| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Array.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV CLASS   [**NEXT CLASS**](http://docs.google.com/java/sql/BatchUpdateException.html) | [**FRAMES**](http://docs.google.com/index.html?java/sql/Array.html)    [**NO FRAMES**](http://docs.google.com/Array.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

## **java.sql**

Interface Array

**All Known Implementing Classes:** [SerialArray](http://docs.google.com/javax/sql/rowset/serial/SerialArray.html)

public interface **Array**

The mapping in the Java programming language for the SQL type ARRAY. By default, an Array value is a transaction-duration reference to an SQL ARRAY value. By default, an Array object is implemented using an SQL LOCATOR(array) internally, which means that an Array object contains a logical pointer to the data in the SQL ARRAY value rather than containing the ARRAY value's data.

The Array interface provides methods for bringing an SQL ARRAY value's data to the client as either an array or a ResultSet object. If the elements of the SQL ARRAY are a UDT, they may be custom mapped. To create a custom mapping, a programmer must do two things:

* create a class that implements the [SQLData](http://docs.google.com/java/sql/SQLData.html) interface for the UDT to be custom mapped.
* make an entry in a type map that contains
  + the fully-qualified SQL type name of the UDT
  + the Class object for the class implementing SQLData

When a type map with an entry for the base type is supplied to the methods getArray and getResultSet, the mapping it contains will be used to map the elements of the ARRAY value. If no type map is supplied, which would typically be the case, the connection's type map is used by default. If the connection's type map or a type map supplied to a method has no entry for the base type, the elements are mapped according to the standard mapping.

All methods on the Array interface must be fully implemented if the JDBC driver supports the data type.

**Since:** 1.2

| **Method Summary** | |
| --- | --- |
| void | [**free**](http://docs.google.com/java/sql/Array.html#free())()            This method frees the Array object and releases the resources that it holds. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getArray**](http://docs.google.com/java/sql/Array.html#getArray())()            Retrieves the contents of the SQL ARRAY value designated by this Array object in the form of an array in the Java programming language. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getArray**](http://docs.google.com/java/sql/Array.html#getArray(long,%20int))(long index, int count)            Retrieves a slice of the SQL ARRAY value designated by this Array object, beginning with the specified index and containing up to count successive elements of the SQL array. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getArray**](http://docs.google.com/java/sql/Array.html#getArray(long,%20int,%20java.util.Map))(long index, int count, [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)            Retreives a slice of the SQL ARRAY value designated by this Array object, beginning with the specified index and containing up to count successive elements of the SQL array. |
| [Object](http://docs.google.com/java/lang/Object.html) | [**getArray**](http://docs.google.com/java/sql/Array.html#getArray(java.util.Map))([Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)            Retrieves the contents of the SQL ARRAY value designated by this Array object. |
| int | [**getBaseType**](http://docs.google.com/java/sql/Array.html#getBaseType())()            Retrieves the JDBC type of the elements in the array designated by this Array object. |
| [String](http://docs.google.com/java/lang/String.html) | [**getBaseTypeName**](http://docs.google.com/java/sql/Array.html#getBaseTypeName())()            Retrieves the SQL type name of the elements in the array designated by this Array object. |
| [ResultSet](http://docs.google.com/java/sql/ResultSet.html) | [**getResultSet**](http://docs.google.com/java/sql/Array.html#getResultSet())()            Retrieves a result set that contains the elements of the SQL ARRAY value designated by this Array object. |
| [ResultSet](http://docs.google.com/java/sql/ResultSet.html) | [**getResultSet**](http://docs.google.com/java/sql/Array.html#getResultSet(long,%20int))(long index, int count)            Retrieves a result set holding the elements of the subarray that starts at index index and contains up to count successive elements. |
| [ResultSet](http://docs.google.com/java/sql/ResultSet.html) | [**getResultSet**](http://docs.google.com/java/sql/Array.html#getResultSet(long,%20int,%20java.util.Map))(long index, int count, [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)            Retrieves a result set holding the elements of the subarray that starts at index index and contains up to count successive elements. |
| [ResultSet](http://docs.google.com/java/sql/ResultSet.html) | [**getResultSet**](http://docs.google.com/java/sql/Array.html#getResultSet(java.util.Map))([Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)            Retrieves a result set that contains the elements of the SQL ARRAY value designated by this Array object. |

| **Method Detail** |
| --- |

### getBaseTypeName

[String](http://docs.google.com/java/lang/String.html) **getBaseTypeName**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the SQL type name of the elements in the array designated by this Array object. If the elements are a built-in type, it returns the database-specific type name of the elements. If the elements are a user-defined type (UDT), this method returns the fully-qualified SQL type name.

