **parenthesis**-ஐ தொடர்ச்சியாக **comma**-ஆல் இணைத்துக் கொண்டே செல்லலாம். இறுதியாக உள்ள **parenthesis**-ன் முடிவில் **semicolon(;** -ஐ அமைப்பதன் மூலம் நாம் கடைசியாக **insert** செய்யப்படும் **row**-வை சுட்டிக்காட்ட முடியும்.

பின்வரும் உதாரணத்தில் **book** எனும் **table**-ல் எவ்வாறு 3 **rows** செலுத்தப்படுகின்றன என்பதைக் காணலாம்.

```
INSERT book (author, title, cond)
VALUES ('Stephen R. Covey',
'The 7 Habits Of Highly Effective People',
'mint'),
('Amish Tripathi',
'The Immortals of Meluha ',
'poor'),
('Mitch Albom', 'Tuesdays with Morrie', 'good');
```

தனித்தனியாக ஒவ்வொரு **query** எழுதுவதற்கு பதிலாக மேற்கூறிய முறை மூலம் நாம் ஒரே **query**-ல் திறமையாக பல **rows**-ஐ **insert** செய்யலாம்.

Person எனும் **table**-ல் **data**-வை செலுத்துதல்:-

அடுத்ததாக **person** எனும் **table**-ல் புத்தகங்களைப் பெறவிரும்பும் நபர்களின் விவரங்களைப் பின்வருமாறு **insert** செய்யலாம்.

```
INSERT person (name, email)
VALUES ('Nithya', 'nithya@example.com');
```

loan எனும் **table**-ல் **data**-வை **insert** செய்தல்:-

இதில் **data**-வை **insert** செய்வதற்கு நமக்கு **book** மற்றும் **person** எனும் இரண்டு **table**-ன் உதவியும் தேவை. அதாவது "நித்யா" எனும் நபர் "Tuesdays with Morrie" எனும் புத்தகத்தைப் பெற விரும்புகிறார் எனில் முதலில் இந்த நபர் மற்றும் புத்தகத்தின் **primary key**-யை

நாம் தெரிந்து கொள்ள வேண்டும். அதன் பின்னர் தான் இந்த **primary key**-யைப் பயன்படுத்தி மதிப்புகளை **loan** எனும் **table**-ல் செலுத்த வேண்டும்.

இது பின்வருமாறு.

நித்யா எனும் நபரின் **primary key**-யைத் தெரிந்து கொள்ள, **query**-யைப் பின்வருமாறு அமைக்கவும்.

```
SELECT person_id FROM person WHERE name = 'Nithya';
```

இதன் **output** பின்வருமாறு அமையும்.

```
+-----+
| person_id |
+-----+
| 1         |
+-----+
```

1 row in set (0.07 sec)

அடுத்ததாக "Tuesdays with Morrie" எனும் புத்தகத்தின் **primary key**-யைத் தெரிந்து கொள்ள, **query**-யைப் பின்வருமாறு அமைக்கவும்.

```
SELECT book_id FROM book
WHERE title = 'Tuesdays with Morrie';
```

இதன் **output** பின்வருமாறு அமையும்.

```
+-----+
| book_id |
+-----+
| 2      |
+-----+
```

1 row in set (0.07 sec)

இப்போது இந்த இரண்டு மதிப்புகளையும் வைத்துக்கொண்டு நித்யா எந்தத் தேதியில் "Tuesdays with Morrie" எனும் புத்தகத்தைப் பெற்றுள்ளார் எனும் விவரத்தை **loan** எனும் **table**-ல் பின்வருமாறு செலுத்தலாம்.

```
INSERT loan (book_id, person_id, date_lent)
VALUES (2, 1, '2005-07-25');
```

இவ்வாறு மதிப்புகளை நேரடியாக **table**-ல் செலுத்தாமல் அதனை ஓர் **user variable**-க்கு **assign** செய்துவிட்டு, அந்த **user variable**-ஐ நாம் **query**-ல் பயன்படுத்தலாம்.

ஒரு **user variable**-க்கான **format**, **@name** என்று இருக்கும். இந்த **name** எனும் இடத்தில் நாம் **A-Z** வரையிலான எழுத்துக்களையோ, **0-9** முடிய இருக்கும் எண்களையோ அல்லது **\$**, -போன்ற குறிகளையோ பயன்படுத்தி ஓர் **user variable**-ஐ உருவாக்கலாம்.

இவ்வாறு உருவாக்கப்பட்ட ஓர் **variable**-க்கு **:=** எனும் **operator**-ஐப் பயன்படுத்தி மதிப்புகளை **assign** செய்யலாம். உதாரணம்:

```
SELECT @bookname := 'Tuesdays with Morrie';
```

இதில் **Tuesdays with Morrie** எனும் மதிப்பை உள்ளடக்கிய ஓர் **@bookname** எனும் **user variable** உருவாக்கப்பட்டுள்ளது. இப்போது பின்வரும் **query**-ஐ அமைத்து, **run** செய்தால் அந்த **variable** அதன் மதிப்பினை வெளிப்படுத்தும்.

```
SELECT @bookname;
```

இவ்வாறாக நாம் **loan** எனும் **table**-ல் **user variable**-ஐப் பயன்படுத்தி , மதிப்புகளை பின்வருமாறு செலுத்தலாம்.

```
SELECT @person_id := person_id FROM person
WHERE name = 'Nithya';
SELECT @book_id := book_id FROM book
WHERE title = 'Tuesdays with Morrie';
SELECT @date := '2005-07-25';
INSERT loan (book_id, person_id, date_lent)
VALUES (@book_id, @person_id, @date);
```

இவ்வாறு **user variable** மூலம் **data**-ஆனது **table**-க்கு செலுத்தப்படும்போது, தெரியாமல் ஏற்படும் தவறுகள் தவிர்க்கப்படுகின்றன.

