Dt : 31/8/2023

**imp*

2.Anonymous InnerClasses

=>The InnerClasses which are declared without name are known as Anonymous InnerClasses.

=>These Anonymous InnerClasses are categorized into two types:

(a)Anonymous InnerClass as Class Extention

(b)Anonymous InnerClass as Implementation class

(a)Anonymous InnerClass as Class Extention:

=>The process of declaring CClass without name is known as "Anonymous InnerClass as Class Extention"

syntax:

```
class PClass
{
 //PClass_body
}

PClass ob = new PClass()
{
 //CClass_body
```

```
};
```

ProjectName : App_Anonymous1

packages,

p1 : PClass.java

```
package p1;
public class PClass
{
    public void m1(int a)
    {
         System.out.println("****PClass m1(a)****");
         System.out.println("The value a:"+a);
    }
    public void m2(int b)
    {
         System.out.println("****PClass m2(b)****");
         System.out.println("The value b:"+b);
    }
}
```

p2 : DemoAnonymous1.java(MainClass)

```
package p2;
import p1.*;
public class DemoAnonymous1
{
    public static void main(String[] args)
    {
         //CClass without name
        PClass ob = new PClass()
        {
       public void m1(int a)//Overriding method
        {
             System.out.println("***Anonymous m1(a)****");
             System.out.println("The value a:"+a);
```

```
        }
        public void m3(int c)//NonOverriding method
        {
             System.out.println("***Anonymous m3(c)****");
             System.out.println("The value c:"+c);
        }
         };

         ob.m1(12);
         ob.m2(13);
         //ob.m3(14);//Error
      }
}
```

o/p:

****Anonymous m1(a)*****

The value a:12

*****PClass m2(b)*****

The value b:13

ClassFiles:

PClass.class

DemoAnonymous1.class(MainClass)

DemoAnonymous1$1.class

==========

**imp*

(b)Anonymous InnerClass as Implementation class:

=>The process of declaring implementation class without name is known

as "Anonymous InnerClass as implementation class"

syntax:

interface ITest

{

//Interface_body

}

ITest ob = new ITest()

{

//Impl_Class_body

};

ProjectName : App_Anonymous2

packages,

p1 : ITest.java

```
package p1;
public interface ITest
{
    public abstract void m1(int x);
    public default void m2(int y)
    {
         System.out.println("****default m2(y)****");
         System.out.println("The value y:"+y);
    }
```

```
}
```

p2 : DemoAnonymous2.java(MainClass)

```
package p2;
import p1.*;
public class DemoAnonymous2
{
     public static void main(String[] args)
     {
          //Implementation class without name
          ITest ob = new ITest()
          {
          public void m1(int x)//Implemented and
Overriding
          {
               System.out.println("****Anonymous
m1(x)****");
               System.out.println("The value x:"+x);
          }
          public void m3(int z)//NonImplemented and
NonOverriding
          {
               System.out.println("****Anonymous
m3(z)****");
               System.out.println("The value z:"+z);
          }
          };

          ob.m1(11);
          ob.m2(12);
        //ob.m3(14);//Error
     }
}
```

o/p:

```
****Anonymous m1(x)****
```

The value x:11

*****default m2(y)*****

The value y:12

ClassFiles:

ITest.class

DemoAnonymous2.class(MainClass)

DemoAnonymous2$1.class

==========

Note:

=>Anonymous InnerClasses are Local InnerClasses without name.

==========

**imp*

Declaring Anonymous InnerClasses in methods of SubClasses:

ProjectName : App_Anonymous3

packages,

p1 : ITest.java

```
package p1;
public interface ITest
{
    public abstract void m1(int a);
    public default void m2(int b)
    {
         System.out.println("****default m2(b)****");
```

```
        System.out.println("The value b:"+b);
    }
}
```

p1 : Access.java

```
package p1;
public class Access
{
     public static ITest getRef()
     {
     ITest ob = new ITest()
     {
         public void m1(int a)
         {
              System.out.println("****Anonymous
m1(a)****");
              System.out.println("The value a:"+a);
         }
     };
     return ob;
     }
}
```

p2 : DemoAnonymous3.java(MainClass)

```
package p2;
import p1.*;
public class DemoAnonymous3 {
     public static void main(String[] args) {
          ITest ob = Access.getRef();
          ob.m1(11);
          ob.m2(12);
     }
}
```

o/p:

