| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DataSource.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**](http://docs.google.com/javax/sql/ConnectionPoolDataSource.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/PooledConnection.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/DataSource.html)    [**NO FRAMES**](http://docs.google.com/DataSource.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#3dy6vkm) |

## **javax.sql**

Interface DataSource

**All Superinterfaces:** [CommonDataSource](http://docs.google.com/javax/sql/CommonDataSource.html), [Wrapper](http://docs.google.com/java/sql/Wrapper.html)

public interface **DataSource**extends [CommonDataSource](http://docs.google.com/javax/sql/CommonDataSource.html), [Wrapper](http://docs.google.com/java/sql/Wrapper.html)

A factory for connections to the physical data source that this DataSource object represents. An alternative to the DriverManager facility, a DataSource object is the preferred means of getting a connection. An object that implements the DataSource interface will typically be registered with a naming service based on the JavaTM Naming and Directory (JNDI) API.

The DataSource interface is implemented by a driver vendor. There are three types of implementations:

1. Basic implementation -- produces a standard Connection object
2. Connection pooling implementation -- produces a Connection object that will automatically participate in connection pooling. This implementation works with a middle-tier connection pooling manager.
3. Distributed transaction implementation -- produces a Connection object that may be used for distributed transactions and almost always participates in connection pooling. This implementation works with a middle-tier transaction manager and almost always with a connection pooling manager.

A DataSource object has properties that can be modified when necessary. For example, if the data source is moved to a different server, the property for the server can be changed. The benefit is that because the data source's properties can be changed, any code accessing that data source does not need to be changed.

A driver that is accessed via a DataSource object does not register itself with the DriverManager. Rather, a DataSource object is retrieved though a lookup operation and then used to create a Connection object. With a basic implementation, the connection obtained through a DataSource object is identical to a connection obtained through the DriverManager facility.

**Since:** 1.4

| **Method Summary** | |
| --- | --- |
| [Connection](http://docs.google.com/java/sql/Connection.html) | [**getConnection**](http://docs.google.com/javax/sql/DataSource.html#getConnection())()            Attempts to establish a connection with the data source that this DataSource object represents. |
| [Connection](http://docs.google.com/java/sql/Connection.html) | [**getConnection**](http://docs.google.com/javax/sql/DataSource.html#getConnection(java.lang.String,%20java.lang.String))([String](http://docs.google.com/java/lang/String.html) username, [String](http://docs.google.com/java/lang/String.html) password)            Attempts to establish a connection with the data source that this DataSource object represents. |

| **Methods inherited from interface javax.sql.**[**CommonDataSource**](http://docs.google.com/javax/sql/CommonDataSource.html) |
| --- |
| [getLoginTimeout](http://docs.google.com/javax/sql/CommonDataSource.html#getLoginTimeout()), [getLogWriter](http://docs.google.com/javax/sql/CommonDataSource.html#getLogWriter()), [setLoginTimeout](http://docs.google.com/javax/sql/CommonDataSource.html#setLoginTimeout(int)), [setLogWriter](http://docs.google.com/javax/sql/CommonDataSource.html#setLogWriter(java.io.PrintWriter)) |

| **Methods inherited from interface java.sql.**[**Wrapper**](http://docs.google.com/java/sql/Wrapper.html) |
| --- |
| [isWrapperFor](http://docs.google.com/java/sql/Wrapper.html#isWrapperFor(java.lang.Class)), [unwrap](http://docs.google.com/java/sql/Wrapper.html#unwrap(java.lang.Class)) |

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

### getConnection

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

Attempts to establish a connection with the data source that this DataSource object represents.

**Returns:**a connection to the data source **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs

### getConnection

[Connection](http://docs.google.com/java/sql/Connection.html) **getConnection**([String](http://docs.google.com/java/lang/String.html) username,  
 [String](http://docs.google.com/java/lang/String.html) password)  
 throws [SQLException](http://docs.google.com/java/sql/SQLException.html)

Attempts to establish a connection with the data source that this DataSource object represents.

**Parameters:**username - the database user on whose behalf the connection is being madepassword - the user's password **Returns:**a connection to the data source **Throws:** [SQLException](http://docs.google.com/java/sql/SQLException.html) - if a database access error occurs**Since:** 1.4

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/DataSource.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**](http://docs.google.com/javax/sql/ConnectionPoolDataSource.html)   [**NEXT CLASS**](http://docs.google.com/javax/sql/PooledConnection.html) | [**FRAMES**](http://docs.google.com/index.html?javax/sql/DataSource.html)    [**NO FRAMES**](http://docs.google.com/DataSource.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | CONSTR | [METHOD](#3znysh7) | DETAIL: FIELD | CONSTR | [METHOD](#3dy6vkm) |

[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).