Данные, полученные в результате SQL – запроса возвращаются в виде множества результатов, которые хранятся в сущности под названием Result Set.

Result set - модель ответа запроса к бд, обеспечивает приложению построчный доступ к результатам запросов. При обработке запроса **ResultSet** поддерживает указатель на текущую обрабатываемую строку.

Доступ к данным **ResultSet** обеспечивает посредством набора get-методов, которые организуют доступ к колонкам текущей строки. Метод ResultSet.next используется для перемещения к следующей строке ResultSet, делая ее текущей.

—-------------------------------------------------------------------------------------------------------------------------

**ResultSet** содержит так называемый курсор, который позиционируется на текущей строке данных. При вызове метода **next**, курсор перемещается на следующую строку. При открытии набора данных **ResultSet** курсор расположен **перед первой строкой**, и первый вызов next передвигает его на первую строку. Если не вызвать этот метод то будет ошибка, resultSet вернёт null. **ResultSet** хранит курсор до самого закрытия или пока не закроется родительский объект Statement.

—-—----------—------------------------------**ОСОБЕННОСТИ РАБОТЫ**—----------------------------------------------

**-**Если в цикле перебирать resultSet и одновременно менять что-то в бд то выскочит ошибка**!**

—--------------------------------------------------**методы get** ---------------------------------------------------

Методы ResultSet.getXXX предоставляют доступ к значениям в колонках в текущей строке. В пределах одной строки значения могут быть считаны в любом порядке. Для обеспечения бо́льшей совместимости рекомендуется считывать их подряд слева направо и делать это только один раз. Для указания колонки можно использовать либо ее имя, либо ее номер. Например, если вторая колонка объекта ResultSet rs называется "title" и хранит строковое значение, то извлечь его можно одним из двух способов:

String s = rs.getString("title");

String s = rs.getString(2);

При обращении к колонке по номеру следует помнить, что колонки нумеруются слева направо, начиная с 1, а имена колонок в вызове методов getXXX нечувствительны к регистру букв.

Наименования колонок совпадает с соответствующими наименованиями колонок в запросе. Если же в выражении select не указываются имена колонок (например "select \* from users"), то необходимо либо использовать номера колонок, либо "подключать" метаданные. Информацию о колонках в ResultSet можно получить с помощью вызова ResultSet.getMetaData. Возвращаемый объект ResultSetMetaData содержит информацию о количестве, типах и свойствах колонок объекта ResultSet.

В некоторых случаях имена двух колонок могут совпадать. Тогда при использовании имен колонок в методах getXXX возвращается значение первой подходящей колонки. Таким образом, чтобы считать значение других колонок с таким же именем, надо использовать индексы колонок. Кроме того, использование индексов немного эффективнее.

Получаем в resultSet столбец состоящий из строк. **Столбцы в таблицах начинаются с 0,** искомый столбец идёт вторым, поэтому ставим номер столбца 1 (он соответствует 2). И на каждой итерации **resultSet.next()** передвигает курсор на одну строку ниже. **ННН да, я тут потестил, походу начинаются всё-же индексы с 1, но если данный столбец Auto-increment то он не идёт в счёт!!!**

resultSet=statment.executeQuery("SELECT link FROM links\_for\_video\_on\_youtube;");

while (resultSet.next()){

lisOfUrls.add(resultSet.getString(1));

}

-------—----------------------**Типы данных и их преобразование**--------------------------------

Различные методы чтения записей типа getXXX конвертируют низкоуровневые данные в типы данных Java. Например, если в таблице БД тип данных VARCHAR, то при использовании метода getString, драйвер JDBC конвертирует VARCHAR в объект String. Т.е. возвращаемым из метода getString значением будет объект String.

Следующая таблица показывает, какие типы данных различные методы getXXX могут считывать и какие JDBC-типы (SQL-типы) рекомендуются для этих методов.

