# Task-3 Object class & methods in object class with descriptions and syntax

# What is Object class in Java?

Object class in Java

The Object class is the parent class of all the classes in java by default. In other words, it is the topmost class of java.

The Object class is beneficial if you want to refer any object whose type you don't know. Notice that parent class reference variable can refer the child class object, know as upcasting.

Let's take an example, there is getObject() method that returns an object but it can be of any type like Employee,Student etc, we can use Object class reference to refer that object. For example:

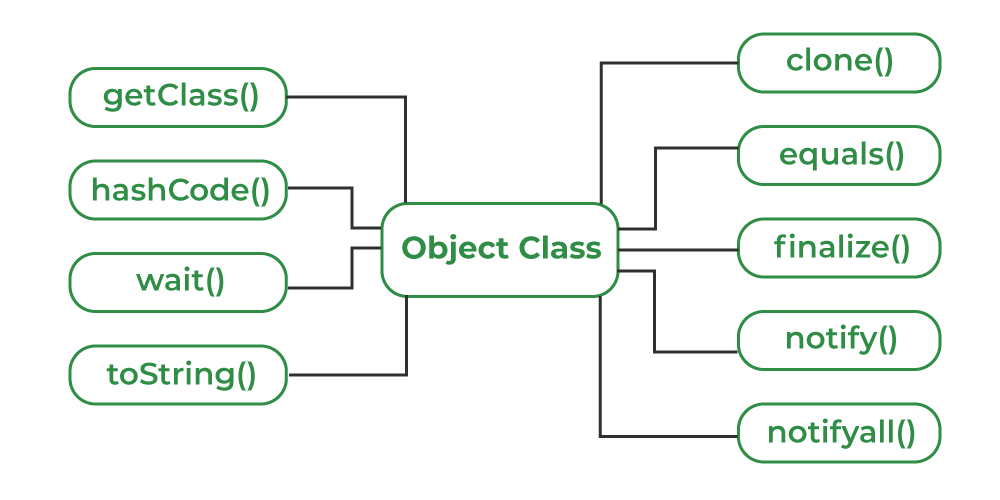
Object obj=getObject();//we don't know what object will be returned from this method

The Object class provides some common behaviors to all the objects such as object can be compared, object can be cloned, object can be notified etc.

## Using Object Class Methods

The Object class provides multiple methods which are as follows:

* tostring() method
* hashCode() method
* equals(Object obj) method
* finalize() method
* getClass() method
* clone() method
* wait() method
* wait(long timeout)method
* wait(long timeout, int nanos) method
* notify() method
* notifyAll() methods



### Methods of Object class

|  |
| --- |
| The Object class provides many methods. They are as follows: |

|  |  |
| --- | --- |
| **Method** | **Description** |
| public final Class getClass() | returns the Class class object of this object. The Class class can further be used to get the metadata of this class. |
| public int hashCode() | returns the hashcode number for this object. |
| public boolean equals(Object obj) | compares the given object to this object. |
| protected Object clone() throws CloneNotSupportedException | creates and returns the exact copy (clone) of this object. |
| public String toString() | returns the string representation of this object. |
| public final void notify() | wakes up single thread, waiting on this object's monitor. |
| public final void notifyAll() | wakes up all the threads, waiting on this object's monitor. |
| public final void wait(long timeout)throws InterruptedException | causes the current thread to wait for the specified milliseconds, until another thread notifies (invokes notify() or notifyAll() method). |
| public final void wait(long timeout,int nanos)throws InterruptedException | causes the current thread to wait for the specified milliseconds and nanoseconds, until another thread notifies (invokes notify() or notifyAll() method). |
| public final void wait()throws InterruptedException | causes the current thread to wait, until another thread notifies (invokes notify() or notifyAll() method). |
| protected void finalize()throws Throwable | is invoked by the garbage collector before object is being garbage collected. |

1. **equals():**

If you want to compare the references of two objects, use equals() method of Object class. This method compares the references of two objects.

It returns true if the object invoking equals() method is equal to the object passed as an argument to the equals() method, otherwise returns false.

The general signature of this method is as:

public boolean equals(Object obj)

1. **getClass():**

This method returns an object containing the name of a class to which an object belongs. In other words, it gives the runtime class of an object. The general signature of this method is as:

Class<? extends Object> getClass()

1. **toString():**

This method gives a string representation of an object. For example, the string representation of an Integer object is an integer number printed as a string. The general signature of this method is:

public String toString()

1. **hashCode():**

This method returns a hash code value of an object. The general signature of this method is as:

public int hashCode()

1. **finalize():**

Garbage collector calls this method when an object is removed from the memory. In other words, the finalize() method is called by garbage collector on an object when there are no more references to an object. The general signature of this method is as:

protected void finalize() throws Throwable

1. **clone()**:

This method creates and returns an exact copy (clone) of an existing object. The general signature of clone() method is as:

protected Object clone() throws CloneNotSupportedException

1. **notify():**

This method wakes up a single thread that is waiting for a particular object. It sends a notification for only one thread at a time.

If multiple threads are waiting for the notification, only one thread receives the notification by notify() method and wakes up. The remaining threads will have to wait for further notification. The general signature of notify() method is:

public final void notify()

This method does not return any value.

1. **notifyAll():**

This method wakes up all threads that are waiting for an object. It sends the notification for all threads at a time. The general signature of notifyAll() method is as:

public void notifyAll()

1. **wait():**

This method causes the current thread to wait until another thread calls notify() or notifyAll() method for this object. The general syntax of wait() method is:

public void wait()

This method returns nothing.

1. **wait(long timeout):**

This method causes the current thread to wait for the specified milliseconds until another thread notifies or calls notify() or notifyAll() method for this object. The general syntax for this method is as:

public void wait(long timeout)

This method does not return anything.

1. **wait(long timeout, int nanos):**

This method causes the current thread to wait for the specified milliseconds and nanoseconds until another thread notifies or calls notify() or notifyAll() method for this object.

The general signature is:

public void wait(long timeout, int nanos)

This overloaded wait() method returns nothing.