**The Object class**

***The Java Standard Library*** has a class named Object that is the parent of all standard classes and your custom classes by default. Any class extends this class implicitly, therefore it's a root of inheritance in Java programs. The class belongs to the java.lang package that is imported by default.

The Object class provides some common methods to all subclasses. It has nine instance methods (excluding overloaded methods) which can be divided into four groups:

* *threads synchronization*: wait, notify, notifyAll;
* *object identity*: hashCode, equals;
* *object management*: finalize*,*clone*,*getClass*;*
* *human-readable representation*: toString;

This way of grouping methods isn't supposed to be perfect, but it can help you remember them. Here's a more detailed explanation of the methods:

* The first group of methods (wait, notify, notifyAll*)* are for working in multithreaded applications.
* The method hashCode returns a hash code value for the object.
* The method equalsindicates whether some other object is **"equal to"** this particular one.
* The method finalize is called by the garbage collector (GC) on an object when the GC wants to clean it up. (**Note:**this method has been deprecated as of JDK 9).
* The method clone creates and returns a copy of the object.
* The method getClass returns an instance of Class, which has information about the runtime class.
* The method toString returns a string representation of the object.

Some of the methods listed above are native which means they are implemented in the **"native**" code. It's typically written in C or C++. Native methods are usually used to interface with system calls or libraries written in other programming languages.