**Returns:**a String that is the database-specific name for a built-in base type; or the fully-qualified SQL type name for a base type that is a UDT **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the type name [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getBaseType

int **getBaseType**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the JDBC type of the elements in the array designated by this Array object.

**Returns:**a constant from the class [Types](http://docs.google.com/java/sql/Types.html) that is the type code for the elements in the array designated by this Array object **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the base type [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getArray

[Object](http://docs.google.com/java/lang/Object.html) **getArray**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the contents of the SQL ARRAY value designated by this Array object in the form of an array in the Java programming language. This version of the method getArray uses the type map associated with the connection for customizations of the type mappings.

**Note:** When getArray is used to materialize a base type that maps to a primitive data type, then it is implementation-defined whether the array returned is an array of that primitive data type or an array of Object.

**Returns:**an array in the Java programming language that contains the ordered elements of the SQL ARRAY value designated by this Array object **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getArray

[Object](http://docs.google.com/java/lang/Object.html) **getArray**([Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves the contents of the SQL ARRAY value designated by this Array object. This method uses the specified map for type map customizations unless the base type of the array does not match a user-defined type in map, in which case it uses the standard mapping. This version of the method getArray uses either the given type map or the standard mapping; it never uses the type map associated with the connection.

**Note:** When getArray is used to materialize a base type that maps to a primitive data type, then it is implementation-defined whether the array returned is an array of that primitive data type or an array of Object.

**Parameters:**map - a java.util.Map object that contains mappings of SQL type names to classes in the Java programming language **Returns:**an array in the Java programming language that contains the ordered elements of the SQL array designated by this object **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getArray

[Object](http://docs.google.com/java/lang/Object.html) **getArray**(long index,  
 int count)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves a slice of the SQL ARRAY value designated by this Array object, beginning with the specified index and containing up to count successive elements of the SQL array. This method uses the type map associated with the connection for customizations of the type mappings.

**Note:** When getArray is used to materialize a base type that maps to a primitive data type, then it is implementation-defined whether the array returned is an array of that primitive data type or an array of Object.

**Parameters:**index - the array index of the first element to retrieve; the first element is at index 1count - the number of successive SQL array elements to retrieve **Returns:**an array containing up to count consecutive elements of the SQL array, beginning with element index **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getArray

[Object](http://docs.google.com/java/lang/Object.html) **getArray**(long index,  
 int count,  
 [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retreives a slice of the SQL ARRAY value designated by this Array object, beginning with the specified index and containing up to count successive elements of the SQL array.

This method uses the specified map for type map customizations unless the base type of the array does not match a user-defined type in map, in which case it uses the standard mapping. This version of the method getArray uses either the given type map or the standard mapping; it never uses the type map associated with the connection.

**Note:** When getArray is used to materialize a base type that maps to a primitive data type, then it is implementation-defined whether the array returned is an array of that primitive data type or an array of Object.

**Parameters:**index - the array index of the first element to retrieve; the first element is at index 1count - the number of successive SQL array elements to retrievemap - a java.util.Map object that contains SQL type names and the classes in the Java programming language to which they are mapped **Returns:**an array containing up to count consecutive elements of the SQL ARRAY value designated by this Array object, beginning with element index **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getResultSet

[ResultSet](http://docs.google.com/java/sql/ResultSet.html) **getResultSet**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves a result set that contains the elements of the SQL ARRAY value designated by this Array object. If appropriate, the elements of the array are mapped using the connection's type map; otherwise, the standard mapping is used.

The result set contains one row for each array element, with two columns in each row. The second column stores the element value; the first column stores the index into the array for that element (with the first array element being at index 1). The rows are in ascending order corresponding to the order of the indices.

