The *Object* class

- Every base class implicitly extends the Object class
- The Object class provides some useful functionality for all classes
- The Object class is special in that the Java compiler and execution environment know about this.

java.lang

Class Object

java.lang.Object

`public class Object`

Class `Object` is the root of the class hierarchy. Every class has `Object` as a superclass. All objects, including arrays, implement the methods of this class.

Since:

JDK1.0

See Also:

`Class`

Constructor Summary

Constructors

Constructor and Description
`Object()`

Method Summary

Methods

Modifier and Type	Method and Description
`protected Object`	`clone()` Creates and returns a copy of this object.
`boolean`	`equals(Object obj)` Indicates whether some other object is "equal to" this one.
`protected void`	`finalize()` Called by the garbage collector on an object when garbage collection determines that there are no more references to the object.
`Class<?>`	`getClass()` Returns the runtime class of this `Object`.

What do we inherit from Object?

protected *Object* *clone()* Creates and returns a copy of this object.

boolean ***equals****(Object obj)* Indicates whether some other object is "equal to" this one.

protected *void* *finalize()* Called by the garbage collector on an object when garbage collection determines that there are no more references to the object.

Class<? *extends* *Object*> *getClass()* Returns the runtime class of an object.

int *hashCode()* Returns a hash code value for the object.

String *toString()* Returns a string representation of the object.

Overriding equals and hashcode

From http://stackoverflow.com/questions/27581/what-issues-should-be-considered-when-overriding-equals-and-hashcode-in-java

(javadoc) must define an equivalence relation (it must be *reflexive*, *symmetric*, and *transitive*). In addition, it must be *consistent* (if the objects are not modified, then it must keep returning the same value). Furthermore, `o.equals(null)` must always return false.

`hashCode()` (javadoc) must also be *consistent* (if the object is not modified in terms of `equals()`, it must keep returning the same value).

The **relation** between the two methods is:

Whenever `a.equals(b)`, then `a.hashCode()` must be same as `b.hashCode()`. (if a is null then a null pointer exception is thrown)

Use the same object fields to compute equals and hashcode.

What do we inherit from Object?

void notify() Wakes up a single thread that is waiting on this object's monitor.

void notifyAll() Wakes up all threads that are waiting on this object's monitor.

void wait() Causes current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object.

*void **wait**(long timeout)* Causes current thread to wait until either another thread invokes the notify() method or the notifyAll() method for this object, or a specified amount of time has elapsed.

*void **wait**(long timeout, int nanos)* Causes current thread to wait until another thread invokes the notify() method or the notifyAll() method for this object, or some other thread interrupts the current thread, or a certain amount of real time has elapsed.