[**Java数据类型和MySql数据类型对应一览**](http://blog.csdn.net/zoucui/article/details/6125522)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 类型名称 | 显示长度 | 数据库类型 | JAVA类型 | JDBC类型索引(int) | 描述 |
|  |  |  |  |  |  |
| VARCHAR | L+N | VARCHAR | java.lang.String | 12 |  |
| CHAR | N | CHAR | java.lang.String | 1 |  |
| BLOB | L+N | BLOB | java.lang.byte[] | -4 |  |
| TEXT | 65535 | VARCHAR | java.lang.String | -1 |  |
|  |  |  |  |  |  |
| INTEGER | 4 | INTEGER UNSIGNED | java.lang.Long | 4 |  |
| TINYINT | 3 | TINYINT UNSIGNED | java.lang.Integer | -6 |  |
| SMALLINT | 5 | SMALLINT UNSIGNED | java.lang.Integer | 5 |  |
| MEDIUMINT | 8 | MEDIUMINT UNSIGNED | java.lang.Integer | 4 |  |
| BIT | 1 | BIT | java.lang.Boolean | -7 |  |
| BIGINT | 20 | BIGINT UNSIGNED | java.math.BigInteger | -5 |  |
| FLOAT | 4+8 | FLOAT | java.lang.Float | 7 |  |
| DOUBLE | 22 | DOUBLE | java.lang.Double | 8 |  |
| DECIMAL | 11 | DECIMAL | java.math.BigDecimal | 3 |  |
| BOOLEAN | 1 | 同TINYINT |  |  |  |
|  |  |  |  |  |  |
| ID | 11 | PK (INTEGER UNSIGNED) | java.lang.Long | 4 |  |
|  |  |  |  |  |  |
| DATE | 10 | DATE | java.sql.Date | 91 |  |
| TIME | 8 | TIME | java.sql.Time | 92 |  |
| DATETIME | 19 | DATETIME | java.sql.Timestamp | 93 |  |
| TIMESTAMP | 19 | TIMESTAMP | java.sql.Timestamp | 93 |  |
| YEAR | 4 | YEAR | java.sql.Date | 91 |  |

以上就是[**Java**](http://lib.csdn.net/base/javase)数据类型和[**MySQL**](http://lib.csdn.net/base/mysql)数据类型对应表。

对于bolb，一般用于对图片的[**数据库**](http://lib.csdn.net/base/mysql)存储，原理是把图片打成二进制，然后进行的一种存储方式，在java中对应byte［］数组。  
  
对于boolen类型，在mysql数据库中，个人认为用int类型代替较好，对bit操作不是很方便，尤其是在具有web页面开发的项目中，表示0/1，对应java类型的Integer较好。

|  |  |  |
| --- | --- | --- |
| BIT(1) (new in MySQL-5.0) | BIT | java.lang.Boolean |
| BIT( > 1) (new in MySQL-5.0) | BIT | byte[] |
| TINYINT | TINYINT | java.lang.Boolean if the configuration property tinyInt1isBit is set to true (the default) and the storage size is 1, orjava.lang.Integer if not. |
| BOOL, BOOLEAN | TINYINT | See TINYINT, above as these are aliases for TINYINT(1), currently. |
| SMALLINT[(M)] [UNSIGNED] | SMALLINT [UNSIGNED] | java.lang.Integer (regardless if UNSIGNED or not) |
| MEDIUMINT[(M)] [UNSIGNED] | MEDIUMINT [UNSIGNED] | java.lang.Integer, if UNSIGNED java.lang.Long (C/J 3.1 and earlier), or java.lang.Integer for C/J 5.0 and later |
| INT,INTEGER[(M)] [UNSIGNED] | INTEGER [UNSIGNED] | java.lang.Integer, if UNSIGNED java.lang.Long |
| BIGINT[(M)] [UNSIGNED] | BIGINT [UNSIGNED] | java.lang.Long, if UNSIGNED java.math.BigInteger |
| FLOAT[(M,D)] | FLOAT | java.lang.Float |
| DOUBLE[(M,B)] | DOUBLE | java.lang.Double |
| DECIMAL[(M[,D])] | DECIMAL | java.math.BigDecimal |
| DATE | DATE | java.sql.Date |
| DATETIME | DATETIME | java.sql.Timestamp |
| TIMESTAMP[(M)] | TIMESTAMP | java.sql.Timestamp |
| TIME | TIME | java.sql.Time |
| YEAR[(2|4)] | YEAR | If yearIsDateType configuration property is set to false, then the returned object type is java.sql.Short. If set to true (the default) then an object of type java.sql.Date (with the date set to January 1st, at midnight). |
| CHAR(M) | CHAR | java.lang.String (unless the character set for the column isBINARY, then byte[] is returned. |
| VARCHAR(M) [BINARY] | VARCHAR | java.lang.String (unless the character set for the column isBINARY, then byte[] is returned. |
| BINARY(M) | BINARY | byte[] |
| VARBINARY(M) | VARBINARY | byte[] |
| TINYBLOB | TINYBLOB | byte[] |
| TINYTEXT | VARCHAR | java.lang.String |
| BLOB | BLOB | byte[] |
| TEXT | VARCHAR | java.lang.String |
| MEDIUMBLOB | MEDIUMBLOB | byte[] |
| MEDIUMTEXT | VARCHAR | java.lang.String |
| LONGBLOB | LONGBLOB | byte[] |
| LONGTEXT | VARCHAR | java.lang.String |
| ENUM('value1','value2',...) | CHAR | java.lang.String |
| SET('value1','value2',...) | CHAR | java.lang.String |

mysql官方文档：

<http://dev.mysql.com/doc/refman/5.0/en/connector-j-reference-type-conversions.html>