**Returns:**a [ResultSet](http://docs.google.com/java/sql/ResultSet.html) object containing one row for each of the elements in the array designated by this Array object, with the rows in ascending order based on the indices. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getResultSet

[ResultSet](http://docs.google.com/java/sql/ResultSet.html) **getResultSet**([Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves a result set that contains the elements of the SQL ARRAY value designated by this Array object. This method uses the specified map for type map customizations unless the base type of the array does not match a user-defined type in map, in which case it uses the standard mapping. This version of the method getResultSet uses either the given type map or the standard mapping; it never uses the type map associated with the connection.

The result set contains one row for each array element, with two columns in each row. The second column stores the element value; the first column stores the index into the array for that element (with the first array element being at index 1). The rows are in ascending order corresponding to the order of the indices.

**Parameters:**map - contains the mapping of SQL user-defined types to classes in the Java programming language **Returns:**a ResultSet object containing one row for each of the elements in the array designated by this Array object, with the rows in ascending order based on the indices. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getResultSet

[ResultSet](http://docs.google.com/java/sql/ResultSet.html) **getResultSet**(long index,  
 int count)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves a result set holding the elements of the subarray that starts at index index and contains up to count successive elements. This method uses the connection's type map to map the elements of the array if the map contains an entry for the base type. Otherwise, the standard mapping is used.

The result set has one row for each element of the SQL array designated by this object, with the first row containing the element at index index. The result set has up to count rows in ascending order based on the indices. Each row has two columns: The second column stores the element value; the first column stores the index into the array for that element.

**Parameters:**index - the array index of the first element to retrieve; the first element is at index 1count - the number of successive SQL array elements to retrieve **Returns:**a ResultSet object containing up to count consecutive elements of the SQL array designated by this Array object, starting at index index. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### getResultSet

[ResultSet](http://docs.google.com/java/sql/ResultSet.html) **getResultSet**(long index,  
 int count,  
 [Map](http://docs.google.com/java/util/Map.html)<[String](http://docs.google.com/java/lang/String.html),[Class](http://docs.google.com/java/lang/Class.html)<?>> map)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Retrieves a result set holding the elements of the subarray that starts at index index and contains up to count successive elements. This method uses the specified map for type map customizations unless the base type of the array does not match a user-defined type in map, in which case it uses the standard mapping. This version of the method getResultSet uses either the given type map or the standard mapping; it never uses the type map associated with the connection.

The result set has one row for each element of the SQL array designated by this object, with the first row containing the element at index index. The result set has up to count rows in ascending order based on the indices. Each row has two columns: The second column stores the element value; the first column stroes the index into the array for that element.

**Parameters:**index - the array index of the first element to retrieve; the first element is at index 1count - the number of successive SQL array elements to retrievemap - the Map object that contains the mapping of SQL type names to classes in the Java(tm) programming language **Returns:**a ResultSet object containing up to count consecutive elements of the SQL array designated by this Array object, starting at index index. **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs while attempting to access the array [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.2

### free

void **free**()  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

This method frees the Array object and releases the resources that it holds. The object is invalid once the free method is called.

After free has been called, any attempt to invoke a method other than free will result in a SQLException being thrown. If free is called multiple times, the subsequent calls to free are treated as a no-op.

**Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if an error occurs releasing the Array's resources [SQLFeatureNotSupportedException](http://docs.google.com/java/sql/SQLFeatureNotSupportedException.html) - if the JDBC driver does not support this method**Since:** 1.6

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Array.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | --- | | | ***Java™ Platform***  ***Standard Ed. 6*** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV CLASS   [**NEXT CLASS**](http://docs.google.com/java/sql/BatchUpdateException.html) | [**FRAMES**](http://docs.google.com/index.html?java/sql/Array.html)    [**NO FRAMES**](http://docs.google.com/Array.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#2et92p0) |

[Submit a bug or feature](http://bugs.sun.com/services/bugreport/index.jsp)

For further API reference and developer documentation, see [Java SE Developer Documentation](http://docs.google.com/webnotes/devdocs-vs-specs.html). That documentation contains more detailed, developer-targeted descriptions, with conceptual overviews, definitions of terms, workarounds, and working code examples.

Copyright 2006 Sun Microsystems, Inc. All rights reserved. Use is subject to [license terms](http://docs.google.com/legal/license.html). Also see the [documentation redistribution policy](http://java.sun.com/docs/redist.html).