Class sun.tools.debug.RemoteClass

```
java.lang.Object
   |
   +----sun.tools.debug.RemoteValue
           |
           +----sun.tools.debug.RemoteObject
                   |
                   +----sun.tools.debug.RemoteClass
```

public class **RemoteClass**
extends RemoteObject

The RemoteClass class allows access to a class in a remote Java interpreter.

See Also:
RemoteDebugger

Author:
Thomas Ball

Method Index

o **catchExceptions**()
Enter the debugger when an instance of this class is thrown.

o **clearBreakpoint**(int)
Clear a breakpoint at a specific address in a class.

o **clearBreakpointLine**(int)
Clear a breakpoint at a specified line.

o **clearBreakpointMethod**(RemoteField)
Clear a breakpoint at the start of a specified method.

o **description**()
Return a (somewhat verbose) description.

o **getClassLoader**()
Return the classloader for this class.

o **getField**(int)
Return the static field, specified by index.

o **getField**(String)
Return the static field, specified by name.

o **getFieldValue**(int)

Return the value of a static field, specified by its index

- o **getFieldValue**(String)
  Return the value of a static field, specified by name.
- o **getFields**()
  Return all the static fields for this class.
- o **getInstanceField**(int)
  Return the instance field, specified by its index.
- o **getInstanceFields**()
  Return all the instance fields for this class.
- o **getInterfaces**()
  Return the interfaces for this class.
- o **getMethod**(String)
  Return the method, specified by name.
- o **getMethodNames**()
  Return the names of all methods supported by this class.
- o **getMethods**()
  Return the class's methods.
- o **getName**()
  Returns the name of the class.
- o **getStaticFields**()
  Return all the static fields for this class.
- o **getSuperclass**()
  Return the superclass for this class.
- o **ignoreExceptions**()
  Don't enter the debugger when an instance of this class is thrown.
- o **isInterface**()
  Is this RemoteClass an interface?
- o **setBreakpointLine**(int)
  Set a breakpoint at a specified source line number in a class.
- o **setBreakpointMethod**(RemoteField)
  Set a breakpoint at the first line of a class method.
- o **toString**()
  Return a (somewhat verbose) description.
- o **typeName**()
  Returns the name of the class as its type.

Methods

o getName

```
public String getName() throws Exception
```

Returns the name of the class.

o typeName

```
public String typeName() throws Exception
```

Returns the name of the class as its type.

Overrides:

typeName in class RemoteObject

o isInterface

```
public boolean isInterface() throws Exception
```

Is this RemoteClass an interface?

o getSuperclass

```
public RemoteClass getSuperclass() throws Exception
```

Return the superclass for this class.

o getClassLoader

```
public RemoteObject getClassLoader() throws Exception
```

Return the classloader for this class.

o getInterfaces

```
public RemoteClass[] getInterfaces() throws Exception
```

Return the interfaces for this class.

o getFields

```
public RemoteField[] getFields() throws Exception
```

Return all the static fields for this class.

Overrides:

getFields in class RemoteObject

o getStaticFields

```
public RemoteField[] getStaticFields() throws Exception
```

Return all the static fields for this class.

o getInstanceFields

```
public RemoteField[] getInstanceFields() throws Exception
```

Return all the instance fields for this class. Note: because this is a RemoteClass method, only the name and type methods will be valid, not the data.

o **getField**

```
public RemoteField getField(int n) throws Exception
```

Return the static field, specified by index.
Throws: ArrayIndexOutOfBoundsException
when the index is greater than the number of instance variables
Overrides:
getField in class RemoteObject

o **getField**

```
public RemoteField getField(String name) throws Exception
```

Return the static field, specified by name.
Overrides:
getField in class RemoteObject

o **getInstanceField**

```
public RemoteField getInstanceField(int n) throws Exception
```

Return the instance field, specified by its index. Note: because this is a RemoteClass method, only the name and type information is valid, not the data.
Throws: ArrayIndexOutOfBoundsException
when the index is greater than the number of instance variables

o **getFieldValue**

```
public RemoteValue getFieldValue(int n) throws Exception
```

Return the value of a static field, specified by its index
Overrides:
getFieldValue in class RemoteObject

o **getFieldValue**

```
public RemoteValue getFieldValue(String name) throws Exception
```

Return the value of a static field, specified by name.
Overrides:
getFieldValue in class RemoteObject

o **getMethod**

```
public RemoteField getMethod(String name) throws Exception
```

Return the method, specified by name.

o getMethods

```
public RemoteField[] getMethods() throws Exception
```

Return the class's methods.

o getMethodNames

```
public String[] getMethodNames() throws Exception
```

Return the names of all methods supported by this class.

o setBreakpointLine

```
public String setBreakpointLine(int lineno) throws Exception
```

Set a breakpoint at a specified source line number in a class.

Parameters:

lineno – the line number where the breakpoint is set

Returns:

an empty string if successful, otherwise a description of the error.

o setBreakpointMethod

```
public String setBreakpointMethod(RemoteField method) throws Exception
```

Set a breakpoint at the first line of a class method.

Parameters:

method – the method where the breakpoint is set

Returns:

an empty string if successful, otherwise a description of the error.

o clearBreakpoint

```
public String clearBreakpoint(int pc) throws Exception
```

Clear a breakpoint at a specific address in a class.

Parameters:

pc – the address of the breakpoint to be cleared

Returns:

an empty string if successful, otherwise a description of the error.

o clearBreakpointLine

```
public String clearBreakpointLine(int lineno) throws Exception
```

Clear a breakpoint at a specified line.

Parameters:

lineno – the line number of the breakpoint to be cleared

Returns:

an empty string if successful, otherwise a description of the error.

o clearBreakpointMethod

```
public String clearBreakpointMethod(RemoteField method) throws Exception
```

Clear a breakpoint at the start of a specified method.

Parameters:

method – the method of the breakpoint to be cleared

Returns:

an empty string if successful, otherwise a description of the error.

o catchExceptions

```
public void catchExceptions() throws Exception
```

Enter the debugger when an instance of this class is thrown.

Throws: ClassCastException

when this class isn’t an exception class

o ignoreExceptions

```
public void ignoreExceptions() throws Exception
```

Don’t enter the debugger when an instance of this class is thrown.

Throws: ClassCastException

when this class isn’t an exception class

o description

```
public String description()
```

Return a (somewhat verbose) description.

Overrides:

description in class RemoteObject

o toString

```
public String toString()
```

Return a (somewhat verbose) description.

Overrides:

toString in class RemoteObject

All Packages   This Package   Previous   Next