

DataBases	Import	Driver	Connection	Build	Model
<b>Aerospike</b>	import's(1,com\,aerospike\,client	-	[^a-zA-Z ^\\_# 0-9]AerospikeClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*aerospike-client	Key-value, Multi-model
<b>Amazon Redshift</b>	-	com\,amazon\,redshift\,jdbc\,Driver com\,amazon\,redshift\,jdbc\,DataSource	jdbc:redshift	<ls*artifactId's*>ls*redshift-jdbc42-no-awsdk	Relational
<b>Cassandra</b>	import's(1,com\,datastax	[^a-zA-Z ^\\_# 0-9]CassandraConnector[^a-zA-Z ^\\_# 0-9]	-	<ls*artifactId's*>ls*cassandra com\datastax\,oss	Wide column
<b>ClickHouse</b>	-	ru\,yandex\,clickhouse\,ClickHouseDriver	jdbc:clickhouse	<ls*artifactId's*>ls*clickhouse-jdbc <ls*artifactId's*>ls*clickhouse-native-jdbc <ls*artifactId's*>ls*clickhouse4j	Relational, Multi-model
<b>Couchbase</b>	import's(1,com\,couchbase\,client\,java	[^a-zA-Z ^\\_# 0-9]CouchbaseCluster[^a-zA-Z ^\\_# 0-9]	-	(com\,couchbase\,client\,+)\n(ls*<artifactId's*>ls*java-client) couchbase-client	Document, Multi-model
<b>CouchDB</b>	import's(1,org\,ektorp	-	[^a-zA-Z ^\\_# 0-9]CouchDbConnector[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*org\,ektorp	Document, Multi-model
<b>DynamoDB</b>	import's(1,com\,amazonaws\,services\,dynamodbv2	-	[^a-zA-Z ^\\_# 0-9]DynamoDB[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>dynamodb	Multi-model
<b>Ehcache</b>	import's(1,org\,ehcache	-	[^a-zA-Z ^\\_# 0-9]CacheManager[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*ehcache <ls*artifactId's*>ls*terracotta-toolkit	Key-value
<b>ElasticSearch</b>	-	org\,elasticsearch\,client	jdbc:es jdbc:elasticsearch	<ls*artifactId's*>ls*elasticsearch	Search engine, Multi-model
<b>Etcd</b>	import's(1,com\,ibm\,etcd\,client import's(1,io\,etcd\,jetcd\,Client	-	-	<ls*artifactId's*>ls*etcd-java <ls*artifactId's*>ls*jetcd-core	Key-value
<b>FileMaker</b>	-	com\,filemaker\,jdbc\,Driver	jdbc:filemaker	-	Relational
<b>Firebase Realtime Database</b>	import's(1,com\,google\,firebase\,database	[^a-zA-Z ^\\_# 0-9]FirebaseDatabase[^a-zA-Z ^\\_# 0-9]	\,firebaseio\,com \,firebase.database\,app	firebase-database	Document
<b>Firebird</b>	-	org\,firebirdsql\,jdbc\,FBDriver	jdbc:firebirdsql	<ls*artifactId's*>ls*jaybird <ls*artifactId's*>ls*fbclient	Relational
<b>Google BigQuery</b>	import's(1,com\,google\,cloud\,bigquery	com\,simba\,googlebigquery\,jdbc\,Driver com\,simba\,googlebigquery\,jdbc\,DataSource	jdbc:bigquery	<ls*artifactId's*>ls*google-cloud-bigquery	Relational
<b>Google Cloud Datastore</b>	import's(1,com\,google\,cloud\,datastore	-	[^a-zA-Z ^\\_# 0-9]Datastore[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*google-cloud-datastore	Document
<b>H2</b>	import's(1,org\,h2\,tools\,Server import's(1,org\,h2	org\,h2\,Driver	jdbc:h2	<ls*artifactId's*>ls*h2	Relational, Multi-model