மேலும் இங்கு தனித்தனியான **select statement**-ஐப் பயன்படுத்தி **user variable** உருவாக்கப்பட்டுள்ளது. இதனை ஒரே **select statement**-ஐப் பயன்படுத்தியும் நாம் பின்வருமாறு உருவாக்கி அதனை **table**-ல் **insert** செய்யலாம்.

```
SELECT @book_id := book_id,
@person_id := person_id,
@date := '2005-07-25'
```

```
FROM book, person
WHERE book.title = 'Tuesdays with Morrie'
AND person.name = 'Nithya';

INSERT loan (book_id, person_id, date_lent)
VALUES (@book_id, @person_id, @date);
```

இந்த இரண்டு queries-ஐயும் கூட நாம் user variables ஏதும் இல்லாமல் ஒரே query-யாக மாற்றி இன்னும் optimize செய்யலாம். இது பின்வருமாறு.

```
INSERT loan (book_id, person_id, date_lent)
SELECT book_id, person_id, '2005-07-25'
FROM book, person
WHERE book.title = 'Tuesdays with Morrie'
AND person.name = 'Nithya';
```

இவ்வாறாக நாம் ஒரு விஷயத்தையே பல்வேறு வழிவகைகளில் வெவ்வேறாகச் செய்யலாம் என இப்போது கற்றுக் கொண்டோம். மேலும் ஒரு பெரிய query-யானது பல்வேறு எளிய queries-ஐக் கொண்டே உருவாக்கப்படுகிறது எனவும் தெரிந்து கொண்டோம்.

- நித்யா இவர் cognizant நிறுவனத்தில் Data Warehouse Testing-ல் பணியாற்றி வருகிறார்.
- மின்னஞ்சல் : nithyadurai87@gmail.com வலை : <http://nithyashrinivasan.wordpress.com>



தேவாலயமும் சந்தையும்

எரிக் எஸ் ரேமண்ட்

Home page of Eric S Raymond

Original version of "The cathedral and the bazaar "

தமிழில் : தியாகராஜன் சண்முகம் citizenofgnu@gmail.com

முகவுரை

லினக்ஸ், ஒரு புரட்சிகரமான படைப்பு. இணையத்தின் வாயிலாக புவியின் பல்வேறு பகுதியில் இருந்து மென்பொருள் படைப்பாளிகள் உருவாக்கிய ஓர் இயங்குதளம் (Operating system). லினக்ஸுக்கு 1993 ஆம் வருடம் நான் அறிமுகமானேன். இதற்கு முன்பு பத்து வருடங்களாக நான் யுனிக்ஸ் மற்றும் திறமூல மென்பொருள் (Open Source Software) வளர்ச்சியில் என்னை அர்ப்பணித்துக் கொண்டிருந்தேன். திறமூல மென்பொருட்களின் வளர்ச்சி பாதையில் 1980 ஆம் ஆண்டில் இருந்த படைப்புகளில் எனக்கும் பங்கு உண்டு. இணையத்தின் வாயிலாக நெட் ஹேக்(nethack), விசி(Emacs's VC), குட் மோட்(GUD modes), எக்ஸ்-லைப்(xlife) மற்றும் வேறு சில மென்பொருட்களில் நான் படைப்பாளியாகவும், இணை படைப்பாளியாகவும் பணியாற்றியுள்ளேன். இதன் காரணமாக இந்த மென்பொருட்கள் எப்படி உருவாக்கப்பட்டன என்று எனக்கு நன்றாகத் தெரியும். யுனிக்ஸ் மற்றும் அதன் சார்ந்த சிறு மென்பொருள் உபகரணங்கள் உருவாக்கும் முறையை நான் பறைசாற்றிக் கொண்டிருந்த வேளையில் லினக்ஸ் உருவாக்கும் முறை என் சித்தாந்தங்களை மாற்றி அமைத்தது. இயங்குதளங்கள் மற்றும் இமேக்ஸ் போன்ற மென்பொருட்கள் உருவாக்கும் முறையும், அவைகள் ஒருங்கிணைந்த மென்பொருள் வடிவமைக்கும் முறையும் (Integrated Software Development), மென்பொருட்கள் முழுமையாக வடிவமைக்கும் வரை அதன் பிரதிகள் வெளியிடாமல் அவை முழுமையடைந்த பின் அதனை வெளியிடுதல். இவ்வண்ணம் மென்பொருள் உருவாக்கும் முறைகள் என் சித்தாந்தங்களாக இருந்தன . இவைகளை லினக்ஸ் உருவாக்கும் முறை மாற்றி அமைத்தது.

