/\*

**This Program Explain Accessing Non Static Method**

**Both Methods Are Non Static Methods**

\*/

class DemoStatic

{

static DemoStatic obj = new DemoStatic(); // instance object(Global object ) so used in any were.

public static void main(String[] args)

{

System.out.println("Hello World!");

DemoStatic.obj.m1();

}

void m1()

{

/\*

if access non static method m2 in m1() we using this keyword

....

this keyword write only in non static method.

\*/

System.out.println("m1....");

this.m2(); // obj.m2()

}

void m2()

{

// non static method

System.out.println("m2....");

}

}

**/\***

**This Program Explain**

**Accessing Static Method In Non Static Method**

**\*/**

class DemoStatic

{

static DemoStatic obj = new DemoStatic(); // instance object(Global object ) so used in any were

public static void main(String[] args)

{

System.out.println("Hello World!");

DemoStatic.obj.m1();

}

void m1()

{

/\*

if access static method m2 in m1() we using this keyword

....

this keyword write only in non static method.

\*/

System.out.println("m1....");

this.m2(); // obj.m2()

}

static void m2()

{

// static method

System.out.println("m2....");

}

}

**/\***

**This Program Explain**

**Accessing Static Method In Static Method**

**\*/**

// Error ............

// Because this keyword should be in non static method

class DemoStatic

{

static DemoStatic obj = new DemoStatic(); // instance object(Global object ) so used in any were

public static void main(String[] args)

{

System.out.println("Hello World!");

DemoStatic.obj.m1();

}

static void m1()

{

/\*

if access static method m2 in m1() we using this keyword

....

this keyword write only in non static method.

\*/

System.out.println("m1....");

this.m2(); // obj.m2()

}

static void m2()

{

// static method

System.out.println("m2....");

}

}

**Static object program**

class First

{

public static void main(String[] args)

{

//System.out.println("Hello World!");

Second obj = new Second();

Second.fun();

obj.funMore();

System.out.println("");

}

}

class Second

{

static void fun()

{

System.out.println("second class in fun method");

}

void funMore()

{

System.out.println("second class in funMore method");

}

}

**Non Static object program with Error**

class Developer

{

public static void main(String[] args)

{

System.out.println("Hello World");

Lead.approve.previlege();

}

}

class Lead

{

Project approve = new Project(); // static Project approve = new Project();

/\*

onces we remove static tham code generate error

\*/

}

class Project

{

void previlege()

{

System.out.println("Ececution of a Process");

}

}

Output:

/\*

First.java:6: error: non-static variable approve cannot be referenced from a static context

Lead.approve.previlege();

^

1 error

Press any key to continue . . .

\*/

**Non Static object program**

class Developer

{

public static void main(String[] args)

{

System.out.println("Hello World");

//app.previlege();

Lead Dev = new Lead(); // object creation of class Lead

Dev.app.previlege(); // "Dev object is non static type"

/\*

Dev is non static object

app is also non static object

should maintain connection between classes through objects

\*/

}

}

class Lead

{

Project app = new Project(); // object for Project

/\*

In The Program We Create Non Static Object So The Object "App" Is Working In The "Lead Class Only Because It Act As Local To Lead Class". So We Want Accessing Previlege Method In "Main Method" Than Create

Lead class object in Developer class

\*/

}

class Project

{

void previlege()

{

System.out.println("Ececution of a Process");

}

}

output:

Hello World

Ececution of a Process

Press any key to continue . . .