<b>Hazelcast</b>	import's(1,com\,hazelcast	-	[^a-zA-Z ^\\_# 0-9]HazelcastClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*shazelcast	Key-value, Multi-model
<b>Hbase</b>	import's(1,org\,apache\,hadoop\,hbase	-	[^a-zA-Z ^\\_# 0-9]HBaseConfiguration[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*hbase	Wide column
<b>Hive</b>	-	org\,apache\,hive\,jdbc\,HiveDriver org\,apache\,hadoop\,hive\,jdbc\,HiveDriver	jdbc:hive	<ls*artifactId's*>ls*hive-jdbc	Relational
<b>IBM DB2</b>	-	com\,ibm\,db2\,jcc\,DB2Driver	jdbc:db2	com\,ibm\,db2	Relational, Multi-model
<b>Ignite - Não Relacional</b>	import's(1,org\,apache\,ignite	-	[^a-zA-Z ^\\_# 0-9]IgniteCache[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*ignite	Multi-model
<b>Ignite - Relacional</b>	-	org\,apache\,ignite\,IgniteJdbcThinDriver	jdbc:ignite:thin	<ls*artifactId's*>ls*ignite	Multi-model
<b>Impala</b>		com\,cloudera\,impala\,jdbc	jdbc:verdict:impala jdbc:impala:	<ls*artifactId's*>ls*ImpalaJDBC	Relacional, Multi-model
<b>InfluxDB</b>	import's(1,com\,influxdb	-	[^a-zA-Z ^\\_# 0-9]InfluxDBClient[^a-zA-Z ^\\_# 0-9] [^a-zA-Z ^\\_# 0-9]FluxClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*influxdb-client-java	Time Series, Multi-model
<b>Informix</b>	-	com\,informix\,jdbc\,IxfDriver	jdbc:informix-sqli	(com\,ibm\,informix\,+)\n(ls*<artifactId's*>ls*jdbc)	Relational, Multi-model
<b>Maria DB</b>	-	org\,mariadb\,jdbc\,Driver	jdbc:mariadb	mariadb-java-client	Relational, Multi-model
<b>MarkLogic - Não Relacional</b>	import's(1,com\,marklogic\,client\,DatabaseClient	-	[^a-zA-Z ^\\_# 0-9]DatabaseClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*marklogic-client-api <ls*artifactId's*>ls*marklogic-xcc	Multi-model
<b>MarkLogic - Relacional</b>	-	com\,marklogic\,Driver	jdbc:marklogic	<ls*artifactId's*>ls*marklogic-client-api <ls*artifactId's*>ls*marklogic-xcc	Multi-model
<b>Memcached</b>	import's(1,com\,whalin\,MemCached import's(1,net\,spy\,memcached import's(1,net\,rubyeye\,xmemcached	-	[^a-zA-Z ^\\_# 0-9]MemCachedClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*Memcached-Java-Client <ls*artifactId's*>ls*spymemcached <ls*artifactId's*>ls*xmemcached	Key-value
<b>Microsoft Azure Cosmos DB</b>	import's(1,com\,azure\,cosmos import's(1,com\,microsoft\,azure\,cosmos	-	[^a-zA-Z ^\\_# 0-9]CosmosClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*azure-cosmos	Multi-model
<b>Microsoft Azure Table Storage</b>	import's(1,com\,azure\,data\,tables	-	[^a-zA-Z ^\\_# 0-9]TableClient[^a-zA-Z ^\\_# 0-9] [^a-zA-Z ^\\_# 0-9]TableServiceClient[^a-zA-Z ^\\_# 0-9]	<ls*artifactId's*>ls*azure-data-tables	Wide column
<b>MongoDB</b>	import's(1,com\,mongodb	-	[^a-zA-Z ^\\_# 0-9]MongoClient[^a-zA-Z ^\\_# 0-9]	org\,mongodb mongodb-java-driver mongodb-driver	Document, Multi-model
<b>MS Access</b>	-	sun\,jdbc\,odbc\,JdbcOdbcDriver	jdbc:ucanaccess	<ls*artifactId's*>ls*ucanaccess	Relational

