Answers

1. Linux is the open source operating system. Linus Torvalds was built Linux operating system with the help of c language. Linux is must important OS for all academic company and offices. Its easy to access all the data base and easy to install all type of software. Linux kernel is the very important and useful for all programmers. When we work Linux we must know all the command in using Linux. Nowadays many OS were available but programmers and hackers must use this Operating system because the Linux commands were very fast access and very secured. We can secured our personal computer safely.

Examples for Linux commands:

1. ls – to show the lists of the file
2. cd – change directory
3. mv – rename or move some fies
4. touch- file creation
5. vi – the editor of Linux all programming files we can write easy
6. Linux kernel:

This is the world wide network access method. The Kernel is based on shell programs or commands. This is a Linux family shell prompt. The Kernel was access the web sites in the way of servers. The same inventor was create who discovered Linux. We can easy to install software and we can easy to find out many data to the help of kernel. Kernel is the best device for all programmers and designers.

3. Difference between 64 bit and 32bit:

A big difference between 32-bit processors and 64-bit processors is the number of calculations per second they can perform, which affects the speed at which they can complete tasks. Another big difference between 32-bit processors and 64-bit processors is the maximum amount of memory (RAM) that is supported. 32-bit computers support a maximum of 3-4GB of memory, whereas a 64-bit computer can support memory amounts over 4 GB.

If your computer has 64bit OS , you had install all software in 64bit type in case 32bit you had all software in 32bit type.

4.Git version in my laptop is git version 2.14.3

When will we store a lot of program in the limit of 100mb. You have a git account you can shared your repo and that man also change some corrections. The correction is changed in our real repo. This is worked like online file edit in gmail file.

5. finish

6. Low level language is directly access to the hardware but high level language is first convert to the low level then access to the hardware. We can understood the high level language but low level language is very hard to understood. Low level language works very fast but high level language performance was too slow to compare low level language.

High level language – Python,C++,Java,PHP

Low level language- Binary numbers

7.OOP:

In object oriented programming was very useful for programmers. In many advantages is here,

* object
* class
* polymorphism
* inheritence
* message passing
* highly protected your computer
* Reuseablity

And many Things etc….

In my knowledge says the object oriented program was the first step to change the next generation for the world because every technologies foundamentaly built the concept of object oriented programming. In class and object we can easily built a function and the must important

is we can run they program many times to input how many objects is there.

The simple c++ program:

#include<iostream.h> //Input output stream //

#include<conio.h> //console input output//

class aswin

{

public:

int a[20],b,c,d,e;

cout<<”The program is to collect your details and last says your character”;

void input();

void output();

};

void aswin::input()

{

cout<<”Enter your name:-”;

cin>>a;

cout<<”Enter your age:-”;

cin>>b;

if(b<25)

{

cout<<”are you enjoy your life y=1,n=0”

cin>>d;

if(d=0)

{

cout<<”ITHUVUM KDANTHU POGUM”;

}

else

{

cout<<”VALZHA VALZHAMUDAN”;

}

}

else

{

cout<<”DONT RUN BACK TO SUCCESS WORK HARD THE SUCCESS IS YOURS ”;

}

}

void aswin::output()

{

if(d=1)

{

for(e=1;e<10;e++)

{

cout<<”YOU ARE NOW ALIVE”;

}

}

else

{

for(e=1;e<10;e++)

{

cout<<”THUNBAM VARUM VEKAIUL SIRINGAL”;

}

}

cout<<”SORRY TO DO THIS I THINK TO MAKE ONE BUT THE OUTPUT IS ANOTHER ONE”:

cout<<”:)”;

}

void main()

{

clrscr();

aswin.ob;

ob.input();

ob.output();

getch();

}

8.PYTHON CODE:

if [ -f /etc/os-release ]; then

# freedesktop.org and systemd

. /etc/os-release

OS=$NAME

VER=$VERSION\_ID

elif type lsb\_release >/dev/null 2>&1; then

# linuxbase.org

OS=$(lsb\_release -si)

VER=$(lsb\_release -sr)

elif [ -f /etc/lsb-release ]; then

# For some versions of Debian/Ubuntu without lsb\_release command

. /etc/lsb-release

OS=$DISTRIB\_ID

VER=$DISTRIB\_RELEASE

elif [ -f /etc/debian\_version ]; then

# Older Debian/Ubuntu/etc.

OS=Debian

VER=$(cat /etc/debian\_version)

elif [ -f /etc/SuSe-release ]; then

# Older SuSE/etc.

...

elif [ -f /etc/redhat-release ]; then

# Older Red Hat, CentOS, etc.

...

else

# Fall back to uname, e.g. "Linux <version>", also works for BSD, etc.

OS=$(uname -s)

VER=$(uname -r)

fi

9.file input:

#!/usr/bin/python

# Open a file

fo = open("foo.txt", "wb")

fo.write( "Python is a great language.\nYeah its great!!\n");

# Close opend file

fo.close()

10.class Person:

TITLES = ('Dr', 'Mr', 'Mrs', 'Ms')

def \_\_init\_\_(self, name, surname):

self.name = name

self.surname = surname

def fullname(self): # instance method

# instance object accessible through self

return "%s %s" % (self.name, self.surname)

@classmethod

def allowed\_titles\_starting\_with(cls, startswith): # class method

# class or instance object accessible through cls

return [t for t in cls.TITLES if t.startswith(startswith)]

@staticmethod

def allowed\_titles\_ending\_with(endswith): # static method

# no parameter for class or instance object

# we have to use Person directly

return [t for t in Person.TITLES if t.endswith(endswith)]

jane = Person("Jane", "Smith")

print(jane.fullname())

print(jane.allowed\_titles\_starting\_with("M"))

print(Person.allowed\_titles\_starting\_with("M"))

print(jane.allowed\_titles\_ending\_with("s"))

print(Person.allowed\_titles\_ending\_with("s"))

I will try to learn python quickly…..here above the python programs I copied……..:)