**Collection Classes And Arrays Class**

* This class contains various methods for manipulating arrays (such as sorting and searching). This class also contains a static factory that allows arrays to be viewed as lists**.**
* This class consists exclusively of static methods that operate on or return collections.
* It contains polymorphic algorithms that operate on collections, "wrappers", which return a new collection backed by a specified collection,

**What Is Generics(ref oracle)**

* When you take an element out of a Collection, you must cast it to the type of element that is stored in the collection. Besides being inconvenient, this is unsafe. The compiler does not check that your cast is the same as the collection's type, so the cast can fail at run time.
* Generics provides a way for you to communicate the type of a collection to the compiler, so that it can be checked. Once the compiler knows the element type of the collection, the compiler can check that you have used the collection consistently and can insert the correct casts on values being taken out of the collection.
* The net effect of using generics, especially in large programs, is improved readability and robustness.

**Date Class in Java.Util package(ref oracle docs)**

* The class Date represents a specific instant in time, with millisecond precision.
* the class Date had two additional functions. It allowed the interpretation of dates as year, month, day, hour, minute, and second values. It also allowed the formatting and parsing of date strings.

|  |  |
| --- | --- |
| **Constructor Summary** | |
| [**Date**](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/util/Date.html#Date())()           Allocates a Date object and initializes it to represent the current time specified number of milliseconds since the standard base time known as "the epoch", namely January 1, 1970, 00:00:00 GMT. |  |
| [**Date**](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/util/Date.html#Date(long))(long date)           Allocates a Date object and initializes it to represent the specified number of milliseconds since the standard base time known as "the epoch", namely January 1, 1970, 00:00:00 GMT. |  |



|  |  |
| --- | --- |
| **Method Summary** | |
| boolean | [**equals**](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/util/Date.html#equals(java.lang.Object))([Object](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/lang/Object.html) obj)           Compares two dates for equality. |
| long | [**getTime**](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/util/Date.html#getTime())()           Returns the number of milliseconds since January 1, 1970, 00:00:00 GMT represented by this Date object. |
| int | [**hashCode**](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/util/Date.html#hashCode())()           Returns a hash code value for this object. |
| void | [**setTime**](https://docs.oracle.com/javame/config/cldc/ref-impl/cldc1.0/jsr030/java/util/Date.html#setTime(long))(long time)           Sets this Date object to represent a point in time that is time milliseconds after January 1, 1970 00:00:00 GMT. |