<b>MS SQL Server</b>	-	com.microsoft.sqlserver.jdbc.SQLServerDriver	jdbc:sqlserver	mssql-jdbc	Relational, Multi-model
<b>MySQL</b>	-	com.mysql.cj.jdbc.Driver com.mysql.jdbc.Driver	jdbc:mysql	mysql-connector-java	Relational, Multi-model
<b>Neo4J</b>	import{s(1,org\neo4j\.	-	[^a-zA-Z ^\\_# 0-9]Neo4jRule[^a-zA-Z ^\\_# 0-9] [^a-zA-Z ^\\_# 0-9]DatabaseManagementService[^a-zA-Z ^\\_# 0-9]	< s*artifactId s*> s*neo4j	Graph
<b>Netezza</b>	-	org.netezza.Driver	jdbc:netezza	< s*artifactId s*> s*netezza	Relational
<b>Oracle</b>	-	oracle.jdbc.driver.OracleDriver oracle.jdbc.OracleDriver	jdbc:oracle	ojdbc com.oracle	Relational, Multi-model
<b>OrientDB</b>	import{s(1,com\tinkerpop\blueprints\impls\orient		[^a-zA-Z ^\\_# 0-9]OrientDB[^a-zA-Z ^\\_# 0-9]	< s*artifactId s*> s*orientdb	Multi-model
<b>PostGIS</b>	import{s(1,org\postgis	org.postgis.PGGeometry	org.postgresql.PGConnection	< s*artifactId s*> s*postgis-jdbc	Spatial DBMS, Multi-model
<b>PostgreSQL</b>	-	org.postgresql.Driver	jdbc:postgresql	< s*artifactId s*> s*postgresql	Relational, Multi-model

<b>Realm</b>	import{s(1,io.realm	-	[^a-zA-Z ^\\_# 0-9]RealmConfiguration[^a-zA-Z ^\\_# 0-9]	< s*artifactId s*> s*realm-android	Document
<b>Redis</b>	redis.clients.jedis.Jedis com.lambdaworks.redis	[^a-zA-Z ^\\_# 0-9]RedisClient[^a-zA-Z ^\\_# 0-9] [^a-zA-Z ^\\_# 0-9]Jedis[^a-zA-Z ^\\_# 0-9]	redis:	< s*artifactId s*> s*lettuce < s*artifactId s*> s*jedis commons\pool2	Key-value, Multi-model
<b>Riak KV</b>	import{s(1,com.basho.riak.client	-	[^a-zA-Z ^\\_# 0-9]RiakCluster[^a-zA-Z ^\\_# 0-9] [^a-zA-Z ^\\_# 0-9]RiakClient[^a-zA-Z ^\\_# 0-9]	< s*artifactId s*> s*riak-client	Key-value
<b>SAP Adaptive Server</b>	-	com.sybase.jdbc4.jdbc.SybDriver	jdbc:sybase	< s*artifactId s*> s*jtds	Relational, Multi-model
<b>SapHana</b>	import{s(1,com.sap.db.jdbc.Driver	com.sap.db.jdbc.Driver	jdbc:sap	< s*artifactId s*> s*ngdbc	Relational, Multi-model
<b>Snowflake</b>	import{s(1,net.snowflake.client.jdbc	net.snowflake.client.jdbc.SnowflakeDriver	jdbc:snowflake	< s*artifactId s*> s*ssnowflake-jdbc	Relational
<b>Solr</b>	import{s(1,org.apache.solr.client.solrj	-	[^a-zA-Z ^\\_# 0-9]SolrClient[^a-zA-Z ^\\_# 0-9]	< s*artifactId s*solr-solrj	Search engine, Multi-model
<b>SQLite</b>	-	SQLite.JDBC.Driver	jdbc:sqlite	sqLitejdbc	Relational
<b>Teradata</b>	-	com.teradata.jdbc.TeraDriver	jdbc:teradata	gt-jdbc-teradata	Relational, Multi-model
<b>Vertica</b>	import{s(1,com.vertica.jdbc	com.vertica.jdbc.Driver	jdbc:vertica	< s*artifactId s*> s*vertica-jdbc	Relational, Multi-model
<b>Virtuoso - Não relacional</b>	import{s(1,virtuosojena.driver import{s(1,com.hp.hpl.jena	-	jdbc:virtuoso	< s*artifactId s*> s*virtjdbc < s*artifactId s*> s*virt_jena	Multi-model
<b>Virtuoso - Relacional</b>	-	virtuoso.jdbc3.Driver	jdbc:virtuoso	< s*artifactId s*> s*virtuoso-jdbc	Multi-model