*******Anonymous m1(a)*******

The value a:11

*******default m2(b)*******

The value b:12

Execution Flow of above application:

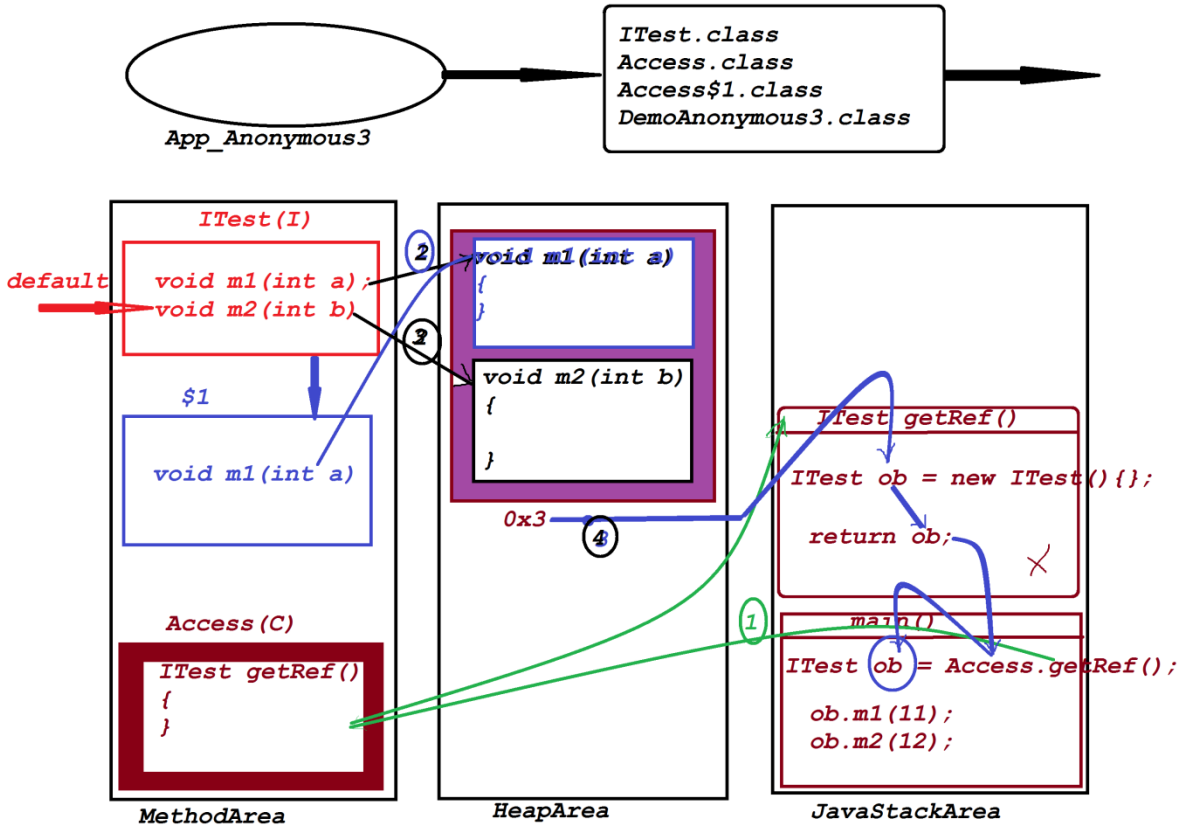
ClassFiles:

ITest.class

Access.class

Access$1.class

DemoAnonymous3.class(MainClass)


App_Anonymous3
ITest.class
Access.class
Access$1.class
DemoAnonymous3.class
ITest(I)
default
void m1(int a);
void m2(int b)
$1
void m1(int a)
Access(C)
ITest getRef()
{
}
MethodArea
2
3
void m1(int a)
{
}
void m2(int b)
{
}
0x3
4
HeapArea
ITest getRef()
ITest ob = new ITest(){};
return ob;
1
main()
ITest ob = Access.getRef();
ob.m1(11);
ob.m2(12);
JavaStackArea