ஒரு தேவாலயம் கட்டி முடிக்கும் வரை அதன் உள்ளே யாரும் பிரவேசிக்க அனுமதி இல்லை. அது போன்று மென்பொருள் உருவாக்கி முடியும் வரை, அதன் பிரதிகளான பீட்டா காப்பியை வெளியிடுவதில்லை. இந்த நிலையை லினஸ் டோர்வால்ட்ஸ் மாற்றி அமைத்தார். அவ்வப்போது உருவாக்கப்படும் பகுதி மென்பொருட்களை வெளியிடுதல், இணையத்தின் வாயிலாக யாவரிடமிருந்தும், அவர்கள் உருவாக்கிய பகுதி மென்பொருட்களை வரவேற்று, அவைகளை ஒன்றுபடுத்தி நிலையானதொரு மென்பொருளை வடிவமைப்பது. இது சந்தையில் நமக்குத் தேவையான பொருட்களை பல கடைகளில் இருந்து வாங்குவதைப் போன்றதொரு தோற்றம் அளித்தது. இந்த முறையில் லினக்ஸ் குழப்பமின்றி வளர்ந்தது. இது தேவாலய முறையில் உருவாக்கப்படும் மென்பொருட்களுக்கும், அவைகளின் படைப்பாளிகளுக்கும் பெரும் சவாலாகவும், அதிர்ச்சியாகவும் இருந்தது.

1996 ஆம் வருடத்தின் மத்தியில் இந்த மென் பொருட்கள் உருவாக்கும் சித்தாந்தங்களை சோதித்துப் பார்க்க எனக்கு ஒரு வாய்ப்பு கிடைத்தது, நான் ஒரு திறமூல மென்பொருளை உருவாக்கக் கிடைத்த சந்தர்ப்பம் அது. லினக்ஸ் உருவாக்கிய வழியில் என் மென்பொருளை உருவாக்கினேன். இவ்வழியில் எனக்கு பெரும் வெற்றி கிடைத்தது. திறமூல மென்பொருளில் 'ஃபெச்மெயில்' (fetchmail) என்னும் எனது படைப்பை வெற்றிகரமாக வெளியிட்டேன். மென்பொருள் பொறியியலில் லினக்ஸின் உருவாக்கத்தில் அறிவுறுத்தப்பட்ட வழிமுறையில் எனது படைப்பு அமைந்தது. மென்பொருள் பொறியியல் வளர்ச்சிக் கோட்பாட்டின் இரு துருவங்களான தேவாலய உருவாக்க முறை, சந்தையில் தேவைக்கு ஏற்ப கொள்முதல் செய்யும் முறை மற்றும் இவைகளின் வேறுபாடுகளை இங்கு விளக்க முயற்சிக்கிறேன். லினக்ஸ் உருவாக்கத்தின் கோட்பாடான பிழையைத் தேடி நீக்குதல், தேவாலய முறையில் உருவாக்கப்படும் மென்பொருளின் பிழை நீக்கும் இடர்பாடுகள் மற்றும் எதிர்காலத்தின் மென்பொருள் உருவாக்கும் முறையையும் இங்கு குறிப்பிடுகிறேன். 'ஃபெச்மெயில்' எனும் எனது படைப்பை உருவாக்கிய முறையை இங்கு விளக்குகிறேன். மேலும் திறமூல மென்பொருட்களின் சீரான வளர்ச்சியில் என் கருத்தை முன் வைக்கிறேன். இங்கு குறிப்பிடப்பட்டுள்ளவை எல்லாம் எனக்கு ஃபெச்மெயில் வடிவமைப்பதற்கு முன் தெரிந்தவைகள் அல்ல. லினக்ஸ் உலகின் வளர்ச்சியின் தனித்தன்மையை இங்கு விளக்குகிறேன். மேலும் லினக்ஸ் எப்படி பல மென்பொருட்களைக் கொண்ட ஓர் ஊற்றாக விளங்குகிறது என்பதை நம்மால் உணர முடியும். இக்கருத்துகள் உங்களுக்கு மிகவும் பயனுள்ளதாக இருக்கும் என நம்புகிறேன்.

தடைகளைத் தாண்டி மின்னஞ்சல் செல்ல வழியுண்டா?

1993 ஆம் ஆண்டு வாக்கில், பென்சில்வேனியாவில் உள்ள மேற்கு செஸ்டரில் உள்ள செஸ்டர் கௌண்டி இன்டர் லிங்க் (**Chester County InterLink - CCIL**), சுருக்கமாக சி.சி.ஐ.எல் என்னும் இலவச இணைய வசதி சேவை நிறுவனத்திற்கு நான் பணியாற்றினேன். சி.சி.ஐ.எல்-ஐ உருவாக்கியதில் இணை படைப்பாளியாக, என் பங்களிப்பைத் தந்துள்ளேன். சி.சி.ஐ.எல்-ல் உள்ள பல பயனர்களின் தகவல் பரிமாற்ற சேவையான பி.பி.எஸ் (**Bulletin Board Service - BBS**)-ஐ நான் வடிவமைத்துள்ளேன். நீங்கள் locke.ccil.org என்னும் தளத்திற்கு டெல்நெட்(**telnet**) செய்வதன் மூலம், இந்த சேவையைப் பார்க்கலாம். இன்று சுமார் மூவாயிரம் பயனர்களுக்கு முப்பது தொலைபேசியின் வாயிலாக, பி.பி.எஸ் இணைய சேவையை அளிக்கிறது. இணையத்தின் வாயிலாக மின்னஞ்சல் பெறுவது எனக்குப் பழக்கமான ஒன்று. ஆனால் என் கணினியிலிருந்து (ஸ்நார்க்-என் கணினியின் பெயர்) ஒவ்வொரு முறையும் நான் டெல்நெட் முறையில் என் மின்னஞ்சலைப் பார்க்க வேண்டியிருந்தது . இதற்கு ஒரு மென்பொருள் தேவைப்பட்டது. அந்த மென்பொருள் குறிப்பிட்ட கால இடைவெளியில் அவ்வப்போது சி.சி.ஐ.எல் நெட்வொர்க்கின் **locke.ccil.org** தளத்தைத் தொடர்பு கொண்டு, எனக்குரிய மின்னஞ்சலை எடுத்து, அதை என் கணினியில் (ஸ்நார்க்கில்) சேர்க்க வேண்டும்.

