Understanding 'this' Keyword:

We can use 'this' keyword in below mentioned '4' scenerios

It is used to resolve the **ambguity** between Instance Variables & Local Variables.

Tt is used to call present class methods.

It is used to return the Instance of the present class



_	this	this()
	1) It is a Keyword	1) It is a Constructor call
	2) It is used to call present class variables & Methods	2) It is used to call present class Constructors.
	3) We cant use 'this' keyword inside a static area.	3) We need to use this() constructor call ONLY inside a constructor, that too as a FIRST statement. If we are using this() anywhere else we will be getting C.E

Understanding 'this' Keyword:

We can use 'this' keyword in below mentioned '4' scenerios



It is used to resolve the ambguity between Instance Variables & Local Variables.

Ambiguity means confusion

```
<terminated> ClassA [Java Application] C:\Progran
 1 package com.pack1;
                                                                                        meth1() called
                                                                                        Instance Variable: 10
 3 public class ClassA
       int x=10;
 6
 78
       void meth1()
 8
            System.out.println("meth1() called");
            System.out.println("Instance Variable : "+x);
10
11
128
       public static void main(String[] args)
13
14
            ClassA aobj=new ClassA();
15
            aobj.meth1();
16
17 }
```

```
<terminated> ClassA [Java Application] C:\Program F
 1 package com.pack1;
                                                                                        meth1() called
                                                                                        Instance Variable : 111
 3 public class ClassA
 4 {
 5
        int x=10;
 6
 78
       void meth1()
 8
 9
            System.out.println("meth1() called");
10
            int x=111;
11
           System.out.println("Instance Variable : "+x);
12
130
       public static void main(String[] args)
14
15
            ClassA aobj=new ClassA();
16
            aobj.meth1();
17
18 }
                                                                                <terminated> ClassA [Java Application] C:\Program
 1 package com.pack1;
                                                                                  meth1() called
                                                                                  Instance Variable : 10
 3 public class ClassA
 4 {
 5
       int x=10;
 6
       void meth1()
 78
 8
           System.out.println("meth1() called");
 9
10
           int x=111;
         System.out.println("Instance Variable : "+new ClassA().x);
11
130
      public static void main(String[] args)
           ClassA aobj=new ClassA();
15
16
          aobj.meth1();
17
18 }
```

In the above 2 copies of instance variable are there

'this' with refers to already created object in your class

```
🔥 📕 <terminated> ClassA [Java Application] C:\Program F
1 package com.pack1;
                                                                                          meth1() called
                                                                                          Instance Variable : 10
 3 public class ClassA
 4 {
        int x=10;
 6
 70
        void meth1()
 8
            System.out.println("meth1() called");
 9
10
            int x=111;
11
            System.out.println("Instance Variable : "+this.x);
12
13=
        public static void main(String[] args)
14
15
            ClassA aobj=new ClassA();
16
            aobj.meth1();
17
18 }
```

```
— «terminated» ClassA [Java Application] C:\Program

1 package com.pack1;
                                                                                           meth1() called
                                                                                           Instance Variable : 10
 3 public class ClassA
                                                                                            317574433
 4 {
                                                                                           317574433
 5
        int x=10;
 6
 78
        void meth1()
 8
 9
            System.out.println("meth1() called");
10
            int x=111;
11
            System.out.println("Instance Variable : "+this.x);
12
            System.out.println(this.hashCode());
13
148
        public static void main(String[] args)
15
16
            ClassA aobj=new ClassA();
17
            aobj.meth1();
            System.out.println(aobj.hashCode());
18
19
20 }
                                                                                  < = | <terminated> ClassA [Java Application] C:\Program
1 package com.pack1;
                                                                                     meth1() called
                                                                                     Instance Variable : 10
 3 public class ClassA
                                                                                     Static Variable : 222
 4
   {
 5
       int x=10;
 6
       static int y=20;
 88
       void meth1()
 9
10
            System.out.println("meth1() called");
11
            int x=111;
12
           int y=222;
            System.out.println("Instance Variable : "+this.x);
13
14
           System.out.println("Static Variable : "+y);
15
16
       }
       public static void main(String[] args)
17=
18
19
           ClassA aobj=new ClassA();
20
           aobj.meth1();
21
22
       }
23 }
```

```
▲ ■ <terminated> ClassA [Java Application] C:\Program
 1 package com.pack1;
                                                                                        meth1() called
                                                                                        Instance Variable: 10
 3 public class ClassA
                                                                                        Static Variable : 20
4 {
 5
        int x=10;
 6
       static int y=20;
 88
       void meth1()
 9
10
            System.out.println("meth1() called");
11
            int x=111;
12
            int y=222;
13
            System.out.println("Instance Variable : "+this.x);
14
            System.out.println("Static Variable : "+this.y);
15
16
178
       public static void main(String[] args)
18
19
            ClassA aobj=new ClassA();
20
            aobj.meth1();
21
22
       }
23 }

∧ ■ <terminated> ClassA [Java Application] C:\Program
 1 package com.pack1;
                                                                                        meth1() called
                                                                                        Instance Variable: 10
 3 public class ClassA
                                                                                        Static Variable : 20
 4 {
 5
        int x=10;
 6
        static int y=20;
 89
        void meth1()
 9
10
            System.out.println("meth1() called");
            int x=111;
11
12
            int y=222;
            System.out.println("Instance Variable : "+this.x);
13
14
            System.out.println("Static Variable: "+ClassA.y);
15
16
17⊜
        public static void main(String[] args)
18
19
            ClassA aobj=new ClassA();
20
            aobj.meth1();
21
22
23 }
3 public class ClassA
                                                                                meth1() called
 4 {
                                                                                Instance Variable : 10
 5
        int x=10;
                                                                                Static Variable : 20
 6
        static int y=20;
                                                                                Local variable x: 111 y : 222
89
        void meth1()
 9
            System.out.println("meth1() called");
10
11
            int x=111;
12
            int y=222;
            System.out.println("Instance Variable : "+this.x);
13
            System.out.println("Static Variable : "+ClassA.y);
System.out.println("Local variable x: "+x+" y : "+y);
14
15
16
        public static void main(String[] args)
178
18
            ClassA aobj=new ClassA();
19
20
            aobj.meth1();
21
22 }
```



