---

# Class java.lang.Float

```
java.lang.Object
   |
   +----java.lang.Number
           |
           +----java.lang.Float
```

---

public final class **Float**
extends Number

The Float class provides an object wrapper for Float data values, and serves as a place for float−oriented operations. A wrapper is useful because most of Java's utility classes require the use of objects. Since floats are not objects in Java, they need to be "wrapped" in a Float instance.

**Version:**
1.29, 10/02/95
**Author:**
Lee Boynton, Arthur van Hoff

---

# Variable Index

o **MAX_VALUE**
The maximum value a float can have.
o **MIN_VALUE**
The minimum value a float can have.
o **NEGATIVE_INFINITY**
Negative infinity.
o **NaN**
Not−a−Number.
o **POSITIVE_INFINITY**
Positive infinity.

# Constructor Index

o **Float**(float)
Constructs a Float wrapper for the specified float value.
o **Float**(double)

Constructs a Float wrapper for the specified double value.

# Method Index


o **doubleValue**()
Returns the double value of this Float.
o **equals**(Object)
Compares this object against some other object.
o **floatToIntBits**(float)
Returns the bit represention of a single−float value
o **floatValue**()
Returns the float value of this Float object.
o **hashCode**()
Returns a hashcode for this Float.
o **intBitsToFloat**(int)
Returns the single−float corresponding to a given bit represention.
o **intValue**()
Returns the integer value of this Float (by casting to an int).
o **isInfinite**(float)
Returns true if the specified number is infinitely large in magnitude.
o **isInfinite**()
Returns true if this Float value is infinitely large in magnitude.
o **isNaN**(float)
Returns true if the specified number is the special Not−a−Number (NaN) value.
o **isNaN**()
Returns true if this Float value is Not−a−Number (NaN).
o **longValue**()
Returns the long value of this Float (by casting to a long).
o **toString**(float)
Returns a String representation for the specified float value.
o **toString**()
Returns a String representation of this Float object.
o **valueOf**(String)
Returns the floating point value represented by the specified String.


# Variables

o **POSITIVE_INFINITY**

```
public final static float POSITIVE_INFINITY
```

Positive infinity.

o **NEGATIVE_INFINITY**

```
public final static float NEGATIVE_INFINITY
```

Negative infinity.

o **NaN**

```
public final static float NaN
```

Not−a−Number. *Note: is not equal to anything, including itself*

o **MAX_VALUE**

```
public final static float MAX_VALUE
```

The maximum value a float can have. The largest maximum value possible is 3.40282346638528860e+38.

o **MIN_VALUE**

```
public final static float MIN_VALUE
```

The minimum value a float can have. The lowest minimum value possible is 1.40129846432481707e−45.

# Constructors

o **Float**

```
public Float(float value)
```

Constructs a Float wrapper for the specified float value.
**Parameters:**
value − the value of the Float

o **Float**

```
public Float(double value)
```

Constructs a Float wrapper for the specified double value.
**Parameters:**
value − the value of the Float

# Methods

o **toString**

```
public static String toString(float f)
```

Returns a String representation for the specified float value.

**Parameters:**
f – the float to be converted

o **valueOf**

```
public static Float valueOf(String s) throws NumberFormatException
```

Returns the floating point value represented by the specified String.
**Parameters:**
s – the String to be parsed
**Throws:** NumberFormatException
If the String does not contain a parsable Float.

o **isNaN**

```
public static boolean isNaN(float v)
```

Returns true if the specified number is the special Not–a–Number (NaN) value.
**Parameters:**
v – the value to be tested

o **isInfinite**

```
public static boolean isInfinite(float v)
```

Returns true if the specified number is infinitely large in magnitude.
**Parameters:**
v – the value to be tested

o **isNaN**

```
public boolean isNaN()
```

Returns true if this Float value is Not–a–Number (NaN).

o **isInfinite**

```
public boolean isInfinite()
```

Returns true if this Float value is infinitely large in magnitude.

o **toString**

```
public String toString()
```

Returns a String representation of this Float object.
**Overrides:**
toString in class Object

o **intValue**

```
public int intValue()
```

Returns the integer value of this Float (by casting to an int).
**Overrides:**
intValue in class Number

o **longValue**

```
public long longValue()
```

Returns the long value of this Float (by casting to a long).
**Overrides:**
longValue in class Number

o **floatValue**

```
public float floatValue()
```

Returns the float value of this Float object.
**Overrides:**
floatValue in class Number

o **doubleValue**

```
public double doubleValue()
```

Returns the double value of this Float.
**Overrides:**
doubleValue in class Number

o **hashCode**

```
public int hashCode()
```

Returns a hashcode for this Float.
**Overrides:**
hashCode in class Object

o **equals**

```
public boolean equals(Object obj)
```

Compares this object against some other object.

*Note: To be useful in hashtables this method considers two Nan floating point values to be equal. This is not according to IEEE specification*
**Parameters:**

obj – the object to compare with

**Returns:**

true if the objects are the same; false otherwise.

**Overrides:**

equals in class Object

o **floatToIntBits**

```
public static int floatToIntBits(float value)
```

Returns the bit represention of a single–float value

o **intBitsToFloat**

```
public static float intBitsToFloat(int bits)
```

Returns the single–float corresponding to a given bit represention.

---