இணையத்தில் எஸ்.எம்.டி.பி ப்ரோடோகால் வழியாக என் மின்னஞ்சலை எடுக்க இயலாது. ஏனென்றால் முழு நேர இணைய வசதி இருந்தால் தான், எஸ்.எம்.டி.பி ப்ரோடோகால் சிறப்பாக செயல்படும். என் கணினிக்கு முழு நேர இணையத் தொடர்பு வசதியோ, நிலையான ஐபி முகவரியோ கிடையாது. எனவே எனக்குத் தேவையானதெல்லாம், அவ்வப்போது சி.சி.ஐ.எல் நெட்வொர்க்கைத் தொடர்பு கொண்டு, என் மின்னஞ்சலை எடுத்து, என் கணினியில் சேர்க்க, ஒரு மென்பொருள். இத்தகைய வசதி கொண்ட மென்பொருள் உண்டு என்று எனக்கு தெரியும். இத்தகைய மென்பொருளின் தகவல் பரிமாற்றத்துக்குப் பயன்படுத்தப்படும் சங்கேத மொழியை போஸ்ட் ஆபிஸ் ப்ரோடோகால் (பாப் கிளைண்ட் - **POP CLIENT**) என்று அழைப்பர் . இன்று பாப் கிளைண்ட் மின்னஞ்சலைக் கையாளும் மென்பொருட்களில் பிரபலமாக உபயோகிக்கப்படுகின்றது. ஆனால் அக்கால கட்டத்தில் என் மின்னஞ்சலைக் கையாளும் மென்பொருளில், பாப் கிளைண்ட் தொழிற்நுட்பம் இல்லை. உடனடியாக எனக்குத் தேவையான மென்பொருளை இணையத்தில் தேட ஆரம்பித்தேன். மூன்று அல்லது நான்கு பாப் கிளைண்ட் தொழிற்நுட்பத்தை கொண்ட மென்பொருளைக் கண்டு அறிந்தேன் . ஆனால் ஒன்று கூட என் தேவையை சரியாகப் பூர்த்தி செய்யவில்லை. நான் இணையத்தில் கண்டறிந்த பாப் கிளைண்டில் சிக்கல் இருந்தது. பிரச்சனை என்னவென்றால், ஜோ என்பவர் **locke.ccil.org** என்னும் நெட்வொர்க்கிலிருந்து எனக்கு அனுப்பிய மின்னஞ்சலுக்கு நான்

பதில் எழுதி அனுப்பினால், அது **locke.ccil.org** நெட்வொர்க்கில் உள்ள ஜோவிற்கு போகாமல், என் கணினியில் ஜோ என்னும் இல்லாத நபருக்கு பதில் மின்னஞ்சலை அனுப்பியது. இதனால் நான் அனுப்பும் மின்னஞ்சலுக்கு, அதன் முகவரியில் பயனர் பெயருடன் மின்னஞ்சலின் நெட்வொர்க் பெயரும் கட்டாயமானது (**joe@ccil.org**). இவ்வண்ணம் நான் எழுதும் ஒவ்வொரு பதில் மின்னஞ்சலுக்கும் அதன் முகவரியில் நெட்வொர்க்கின் பெயரைச் சேர்த்து, கையால் டைப் செய்வது கடினமான வேலையாக இருந்தது. இந்நிலையில் இணையத்தில் எனக்கு கிடைக்கப் பெற்ற மென்பொருட்கள் எல்லாம், என் தேவைக்கு நேரடி தீர்வாக அமையவில்லை. இவைகள் நமக்கு சில படிப்பிணையைத் தந்தன.

1. மென்பொருளின் சிறப்பம்சங்கள், அவைகளின் படைப்பாளிகளின் தனிப்பட்டத் தேவையைப் பொறுத்தே அமைகின்றன.