It is used to call present class methods.

```
· 🔛 Training → 😷 src → 🏭 com.pack1 → 🝳 ClassA → 🔺 meth2() : void
                                                                                    <terminated> ClassA [Java Applic
                                                                                    meth1() called
1 package com.pack1;
                                                                                    meth2() called
                                                                                    meth3() called
 3 public class ClassA
4 {
 5e
       void meth1()
                                                                                                  I
 6
            System.out.println("meth1() called");
 8
 98
       void meth2()
10
11
            new ClassA().meth1();
12
            System.out.println("meth2() called");
13
140
       void meth3()
15
16
            new ClassA().meth2();
            System.out.println("meth3() called");
17
18
198
       public static void main(String[] args)
20
21
            new ClassA().meth3();
22
        }
23 }
                                                                                    meth1() called
```

```
1 package com.pack1;
                                                                             meth2() called
                                                                             meth3() called
3 public class ClassA
4 {
 59
       void meth1()
 6
           System.out.println("meth1() called");
 7
 8
 98
       void meth2()
10
       {
11
           meth1();
12
           System.out.println("meth2() called");
13
148
       void meth3()
15
16
           meth2();
17
           System.out.println("meth3() called");
18
19
       public static void main(String[] args)
20
21
           new ClassA().meth3();
22
```

Here we are writing meth1();

But internally the compiler consider this.meth1();

We cannot use 'this' key word in static areas, but by using 'this' key word we can call static method.

```
1 package com.pack1;
 3 public class ClassA
 5€
       void meth1()
 6
 7
           System.out.println("meth1() called");
 8
      Istatic void meth2()
 98
10
           this.meth1();
11
12
           System.out.println("meth2() called");
13
       void meth3()
146
15
16
           meth2();
           System.out.println("meth3() called");
17
18
       public static void main(String[] args)
199
20
21
           new ClassA().meth3();
           //this.meth3(); // C.E because we cant use 'this' keyword inside a static area
22
23
```

```
1 package com.pack1;
 3 public class ClassA
4 {
 5e
       void meth1()
 6
 7
           System.out.println("meth1() called");
 8
 90
       static void meth2()
10
           //this.meth1(); // C.E
11
12
           System.out.println("meth2() called");
13
149
       void meth3()
15
           this.meth2(); warning line because the best way to call satic method is by using its class name
16
17
           System.out.println("meth3() called");
18
198
       public static void main(String[] args)
                                                                Ι
20
21
           new ClassA().meth3();
22
           //this.meth3(); // C.E because we cant use 'this' keyword inside a static area
23
       }
24.3
```

```
3 public class ClassA
 4 {
58
       static void meth1()
6
           System.out.println("meth1() called");
 8
           //this.msg();// C.E
9
100
       void meth2()
11
12
           this.meth1();// We can call a static method with the help of this keyword but it is not recommended
          msg();// Compiler will by default add 'this' keyword
13
14
           System.out.println("meth2() called");
15
16⊜
       void meth3()
17
18
          this.meth2();
19
          System.out.println("meth3() called");
20
218
       void msg()
22
       {
23
          System.out.println("Java is awesome!!!");
24
25≋
       public static void main(String[] args)
26
 27
              new ClassA().meth3();
 28
              //this.meth3(); // C.E because we cant use 'this' keyword inside a static area
 29
 30 }
```

```
4 1
                                                                                             meth1() called
 5e
       static void meth1()
 6
 7
            System.out.println("meth1() called");
 8
            //this.msg();// C.E
 9
109
       void meth2()
11
       {
12
           this.meth1();// We can call a static method with the help of this keyword !
           msg();// Compiler will by default add 'this' keyword
13
14
           System.out.println("meth2() called");
15
169
       void meth3()
17
18
            this.meth2();
19
            System.out.println("meth3() called");
20
218
       void msg()
                                                                                                              I
22
23
            System.out.println("Java is awesome!!!");
24
25⊕
       public static void main(String[] args)
26
27
            new ClassA().meth3();
28
            //this.meth3(); // C.E because we cant use 'this' keyword inside a static :
```

