Java static nested class

A static class i.e. created inside a class is called static nested class in java. It cannot access non-static data members and methods. It can be accessed by outer class name.

* It can access static data members of outer class including private.
* Static nested class cannot access non-static (instance) data member or method.

Java static nested class example with instance method

1. **class** TestOuter1{
2. **static** **int** data=30;
3. **static** **class** Inner{
4. **void** msg(){System.out.println("data is "+data);}
5. }
6. **public** **static** **void** main(String args[]){
7. TestOuter1.Inner obj=**new** TestOuter1.Inner();
8. obj.msg();
9. }
10. }

[**Test it Now**](http://www.javatpoint.com/opr/test.jsp?filename=TestOuter1)

Output:

data is 30

In this example, you need to create the instance of static nested class because it has instance method msg(). But you don't need to create the object of Outer class because nested class is static and static properties, methods or classes can be accessed without object.

Internal class generated by the compiler

1. **import** java.io.PrintStream;
2. **static** **class** TestOuter1$Inner
3. {
4. TestOuter1$Inner(){}
5. **void** msg(){
6. System.out.println((**new** StringBuilder()).append("data is ")
7. .append(TestOuter1.data).toString());
8. }
9. }

Java static nested class example with static method

If you have the static member inside static nested class, you don't need to create instance of static nested class.

1. **class** TestOuter2{
2. **static** **int** data=30;
3. **static** **class** Inner{
4. **static** **void** msg(){System.out.println("data is "+data);}
5. }
6. **public** **static** **void** main(String args[]){
7. TestOuter2.Inner.msg();//no need to create the instance of static nested class
8. }
9. }

**[Test it Now](http://www.javatpoint.com/opr/test.jsp?filename=TestOuter2" \t "_blank)**

Output:

data is 30