"தேவையே, கண்டறிதலின் தாய்" என்னும் முதுமொழிக்கேற்ப, படைப்பாளியின் தேவையைப் பொறுத்தே அந்த மென்பொருளின் செயல்தன்மை அமைகிறது. பொதுவாக சம்பளத்திற்காகவோ, பண ஆதாயத்திற்காகவோ செய்யப்படும் மென்பொருட்கள் அதன் படைப்பாளியின் பண ஆதாயத்திற்காக அமைகிறதே தவிர, அவற்றில் படைப்பாளியின் தேவையோ, காதலோ வெளிப்படுவதில்லை. ஆனால் லினக்ஸ் உலகில் எழுதப்படும் மென்பொருட்கள் படைப்பாளியின் தேவையையும், காதலையும் பொறுத்து அமைகின்றன. இதனால் லினக்ஸ் உலகில் செய்யப்படும் மென்பொருட்கள் உயர் தரத்தில் விளங்குகின்றன. என் தேவையைப் பூர்த்தி செய்ய நான் ஒரு பாப் கிளைண்டை புதியதாக உருவாக்க முயற்சி எடுக்க வேண்டுமா, என்று நினைத்த வேளையில், இணையத்தின் வாயிலாக கிடைக்கப் பெற்ற பாப்கிளைண்ட் மென்பொருட்களின் நிரல்களை அலசினேன். எந்த மென்பொருளின் நிரல் என் தேவையைப் பூர்த்தி செய்வதில் ஒத்து இருந்தது என்று ஆராய்ந்தேன்.

2. தன் தேவைக்கேற்ப எப்படி மென்பொருளை வடிவமைப்பது என்று அறிந்தவர், நல்ல நிரலாளர்(Programmer). ஏற்கனவே உள்ள மென்பொருள் நிரல்களை எப்படி மாற்றியமைத்து உபயோகிப்பது என்று அறிந்தவர், சிறந்த நிரலாளர்..

நான் சிறந்த நிரலாளர் என்று இங்கு கூற வரவில்லை. ஆனால் சிறந்த நிரலாளராக என்னை இங்கு புனைந்து கொள்கின்றேன். மென்பொருளின் வெற்றி, தேவையை அது பூர்த்தி செய்வதில் இருக்கிறதே தவிர, அதற்கான நம் முயற்சியில் இல்லை. புதியதாக ஒரு மென்பொருளை உருவாக்கும் முயற்சியைத் தவிர்த்து, தேவையைப் பூர்த்தி செய்வதில் ஒத்திருக்கும் மென்பொருளைத் தேவையைப்

பொறுத்து மாற்றி அமைப்பதில் தான் வெற்றி அமைகிறது. லினஸ் டோர்வால்ட்ஸ் லினக்ஸை அரிச்சுவடியில் இருந்து உருவாக்கவில்லை. மாறாக மினிக்ஸ் என்னும் இயங்குதளத்தை மாற்றி அமைத்து, அவற்றின் சிறப்பம்சங்களை உள்வாங்கி லினக்ஸ் என்னும் இயங்குதளத்தை உருவாக்கினார். லினக்ஸ், மினிக்ஸை உள்வாங்கி பின்பு லினக்ஸ் முழுமையாக மாற்றி வடிவமைக்கப்பட்டது. லினக்ஸ் வடிவமைத்த முறையை முன் மாதிரியாக வைத்து, என்னிடத்தில் இருந்த மென்பொருளை என் தேவையைப் பூர்த்தி செய்வதில் ஒத்திருந்த மென்பொருளை, என் தேவைக்காக மாற்றி அமைக்க முற்பட்டேன். மென்பொருள் திறமூல நிலையினால் அவற்றின் நிரல்களை மாற்றி அமைப்பது சுலபமாக இருந்தது. இதனால் தான் குனு(GNU) உலகம் வெற்றியடைந்து. லினக்ஸின் திறமூல வடிவமைப்பு , பல டெரா பைட் அளவிலான நிரல்களை நமக்கு உண்டான தேவையைப் பூர்த்தி செய்வதில் மற்றும் மாற்றி அமைப்பதில் உற்ற துணையாக இருக்கிறது. இந்த அம்சம் வேறு எங்கும் இருப்பதில்லை.

