

04 - JdbcTemplate

JdbcTemplate is the core component that simplifies the interaction with the database in a Spring-based application. It provides various methods for executing SQL operations like **insert, update, delete, and retrieving data.**

👉 update() method:

- update() method is used to execute insert, delete, and update queries in the database.
- It accepts two parameters:
 - SQL Query: The SQL statement to execute
 - Values: The values to pass into the SQL query
- The return value of update() is an integer that indicates the number of rows affected by the operation.

Example:

StudentRepo.java:

```
package com.telusko.SpringJDBCEx.repo;

@Repository
public class StudentRepo {
    private JdbcTemplate jdbc;
    public JdbcTemplate getJdbc() {
        return jdbc;
    }

    @Autowired
    public void setJdbc(JdbcTemplate jdbc) {
        this.jdbc = jdbc;
    }

    // Save method using JdbcTemplate
    public void save(Student s) {

        String sql = "INSERT INTO student (rollno, name, marks) VALUES (?, ?, ?)";
        int rows = jdbc.update(sql, s.getRollNo(), s.getName(), s.getMarks());
        System.out.println(rows + " row's affected.");
    }
}
```

```
// Method to retrieve all students
public List<Student> findAll() {
    List<Student> students = new ArrayList<>();
    return students;
}
}
```

- The JdbcTemplate object is injected into the StudentRepo class using Spring's @Autowired annotation. This provides the repository class with methods to interact with the database.
- In the save() method, the SQL INSERT query is defined, where the rollNo, name, and marks values are passed as query parameters.
- The jdbc.update() method is called with the query and student data that inserts the data into the database and returns the number of rows affected.

Code Link:

<https://github.com/navinreddy20/spring6-course/tree/c6690e4f2c70d8f530d70623f13d14ff0ffd7e7d/4%20Spring%20JDBC/4.4%20Jdbcmplate>