DAY-34

-------

members of the class --> refer the fig

EXAMPLE:1

---------

import java.util.\*;

class Student

{

private String name;

private int age;

private String schoolName;

public void setData()

{

Scanner sc = new Scanner(System.in);

System.out.println("ENTER THE NAME : ");

name = sc.next();

System.out.println("ENTER THE AGE : ");

age = sc.nextInt();

System.out.println("ENTER THE SCHOOL NAME : ");

schoolName = sc.next();

}

public void display()

{

System.out.println("name is :"+name);

System.out.println("age is :"+age);

System.out.println("schoolname is :"+schoolName);

}

}

class Demo

{

public static void main(String[] args)

{

Student s1 = new Student();

s1.setData();

s1.display();

}

}

OUTPUT:

--------

sagar

25

jain

NOTE: Whenever the value is same for multiple objects instead of having multiple copies we can have a single copy.

In the above program schoolname is common for all the students hence we can have a single copy and it can be shared among the objects

such common elements we should make it as static in the program to imporve memory effieceny.

EXAMPLE:2

----------

private String name;

private int age;

static private String schoolName = "jain";

NOTE: when ever java program stats executing always static members will execute first.

the order of execution is:

1. --> static variables

2. --> static blocks

3. --> static methods

HOW TO ACCESS THE STATIC VARIABLES:

-----------------------------------

// Static variable can be accessed using object refernce variable.

class Test

{

static int x=20;

}

class Demo1

{

public static void main(String[] args)

{

Test t1 =new Test();

System.out.println(t1.x);

}

}

// Static variable can be accessed using object refernce variable.

class Test

{

static int x=20;

}

class Demo1

{

public static void main(String[] args)

{

Test t1 =new Test();

System.out.println(t1.x);

}

}

STATIC BLOCKS:

--------------

Static blocks are executed during the class loading.There execution is before the execution of main method.

They always initilize the static variables.

If a block is prefixed with static keyword then it is called as staic block

example:

---------

static{ static

{

} }

EXAMPLE:

----------

// Static blocks

class A

{

int x;

static int y;

{

x=10;

System.out.println("inside the instance block");

System.out.println(x);

}

static

{

y=20;

System.out.println("inside the static block");

System.out.println(y);

}

A()

{

System.out.println("iniside the constructor ");

}

}

class Demo1

{

public static void main(String[] args)

{

A a =new A();

}

}

INSTANCE BLOCK:

---------------

1. It is used to initilize the instance varibale.

2. It will always executes during object creation and before the constructor call.

3. memory will be allocated each time when a new object is created

STATIC BLOCKS:

--------------

1. It is used to initilize the static varibale.

2. It is executed during the class loading and bfore the execution of main method

3. Memory will be allocated only once.

CAN WE EXECUTE A JAVA PROGRAM WITHOUT A MAIN METHOD.

-----------------------------------------------------

--> Till java 1.6 version a java program can be executed without a main method.

From 1.7 version owards main method is manadantory.

STATIC METHODS:

---------------

Static methods are such methods present in the class where a method is prefixed with static keyword.

static methods are class methods which are shared by multiple objects.they are not associated with objetcs.

They can be accessed using the reference variable or class name

EXAMPLE:

--------

class Test

{

void fun1()

{

System.out.println("inside the instance method");

}

static void fun2()

{

System.out.println("inside the static method");

}

}

class Demo2

{

public static void main(String[] args)

{

Test t1 =new Test();

t1.fun1();

t1.fun2();

Test.fun2();

}

}

OUTPUT:

-------

inside the instance method

inside the static method

inside the static method

refer the fig:

EXAMPLE:

---------

class Test

{

static int a,b,c;

int x,y,z;

static

{

a=10;

b=20;

c=30;

}

static void fun1()

{

System.out.println(a);

System.out.println(b);

System.out.println(c);

}

{

x=40;

y=50;

z=60;

a=100;

b=200;

c=300;

}

void fun2()

{

System.out.println(x);

System.out.println(y);

System.out.println(z);

System.out.println(a);

System.out.println(b);

System.out.println(c);

}

}

class Demo3

{

public static void main(String[] args)

{

Test t1 = new Test();

Test.fun1();

Test t2 = new Test();

t2.fun2();

}

}

OUTPUT:

--------

100

200

300

40

50

60

100

200

300

INSATNCE METHOD:

----------------

1. It is also called as non-static method.

2. If we want to call the method where obejct should be created.

3. Can be accessed using the object reference variable

4. It is always deals wuth object level

STATIC METHOD:

---------------

1. It is also called as class method.

2. No need of object creation for method call

3. Can be accessed using the both object ref and class name

4. it is always deals with class level

---------------------------------------------------------------------------------------------------------------------------------------------------------------------