என்னுடைய இரண்டாவது தேடல் மூலம் இணையத்தில் இருந்து ஒன்பது திறமூல மென் பொருட்களை கண்டு அறிந்தேன். அவைகள் ஃபெச்பாப்(fetchpop), பாப் டார்ட்(PopTart), கெட்-மெயில்(get-mail), ஜி டபள்யூ பாப்(gwpop), பிம்ப்(pimp), பாப்-பேர்ல்(pop-perl), பாப் சி(popc), பாப் மெயில்(popmail) மற்றும் யூ பாப்(upop). ஃபெச்பாப்-ஐ எழுதியவர் சியாங் ஹாங் ஒ. ஃபெச்பாப், என் தேவையைப் பூர்த்தி செய்வதில் நெருங்கி இருந்தது. இதனை என் தேவைக்கேற்ப மாற்றி அமைத்தேன். இதை இதன் ஆசிரியர் சியாங் ஹாங் தன்னுடைய 1.9 வெளியீட்டில் ஏற்றுக்கொண்டார். சில வாரங்கள் கழிந்த பின், ஃபெச்பாப் மென்பொருளில் சில குளறுபடிகளைக் கண்டறிந்தேன். ஃபெச்பாப், பாப்3(pop3) ப்ரோடோகாலை மட்டுமே கையாளக் கூடியதாக இருந்தது. மேலும் ஃபெச்பாப்-ன் நிரல் மிகவும் அனுபவம் இல்லாத நபர் எழுதியது போல் இருந்தது. சியாங் ஹாங் மதிநுட்பம் கொண்டவராகவும், அனுபவம் இல்லாதவராகவும் இருந்தார். அவருடைய படைப்பின் மூலம் நான் இதை உணர்ந்தேன். கார்ல் ஹாரிஸின் படைப்பான பாப்கிளைண்டை நான் கண்டறிந்தேன். கார்லின் படைப்பு மிகவும் நேர்த்தியாக வடிவமைக்கப்பட்டு இருந்தது. ஆனால் ஃபெச்பாப்பில் உள்ள அம்சங்கள் கார்லின் படைப்பான பாப்கிளைண்டில் இல்லை. ஃபெச்பாப்புடன் இருப்பதா? அல்லது அதை துறப்பதா? துறந்தால், ஃபெச்பாப்பில் நான் எழுதிய நிரல்களை இழக்க வேண்டும். ஒரு நல்ல மென்பொருளை உருவாக்க, துறப்பது சரியா? ஃபெச்பாப்பை துறப்பதற்கு சில முக்கிய காரணங்கள் இருந்தது. அது பல ப்ரோடோகால்களை அந்த மென்பொருளால் கையாள முடியும். ஃபெச்பாப்பில் பாப்2, ஆர் பாப் மற்றும் ஏ பாப் போன்ற ப்ரோடோகால்களுக்கு வசதியில்லை. ஐ மேப் (IMAP- Internet Message Access Protocol)-ஐப் பற்றி எனக்கு சமாராக தகவல் தெரிந்து இருந்தது. ஆர்வமிகுதியால் என் மென்பொருளில் இந்த வசதியைச் சேர்க்க நினைத்தேன். நல்ல மென்பொருளை உருவாக்க என் முயற்சியில் சிலவற்றைத் துறப்பதில் எனக்கு உடன்பாடு இருந்தது. இது லினக்ஸை தழுவுவதற்கு முன் நான் கற்றறிந்தது. -

தொடரும்.....

ஓபன் சோர்ஸ் தொடர்பான பணிகளில் ஈடுபடும் நிறுவனங்கள்

ஓபன் சோர்ஸ் தொடர்பான பணிகளில் ஈடுபடும் நிறுவனங்கள் பற்றிய விவரங்களை சேகரித்து வருகிறோம். உங்களுக்கு தெரிந்த தகவல்களை விக்கியில் பகிரவும்.

http://wiki.ilugc.in/index.php?title=Foss_based_solutions_providing_Organizations

நிறுவனம்	சேவை	ஊர்	முகவரி	வலைத்தளம்
IgniteMindZ	Training, IT services and Products in Perl, Python, Advaned Linux Shell Scripting, JBoss with clustering, Ruby on Rails, Android, Embedded Systems, Linux Administration,	Chennai	No 2/8, 2nd floor Saradambal St, T.Nagar, Chennai 600017. Phone: 9092774587	www.ignitemindz.com
CollabNet	Support for Subversion	Chennai	The Lords, Block II,5th Floor #1,2,North Extension Area,Ekkatuthangal, Guindy,Chennai-600032, India,Phone: +91 44 4220-3700,Fax: +91 44 4220-3900,	www.collab.net
LinuXpert Systems	All Popular GNU/Linux Distributions with complete repository, Linux Administration Training, FOSS Lab Setup, LTSP based Thinclient, DSpace based Digital Repository, BigBlueButton based Web Conferencing	Chennai	Mr.S. Baskar, New No.1, Vembuliamman Koil St, Pazhavanthangal, Chennai -- 600 114. Phone: 98841-65649, baskar@linuxpert.in	www.linuxpert.in
HexSource Technologies	We are providing Training and Services in Linux Device Driver, GCC, Python, Arduino, Embedded Systems, Shell Scripting, Zen Cart, PhpBB, SVN, Git, Tcl, Wordpress	Chennai	No 31/7, 1st floor Duraisamy Road, T.Nagar, Chennai 600017. Phone: 08525085655/044 43066505	www.hexsource.com

