java.lang

Class Enum<E extends Enum<E>>

java.lang.Object
  java.lang.Enum<E>

Type Parameters:

`E` - The enum type subclass

All Implemented Interfaces:

Serializable, Comparable<E>

```
public abstract class Enum<E extends Enum<E>>
extends Object
implements Comparable<E>, Serializable
```

This is the common base class of all Java language enumeration types. More information about enums, including descriptions of the implicitly declared methods synthesized by the compiler, can be found in section 8.9 of *The Java™ Language Specification*.

Note that when using an enumeration type as the type of a set or as the type of the keys in a map, specialized and efficient set and map implementations are available.

Since:

1.5

See Also:

`Class.getEnumConstants()`, `EnumSet`, `EnumMap`, Serialized Form

Constructor Summary

Constructors

Modifier	Constructor and Description
protected	**Enum**(**String** name, int ordinal) Sole constructor.

Method Summary

Methods

Modifier and Type	Method and Description
protected **Object**	**clone**() Throws CloneNotSupportedException.
int	**compareTo**(**E** o) Compares this enum with the specified object for order.
boolean	**equals**(**Object** other) Returns true if the specified object is equal to this enum constant.

protected void	**finalize**() enum classes cannot have finalize methods.
Class<E>	**getDeclaringClass**() Returns the Class object corresponding to this enum constant's enum type.
int	**hashCode**() Returns a hash code for this enum constant.
String	**name**() Returns the name of this enum constant, exactly as declared in its enum declaration.
int	**ordinal**() Returns the ordinal of this enumeration constant (its position in its enum declaration, where the initial constant is assigned an ordinal of zero).
String	**toString**() Returns the name of this enum constant, as contained in the declaration.
static <T extends Enum<T>> T	**valueOf**(Class<T> enumType, String name) Returns the enum constant of the specified enum type with the specified name.

Methods inherited from class java.lang.Object

getClass, notify, notifyAll, wait, wait, wait

Constructor Detail

Enum

```
protected Enum(String name,
    int ordinal)
```

Sole constructor. Programmers cannot invoke this constructor. It is for use by code emitted by the compiler in response to enum type declarations.

Parameters:

`name` - - The name of this enum constant, which is the identifier used to declare it.

`ordinal` - - The ordinal of this enumeration constant (its position in the enum declaration, where the initial constant is assigned an ordinal of zero).

Method Detail

name

```
public final String name()
```

Returns the name of this enum constant, exactly as declared in its enum declaration. **Most programmers should use the `toString()` method in preference to this one, as the toString method may return a more user-friendly name.** This method is designed primarily for use in specialized situations where correctness depends on getting the exact name, which will not vary from release to release.

Returns:

the name of this enum constant

ordinal

```
public final int ordinal()
```

Returns the ordinal of this enumeration constant (its position in its enum declaration, where the initial constant is assigned an ordinal of zero). Most programmers will have no use for this method. It is designed for use by sophisticated enum-based data structures, such as `EnumSet` and `EnumMap`.

Returns:

the ordinal of this enumeration constant

toString

```
public String toString()
```

Returns the name of this enum constant, as contained in the declaration. This method may be overridden, though it typically isn't necessary or desirable. An enum type should override this method when a more "programmer-friendly" string form exists.

Overrides:

`toString` in class `Object`

Returns:

the name of this enum constant

equals

```
public final boolean equals(Object other)
```

Returns true if the specified object is equal to this enum constant.

Overrides:

`equals` in class `Object`

Parameters:

`other` - the object to be compared for equality with this object.

Returns:

true if the specified object is equal to this enum constant.

See Also:

`Object.hashCode()`, `HashMap`

hashCode

```
public final int hashCode()
```

Returns a hash code for this enum constant.

Overrides:

`hashCode` in class `Object`

Returns:

a hash code for this enum constant.

See Also:

`Object.equals(java.lang.Object)`, `System.identityHashCode(java.lang.Object)`

clone

```
protected final Object clone()
                     throws CloneNotSupportedException
```

Throws CloneNotSupportedException. This guarantees that enums are never cloned, which is necessary to preserve their "singleton" status.

Overrides:

`clone` in class `Object`

Returns:

(never returns)

Throws:

`CloneNotSupportedException` - if the object's class does not support the `Cloneable` interface. Subclasses that override the `clone` method can also throw this exception to indicate that an instance cannot be cloned.

See Also:

`Cloneable`

compareTo

```
public final int compareTo(E o)
```

Compares this enum with the specified object for order. Returns a negative integer, zero, or a positive integer as this object is less than, equal to, or greater than the specified object. Enum constants are only comparable to other enum constants of the same enum type. The natural order implemented by this method is the order in which the constants are declared.

Specified by:

`compareTo` in interface `Comparable<E extends Enum<E>>`

Parameters:

o - the object to be compared.

Returns:

a negative integer, zero, or a positive integer as this object is less than, equal to, or greater than the specified object.

getDeclaringClass

```
public final Class<E> getDeclaringClass()
```

Returns the Class object corresponding to this enum constant's enum type. Two enum constants e1 and e2 are of the same enum type if and only if e1.getDeclaringClass() == e2.getDeclaringClass(). (The value returned by this method may differ from the one returned by the `Object.getClass()` method for enum constants with constant-specific class bodies.)

Returns:

the Class object corresponding to this enum constant's enum type

valueOf

```
public static <T extends Enum<T>> T valueOf(Class<T> enumType,
                                            String name)
```

Returns the enum constant of the specified enum type with the specified name. The name must match exactly an identifier used to declare an enum constant in this type. (Extraneous whitespace characters are not permitted.)

Note that for a particular enum type `T`, the implicitly declared `public static T valueOf(String)` method on that enum may be used instead of this method to map from a name to the corresponding enum constant. All the constants of an enum type can be obtained by calling the implicit `public static T[] values()` method of that type.

Type Parameters:

`T` - The enum type whose constant is to be returned

Parameters:

`enumType` - the `Class` object of the enum type from which to return a constant

`name` - the name of the constant to return

Returns:

the enum constant of the specified enum type with the specified name

Throws:

`IllegalArgumentException` - if the specified enum type has no constant with the specified name, or the specified class object does not represent an enum type

`NullPointerException` - if `enumType` or `name` is null

Since:

1.5

finalize

```
protected final void finalize()
```

enum classes cannot have finalize methods.

Overrides:

`finalize` in class `Object`