Difference Between Execute(), query() and Update() Methods in Java

Last Updated : 05 Oct, 2021

Before getting started, let us have prior knowledge of parameters that makes use of the following three queries parameters, which are as follows:

- **boolean execute(String SQL):** Returns a boolean value of true if a ResultSet object can be retrieved; otherwise, it returns false. Use this method to execute SQL DDL statements or when you need to use truly dynamic SQL.
- **int executeUpdate(String SQL):** Returns the number of rows affected by the execution of the SQL statement. Use this method to execute SQL statements, for which you expect to get a number of rows affected – for example, an INSERT, UPDATE, or DELETE statement.
- **ResultSet executeQuery(String SQL):** Returns a ResultSet object. Use this method when you expect to get a result set, as you would with a SELECT statement.

They are discussed in below listed manner as follows:

1. execute()
2. execute Query()
3. execute Update()

Method 1: execute()

- **Description:** The method used for all types of SQL statements, and that is, returns a Boolean value of TRUE or FALSE.

- **Return type:** This method return a Boolean value. TRUE indicates that query returned a Result Set object and FALSE indicate returned an int value or returned nothing.
- **Usage:** This method is use to execute Both select and non select queries.
- **Example**: All SQL statements.

Illustration:

Java

```
// Java Program to Illustrate usage of execute() Method

// Loading the driver using forName() method
Class.forName(driver);

// Registering the driver using Drivermanager.getConnection() method
Connection conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/
                    "root", "1234");

// Get database connection
stmt = conn.createStatement();

// Use Connection to create a Statement object

// Execute SQL and return boolean value to
// indicate whether it contains ResultSet
boolean hasResultSet = stmt.execute(sql);

// Condition holds true till there is a single element
if (hasResultSet)
{

    // If there is a ResultSet result set after execution
    rs = stmt.getResultSet();
    // Get the result set

    ResultSetMetaData rsmd = rs.getMetaData();

    // ResultSetMetaData is a metadata interface for analyzing result sets
    int  columnCount = rsmd.getColumnCount();

    // Getting the output ResultSet object
    // with help of object of ResultSet
    while (rs.next ())
    {
        for (int i = 0 ; i < columnCount ; i++ )
        {
            System.out.print(rs.getString(i + 1) + "/t");
        }
        System.out.print("/n");
    }
}
else
{

    System.out.println ("The records affected by this SQL statement are"
    + stmt.getUpdateCount () + "Article");
}
```

Method 2: execute Query()

- **Description:** Now this method execute statements that returns a result set by fetching some data from the database.
- **Usage:** This method is use to execute select query.
- **Return type:** This method returns a Result Set object which contains the result returned by query.
- One of it’s example that is widely common: ‘SELECT’

Illustration:

Java

```
// Java Program to Illustrate execute Query() Method

// Again first step is to load and register drivers
Class.forName("com.mysql.jdbc.Driver");

Connection  conn = null;
conn = DriverManager.getConnection("jdbc:mysql://localhost:3306/test",
            "root","root");
// Using DriverManager to get database connection

Statement  stmt = conn.createStatement();
// Use Connection to create a Statement object

// Creating an object of ResultSet class
ResultSet rs =stmt.executeQuery("select * from teacher");
```

```
// Execute the query statement and save the result
// Iterating over elements in above object
while (rs.next())
{

          // Getting the output the query result
          System.out.println(rs.getInt(1) + "/t" +    rs.getString(2));
}
```

Method 3: execute Update()

- **Description:** This method is used for execution of DML statement(INSERT, UPDATE and DELETE) which is return int value, count of the affected rows.
- **Usage:** This method is use to execute non select query. This method is use to execute select and non select queries.
- **Return type:** An integer value which represent number of rows affected by the query. This will be 0 for statement which are returning nothing.
- **Example:**

```
DML->INSERT , UPDATE and DELETE
DDL-> CREATE, ALTER
```

Illustration:

```
  Class.forName("com.mysql.jdbc.Driver");
    // Load the database driver
 Connection  conn = null;
 conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/test",
        "root","1234");
    // Use DriverManager to get database connection
 Statement  stmt = conn.createStatement();
    // Use Connection to create a Statement object


 return stmt.executeUpdate(sql);
```

```
    // Execute the DML statement and return the number of records
affected
```

Now let us finally conclude out the differences in return types as spotted in the above illustrations

1. ***execute():*** *The return type is Boolean, indicating whether ResultSet return*
2. ***executeQuery():*** *Type method returns a ResultSet, execute returns the results of the query, often used to perform the query*
3. ***executeUpdate():*** *The return type is int, that the implementation of a number of rows affected after the sql statement, usually used to execute modification statements.*

Comment

More info

Placement Training Program

Next Article

Difference between an Iterator and ListIterator in Java

Similar Reads

How to execute an SQL query and fetch results using PHP ?