opennovus	moodle,fedena,italc and all open source lms customization and training	chennai	#3,ramamoorthy avenue , sakthi Nagar,porur ,chennai India, Phone:04443806139	www.opennovus.in
<u>Got Love?</u> - Marry for Love	We do all our development using FoSS & helping village schools with FoSS	Chennai	ECR, Vennangupattu, Kottaikadu P.O, Cheyyur Tk, Kanchi Dt, TN, India, Phone:044-27506020	www.gotlove.in
<u>Blue Light</u>	Non-profit providing Linux/FOSS support and consulting to Auroville community	Auroville, Pondicherry	Saracon, Auroville, TN, India, Phone:0413-2622500	http://bluelightav.org facebook
Magendhiran Consultancy Services	GIS & Remote Sensing	Kanchipuram	05, #rd Street, Gandhi Nagar, Kanchipuram - 631501, India, Phone:+919047134181	www.mcsmapping.com
RedSupport Services Pvt LTD	Installation,Training In Linux For New Users,Implementation Of Ippbx,Call Center,Firewall,Crm And More From Open Source Commerically With Support	Coimbatore, Chennai,Trichy, Salem,Cochin And Pondicerry	N.MAHESH Kumar ,#3.B.K.R Nagar,Opp K.P.N Parcel Service,Sathy Road,Gandhipuram,Coimbatore -641012,Phone-9244432444	WWW.REDSUPPORT.IN
Restart Technologies	Training on Linux Administration, Support for Linux Servers	Ooty	Mr. Raghavendra, 1st floor, UMC market building, ootacamund - 643 001, restarttechnologies@gmail.com, 98434 59995	
Career Gear	Industry Driven Talent Factory; Super affordable cloud computing training and guaranteed placement at emerging Indian cloud based businesses	Trichy	75K/1, B2, 2nd Floor, Shifana Complex, Salai Road, Thillai Nagar, Trichy - 620018, mail AT careergear DOT com, 96299 21512	www.careergear.in
<u>Initcron</u>	Design, Implementation, Automation, Monitoring and Support for Libre/Open Source Technologies including but not limited to Cloud Computing, DevOps, Big Data : Hadoop/Hbase/Pig/Zookeeper/Hive/Oozie, Chef/Puppet Configuration Management, Git/SVN, AWS tools: ec2/s3/AutoScaling/CloudFormation/CloudFront/Route53/RDS/VPC/ElastiCache/Beantalk, Virtualization and Private Cloud: OpenVZ/Xen/Eucalyptus/OpenStack/OpenQRM, DBs:	Chennai/Pune	206, Sai Niketan, 6/12 Lock St. Kottur Gardens, Chennai. Ph: 9962232328	http://www.initcron.com , http://www.initcron.org

	MySQL/PostgreSQL/MongoDB/Cassandra/NoSQL, Monitoring: Nagios/Zenoss/Zabbix/Groundwork, Network: Pfsense/OpenVPN Web: LAMP,RoR,Joomla,Drupal, Wordpress, Web2py,Turbogear s,Pylons,Django,PHPMotion,Magento etc.			
Samgha IT Services Private Limited	Develop Cross Platform Mobile Application and Specialize in Android Platform. Works in Open Source CMS, which includes Wordpress and Joomla. Also Provides advanced Ruby On Rails and J2EE Solutions.	Chennai	1/601, Mugappair west (Near D.R.Furnitures), Chennai-600037, reachus AT samgha DOT com, 9790902787	www.samgha.org
GVSolutions Pvt Ltd	Supporting Opensource products and Linux servers	Chennai	28-B,AnnaiVelankanniNagar,Phase-II,Mugalivakkam,Chennai-600116	
P.V.S.Giridhar & Sai Advocates Legal & Allied Services lawgonindia.com	legal assistance to the open source community in compliance and infringement and related issues	Chennai	No.319(Old No.155), Linghi Chetty Street (III Floor), George Town, Chennai 600 001. Phone: 91-44- 2524 3949/ 4216 3949 Email: giridhar@lawgonindia.com / sai@lawgonindia.com	
Chrisranjana	Php, Mysql Programming. Payment gateways api. Opencart, Moodle, Drupal Customizations. Can provide basic training in Php.	Chennai	32/94 Velachery Road, Guindy. Phone: 65636224	

கணியம் வெளியீட்டு விவரம்

பதிப்புரிமம் © 2012 கணியம்.

கணியத்தில் வெளியிடப்படும் கட்டுரைகள் <http://creativecommons.org/licenses/by-sa/3.0/> பக்கத்தில் உள்ள கிரியேடிவ் காமன்ஸ் நெறிகளையொத்து வழங்கப்படுகின்றன.

இதன்படி,

கணியத்தில் வெளிவரும் கட்டுரைகளை **கணியத்திற்கும் படைத்த எழுத்தாளருக்கும்** உரிய சான்றளித்து, நகலெடுக்க, விநியோகிக்க, பறைசாற்ற, ஏற்றபடி அமைத்துக் கொள்ள, தொழில் நோக்கில் பயன்படுத்த அனுமதி வழங்கப்படுகிறது. கிரியேடிவ் காமன்ஸ் ஒத்த நெறிகளில் வெளியிட வேண்டும்.

ஆசிரியர்: த. ஸ்ரீநிவாஸன் - editor@kaniyam.com

வெளியீட்டாளர்: ம. ஸ்ரீ ராமதாஸ், தொ. பே: +91 94455 75322 – amachu@kaniyam.com

இதழ் ஆக்க மென்பொருட்கள்:

LibreOffice Writer 3.5.3.2 | Ubuntu Linux 12.04 | Gimp 2.8

இதுவரை வெளியான கணியம் இதழ்கள் : <http://www.kaniyam.com/all-releases/>

பதிவிறக்கி பகிருங்கள். **கணியம்** இதழின் மூல ஆவணங்கள் அனைத்தும் <http://dev.kaniyam.com/projects/kaniyam/files> ல் சேமிக்கப்படுகின்றன. அவற்றை பயன்படுத்தி, இதழின் கட்டுரைகளை நீங்கள் மறு பிரசுரம் செய்து கொள்ளலாம்.

நீங்களும் மொழிபெயர்க்கலாமே

http://dev.kaniyam.com/projects/kaniyam/wiki/Translation_Recommendations

மேற்கண்ட சுட்டியில் காணும் கட்டுரைகளை மொழிபெயர்க்கலாமே!

கணியம் பற்றி...