It is used to return the Instance of the present class

Here Access modifier is class it needs to return instance/object of present class

```
meth1() called
1 package com.pack1;
 3 public class ClassA
4 {
 5⊕
       ClassA meth1()
 6
           System.out.println("meth1() called");
 7
 8
           return new ClassA();
 9
100
       void meth2()
11
12
           System.out.println("meth2() called");
13
148
       void meth3()
15
       {
16
           System.out.println("meth3() called");
17
       public static void main(String[] args)
188
19
20
           new ClassA().meth1();
21
22 }
```

```
meth1() called
 1 package com.pack1;
                                                                           meth2() called
                                                                           meth3() called
 3 public class ClassA
 4 {
 58
       ClassA meth1()
 6
           System.out.println("meth1() called");
 8
           return new ClassA();
 9
108
       ClassA meth2()
11
12
           System.out.println("meth2() called");
13
           return new ClassA();
14
158
       void meth3()
16
17
           System.out.println("meth3() called");
18
198
       public static void main(String[] args)
20
21
           new ClassA().meth1().meth2().meth3();
                                                  // Method Chaining
22
23 }
```

```
1 package com.pack1;
3 public class ClassA
      ClassA meth1()
          System.out.println("meth1() called");
          return this;
 9
100
      ClassA meth2()
11
12
          System.out.println("meth2() called");
13
          return this;
14
158
      void meth3()
16
          System.out.println("meth3() called");
17
18
      public static void main(String[] args)
19
20
21
          new ClassA().meth1().meth2().meth3(); // Method Chaining
22
23 }
```

Instance and object both are same

Here new ClassA().meth1() means calling meth1 and but meth1 is returning instance/object

new ClassA().meth1().meth2() will becomes new ClassA().meth2() becomes meth1 is returning object

this

1) It is a Keyword

2) It is used to call present class variables & Methods

3) We cant use 'this' keyword inside a static area.

1) It is a Constructor call

2) It is used to call present class Constructors.

3) We need to use this() constructor call ONLY inside a constructor, that too as a FIRST statement. If we are using this() anywhere else we will be getting C.E

```
Default constructor
1 package com.pack1;
                                                                             Parameterized constructor :
 3 public class ClassA
4 {
       void meth1()
 6
           System.out.println("meh1() called");
 7
 8
       ClassA()
 98
10
           System.out.println("Default constructor");
11
12
           new ClassA(10);
13
       ClassA(int a)
140
15
           System.out.println("Parameterized constructor : "+a);
16
17
189
       public static void main(String[] args)
19
20
           new ClassA();
       }
21
22 }
```

```
1 package com.pack1;
 3 public class ClassA
4 {
 5e
       void meth1()
 6
 7
           System.out.println("meh1() called");
 8
 90
       ClassA()
10
11
           System.out.println("Default constructor");
12
           I/new ClassA(10);
           this(10); _
13
14
       ClassA(int a)
15⊕
16
           System.out.println("Parameterized constructor: "+a);
17
18
19⊕
       public static void main(String[] args)
20
21
           new ClassA();
22
23 }
```

```
Parameterized constructor: 10
1 package com.pack1;
3 public class ClassA
4 {
58
       void meth1()
 6
           System.out.println("meh1() called");
 8
90
      ClassA()
10
11
           this(10);
           System.out.println("Default constructor");
12
13
           //new ClassA(10);
14
15
168
       ClassA(int a)
17
           System.out.println("Parameterized constructor: "+a);
18
19
      public static void main(String[] args)
200
21
22
           new ClassA();
23
```

```
Parameterized constructor : 10
3 public class ClassA
                                                                           meh1() called
4 {
                                                                           Default constructor
58
      void meth1()
6
          System.out.println("meh1() called");
 8
98
      ClassA()
10
11
          this(10);
          this.meth1();
12
          System.out.println("Default constructor");
13
          //new ClassA(10);
15
     ClassA(int a)
168
          System.out.println("Parameterized constructor: "+a);
18
19
200
      public static void main(String[] args)
21
22
          new ClassA();
23
24 }
```

Assignment

```
public String meth4(int a, String s)
        System.out.println("meth4 called");
        a=a+4;
        System.out.println("a value===>"+a);
        System.out.println("s value===>"+s);
        return s+" is Object oriented programming language";
 }
public int meth5(int a)
        System.out.println("meth5() called");
        return 10-a;
public ClassA(String s)
        String result=this.meth4(this.meth1(this.meth2(), this.meth5(5)+'A'-('a')), "Java");
        System.out.println(result);
        System.out.println(s);
}
          public static void main(String[] args)
                    new ClassA();
}
```