# Java.lang.Class.getClasses() Method

The **java.lang.Class.getClasses()** method returns an array containing Class objects representing all the public classes and interfaces that are members of the class represented by this Class object. This includes public class and interface members inherited from superclasses and public class and interface members declared by the class.

getClasses trả về các class và các interface là các member của class cha, tuy nhiên class phải là public.

**package** ce.tma;

**public** **class** ClassDemo {

**public** **class** InnerClass1 {

**public** InnerClass1() {

System.***out***.println("Inner Class1");

}

}

**public** **class** InnerClass2 {

**public** InnerClass2() {

System.***out***.println("Inner Class2");

}

}

**private** **class** InnerPrivateClass {

**public** InnerPrivateClass() {

System.***out***.println("Inner Private Class");

}

}

**public** **static** **void** main(String[] args) {

Class cls = ClassDemo.**class**;

Class[] classes = cls.getClasses();

**for** (**int** i = 0; i < classes.length; i++) {

System.***out***.println("Class found = " + classes[i].getName());

}

}

}

Result:



# Java.lang.Class.getComponentType() Method

The **java.lang.Class.getComponentType()** method returns the Class representing the component type of an array. If this class does not represent an array class this method returns null.

getComponentType trả về giá trị type của class

**package** ce.tma;

**public** **class** ClassDemo {

**public** **static** **void** main(String[] args) {

String[] str = **new** String[] {"admin"};

//return the Class representing the component type

Class strClass = str.getClass();

Class componentType = strClass.getComponentType();

**if** (componentType != **null**) {

System.***out***.println("ComponentType = " + componentType.getName());

} **else** {

System.***out***.println("ComponentType is null");

}

}

}



# Java.lang.Class.getMethod() Method

The **java.lang.Class.getMethod()** returns a Method object that reflects the specified public member method of the class or interface represented by this Class object. The **name** parameter is a String specifying the simple name of the desired method.

The **parameterTypes** parameter is an array of Class objects that identify the method's formal parameter types, in declared order. If **parameterTypes** is null, it is treated as if it were an empty array. .

getMethod trả về thông tin cho các method trong class.

**package** ce.tma;

**import** java.lang.reflect.\*;

**public** **class** ClassDemo {

**long** l = 78655;

**public** Integer show() {

**return** 1;

}

**public** **void** showLong(Long l) {

**this**.l = l;

}

**public** **static** **void** main(String[] args) {

ClassDemo cls = **new** ClassDemo();

Class c = cls.getClass();

**try** {

Method m = c.getMethod("show", **null**);

System.***out***.println("method = " + m.toString());

} **catch** (NoSuchMethodException e) {

e.printStackTrace();

} **catch** (SecurityException e) {

e.printStackTrace();

}

Class[] cArg = **new** Class[1];

cArg[0] = Long.**class**;

**try** {

Method lMethod = c.getMethod("showLong", cArg);

System.***out***.println("method = " + lMethod.toString());

} **catch** (NoSuchMethodException e) {

e.printStackTrace();

} **catch** (SecurityException e) {

e.printStackTrace();

}

}

}