Условные обозначения :

"x" означает, что метод getXXX может быть использован,

"X" означает, что данный метод рекомендуется использовать для этого типа данных.

Например, для типа данных LONGVARCHAR значение можно извлечь любым из методов getXXX кроме getBytes и getBinaryStream, но рекомендуется использовать методы getAsciiStream и getUnicodeStream.

Метод getObject возвращает значение как Object и может быть использован в тех случаях, когда соответствующий низкоуровневый тип данных является специфичным для данной СУБД, или когда приложению необходимо принять любой тип данных.

Таблица соответствия методов ResultSet.getXXX при чтении значений различных типам данных SQL.

|  | **T**  **I**  **N**  **Y**  **I**  **N**  **T** | **S**  **M**  **A**  **L**  **L**  **I**  **N**  **T** | **I**  **N**  **T**  **E**  **G**  **E**  **R** | **B**  **I**  **G**  **I**  **N**  **T** | **R**  **E**  **A**  **L** | **F**  **L**  **O**  **A**  **T** | **D**  **O**  **U**  **B**  **L**  **E** | **D**  **E**  **C**  **I**  **M**  **A**  **L** | **N**  **U**  **M**  **E**  **R**  **I**  **C** | **B**  **I**  **T** | **C**  **H**  **A**  **R** | **V**  **A**  **R**  **C**  **H**  **A**  **R** | **L**  **O**  **N**  **G**  **V**  **A**  **R**  **C**  **H**  **A**  **R** | **B**  **I**  **N**  **A**  **R**  **Y** | **V**  **A**  **R**  **B**  **I**  **N**  **A**  **R**  **Y** | **L**  **O**  **N**  **G**  **V**  **A**  **R**  **B**  **I**  **N**  **A**  **R**  **Y** | **D**  **A**  **T**  **E** | **T**  **I**  **M**  **E** | **T**  **I**  **M**  **E**  **S**  **T**  **A**  **M**  **P** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| getByte | **X** | x | x | x | x | x | x | x | x | x | x | x | x |  |  |  |  |  |  |
| getShort | x | **X** | x | x | x | x | x | x | x | x | x | x | x |  |  |  |  |  |  |
| getInt | x | x | **X** | x | x | x | x | x | x | x | x | x | x |  |  |  |  |  |  |
| getLong | x | x | x | **X** | x | x | x | x | x | x | x | x | x |  |  |  |  |  |  |
| getFloat | x | x | x | x | **X** | x | x | x | x | x | x | x | x |  |  |  |  |  |  |
| getDouble | x | x | x | x | x | **X** | **X** | x | x | x | x | x | x |  |  |  |  |  |  |
| getBigDecimal | x | x | x | x | x | x | x | **X** | **X** | x | x | x | x |  |  |  |  |  |  |
| getBoolean | x | x | x | x | x | x | x | x | x | **X** | x | x | x |  |  |  |  |  |  |
| getString | x | x | x | x | x | x | x | x | x | x | **X** | **X** | x | x | x | x | x | x | x |
| getBytes |  |  |  |  |  |  |  |  |  |  |  |  |  | **X** | **X** | x |  |  |  |
| getDate |  |  |  |  |  |  |  |  |  |  | x | x | x |  |  |  | **X** |  | x |
| getTime |  |  |  |  |  |  |  |  |  |  | x | x | x |  |  |  |  | **X** | x |
| getTimestamp |  |  |  |  |  |  |  |  |  |  | x | x | x |  |  |  | x |  | **X** |
| getAsciiStream |  |  |  |  |  |  |  |  |  |  | x | x | **X** | x | x | x |  |  |  |
| getUnicodeStream |  |  |  |  |  |  |  |  |  |  | x | x | **X** | x | x | x |  |  |  |
| getBinaryStream |  |  |  |  |  |  |  |  |  |  |  |  |  | x | x | **X** |  |  |  |
| getObject | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |