

Marker Interface

A marker interface is an interface that doesn't have any methods or constants inside it. It provides run-time type information about objects, so the compiler and JVM have additional information about the object. A marker interface is also called a tagging interface.

eg. Cloneable, Serializable, Remote

instanceof

The **java instanceof operator** is used to test whether the object is an instance of the specified type (class or subclass or interface).

The instanceof in java is also known as type *comparison operator* because it compares the instance with type. It returns either true or false. If we apply the instanceof operator with any variable that has null value, it returns false.

```
import java.util.ArrayList;
import java.util.List;

public class TestInstanceOf {

    public static void main(String[] args) {
        List list = new ArrayList();
        list.add(1);
        list.add("hello");
        list.add(2L);
        list.add(3.00F);
        list.add(3.00D);

        for(Object o : list){

            if(o instanceof String){
                System.out.println(o+" is String");
            }else if(o instanceof Integer){
                System.out.println(o+" is Integer");
            }else if(o instanceof Long){
                System.out.println(o+" is Long");
            }else if(o instanceof Double){
                System.out.println(o+" is Double");
            }else if(o instanceof Float){
                System.out.println(o+" is Float");
            }else{
                System.out.println("something else");
            }
        }
    }
}
```

Cloneable

To create a clone of an object we need to override clone method from Object class

But class does not implement Cloneable interface then CloneNotSupportedException will be thrown at runtime.

So the class which is having clone method should implement Cloneable interface

This information is used at Runtime.

Cloneable is a marker interface it does not have any method.

From java doc:

[Cloneable \(Java Platform SE 8 \) \(oracle.com\)](https://docs.oracle.com/javase/8/docs/api/java/lang/Cloneable.html)

A class implements the Cloneable interface to indicate to the `Object.clone()` method that it is legal for that method to make a field-for-field copy of instances of that class.

Invoking Object's clone method on an instance that does not implement the Cloneable interface results in the exception CloneNotSupportedException being thrown.

By convention, classes that implement this interface should override `Object.clone()` (which is protected) with a public method. See `Object.clone()` for details on overriding this method.

Note that this interface does *not* contain the clone method. Therefore, it is not possible to clone an object merely by virtue of the fact that it implements this interface. Even if the clone method is invoked reflectively, there is no guarantee that it will succeed.

```
interface MyCloneable { }
class MyTest implements MyCloneable { }

//Marker Interfaces - having no method
//eg. Cloneable, Serializable, Remote
//if Account class has override clone method from object class,
//but if Account class does not implements Cloneable interface then CloneNotSupportedException will be thrown
class Account implements Cloneable {
    int accountNumber;

    public Account(int accountNumber) {
        this.accountNumber = accountNumber;
    }

    public int getAccountNumber() {
        return accountNumber;
    }

    public void setAccountNumber(int accountNumber) {
        this.accountNumber = accountNumber;
    }

    @Override
    protected Account clone() throws CloneNotSupportedException {
        return (Account) super.clone();
        //return new Account(this.accountNumber);
    }
}

public class ClonnableTest {

    public static void main(String[] args) throws CloneNotSupportedException {
```

```
Account v = new Account(100);
Account clone = v.clone();//CloneNotSupportedException when without implements Cloneable

if(v instanceof Cloneable){
    System.out.println("yes it is Cloneable, then clone it else throw CloneNotSupportedException");
}

System.out.println(clone.getAccountNumber());
}
}
```

Reference link:

[Cloneable \(Java Platform SE 8 \) \(oracle.com\)](#)

[Cloneable Interface in Java - GeeksforGeeks](#)

[Cloneable in Java - Javatpoint](#)