இலக்குகள்

- கட்டற்ற கணிநுட்பத்தின் எளிய விஷயங்கள் தொடங்கி அதிநுட்பமான அம்சங்கள் வரை அறிந்திட விழையும் எவருக்கும் தேவையான தகவல்களை தொடர்ச்சியாகத் தரும் தளமாய் உருபெறுவது.
- உரை, ஒலி, ஒளி என பல்லுடக வகைகளிலும் விவரங்களை தருவது.
- இத்துறையின் நிகழ்வுகளை எடுத்துரைப்பது.
- எவரும் பங்களிக்க ஏதுவாய் யாவருக்குமான நெறியில் விவரங்களை வழங்குவது.
- அச்ச வடிவிலும், புத்தகங்களாகவும், வட்டுக்களாகவும் விவரங்களை வெளியிடுவது.

பங்களிக்க

- விருப்பமுள்ள எவரும் பங்களிக்கலாம்.
- கட்டற்ற கணிநுட்பம் சார்ந்த விஷயமாக இருத்தல் வேண்டும்.
- பகிர்வதை கட்டுப்படுத்தாத வண்ணம் படைப்புகள் இருத்தல் அவசியம். உதாரணத்திற்கு [காபிலெப்ட்](#) & கிரியேடிவ் காமன்ஸ்.
- தாங்கள் பங்களிக்க விரும்பும் ஒரு பகுதியில் வேறொருவர் ஏற்கனவே பங்களித்து வருகிறார் எனின் அவருடன் இணைந்து பணியாற்ற முனையவும்.
- கட்டுரைகள் மொழிபெயர்ப்புகளாகவும், விஷயமறிந்த ஒருவர் சொல்லக் கேட்டு கற்று இயற்றப்பட்டவையாகவும் இருக்கலாம்.
- படைப்புகள் தொடர்களாகவும் இருக்கலாம்.
- தொழில் நுட்பம், கொள்கை விளக்கம், பிரச்சாரம், கதை, கேலிச்சித்திரம், நையாண்டி எனப் பலசுவைகளிலும் இத்துறைக்கு பொருந்தும்படியான ஆக்கங்களாக இருக்கலாம்.
- தங்களுக்கு இயல்பான எந்தவொரு நடையிலும் எழுதலாம்.

- தங்களது படைப்புகளை எளியதொரு உரை ஆவணமாக editor@kaniyam.com முகவரிக்கு அனுப்பிவைக்கவும்.

- தள பராமரிப்பு, ஆதரவளித்தல் உள்ளிட்ட ஏனைய விதங்களிலும் பங்களிக்கலாம்.
- ஐயங்களிருப்பின் editor@kaniyam.com மடலியற்றவும்.

விண்ணப்பங்கள்

- கணித் தொழில்நுட்பத்தை அறிய விழையும் மக்களுக்காக மேற்கொள்ளப்படும் முயற்சியாகும் இது.
- இதில பங்களிக்க தாங்கள் அதிநுட்ப ஆற்றல் வாய்ந்தவராக இருக்க வேண்டும் என்ற கட்டாயமில்லை.
- தங்களுக்கு தெரிந்த விஷயத்தை இயன்ற எளிய முறையில் எடுத்துரைக்க ஆர்வம் இருந்தால் போதும்.
- இதன் வளர்ச்சி நம் ஒவ்வொருவரின் கையிலுமே உள்ளது. குறைகளிலிருப்பின் முறையாக தெரியப்படுத்தி முன்னேற்றத்திற்கு வழி வகுக்கவும்.

துறை சார் - இடம் சார் பொறுப்பாளர்கள் - மாணவ மன்றங்கள்

- தமிழகத்தோறும் கட்டற்ற மென்மம் தொடர்பாக மேற்கொள்ளப்படும் பணிகளை வெளிக்கொணர வேண்டி முதற்கட்டமாக மாவட்டந்தோறும் கணியத்திற்கு தன்னார்வப் பொறுப்பாளர்கள் தேவைப்படுகிறார்கள்.
- கணியம் மாணவ மன்றம் ஒன்றை அமைக்கவும் திட்டமிடப்பட்டுள்ளது.
- கணியத்திற்கு துறை சார்ந்த நுண்ணிய படைப்புக்களை கொண்டு வரக்கூடிய துறைப் பொறுப்பாளர்களும் தேவைப்படுகிறார்கள். தாங்கள் கட்டற்ற கணிநுட்பம் சார்ந்த ஓர் துறையில் நிபுணராக இருந்தால் அத்துறை தொடர்பான செய்திகளை கணியத்திற்கு படைத்துத் தருவதில் பொறுப்பு வகிக்கலாம். உதாரணத்திற்கு லினக்ஸ் கரு உருவாக்கம், டெபியன் இயக்குதளம்.
- தமிழகத்திற்கு முக்கியத்துவம் அளிக்கும் அதே வேலை - கணியத்திற்கு பல்வேறு மாநிலங்களிலிருந்தும் தேசங்களிலிருந்தும் முக்கியச் செய்திகளை தமிழில் வடித்துத் தர ஆர்வமிருப்போரும் வேண்டும்.
- விருப்பமுடையோர் உங்களைப்பற்றிய விவரங்களுடன் nirvagi@kaniyam.com என்ற முகவரியில் தொடர்பு கொள்ளவும்.