

Javarevisited

Blog about Java, Programming, Spring, Hibernate, Interview Questions, Books and Online Course Recommendations from Udemy, Pluralsight etc

[Home](#) [core java](#) [spring](#) [hibernate](#) [collections](#) [multithreading](#) [design patterns](#) [interview questions](#) [coding](#) [data structure](#) [OOP](#) [java 8](#) [books](#) [About Me](#)

[Java Certifications](#) [JDBC](#) [jsp-servlet](#) [JSON](#) [SQL](#) [Linux](#) [Courses](#) [online resources](#) [jvm-internals](#) [REST](#) [Eclipse](#) [jQuery](#) [Java IO](#) [Java XML](#)

THURSDAY, JUNE 15, 2017

JDBC Database Connection Pool in Spring Framework - How to Setup Example

Setting up JDBC Database Connection Pool in Spring framework is easy for any Java application, just matter of changing few configuration in spring configuration file. If you are writing core java application and not running on any web or application server like Tomcat or Weblogic, Managing Database connection pool using **Apache Commons DBCP** and Commons Pool along-with Spring framework is nice choice but if you have luxury of having web server and managed J2EE Container, consider using **Connection pool managed by J2EE server** those are better option in terms of maintenance, flexibility and also help to prevent [java.lang.OutOfMemoryError:PermGen Space in tomcat](#) by avoiding loading of JDBC driver in web-app class-loader, Also keeping JDBC connection pool information in Server makes it easy to change or include settings for JDBC over SSL. In this article we will see **how to setup Database connection pool in spring framework** using Apache commons DBCP and commons pool.jar

This article is in continuation of my tutorials on spring framework and database like [LDAP Authentication in J2EE with Spring Security](#) and [manage session using Spring security](#). If you haven't read those article than you may find them useful.

Spring Example JDBC Database Connection Pool

Spring framework provides convenient `JdbcTemplate` class for performing all Database related operation. if you are not using Hibernate than using Spring's `JdbcTemplate` is good option. `JdbcTemplate` requires a `DataSource` which is `javax.sql.DataSource` implementation and you can get this directly using [spring bean](#) configuration or by using **JNDI** if you are using [J2EE web server or application server](#) for managing Connection Pool. See *How to setup JDBC connection Pool in tomcat and Spring* for JNDI based connection pooling for more details. In order to setup `DataSource` you will require following configuration in your `applicationContext.xml` (spring configuration) file:

```
//DataSource connection settings in Spring
<bean id="springDataSource" class="org.apache.commons.dbcp.BasicDataSource" destroy-
method="close" >
    <property name="url" value="jdbc:oracle:thin:@localhost:1521:SPRING_TEST" />
    <property name="driverClassName" value="oracle.jdbc.driver.OracleDriver" />
    <property name="username" value="root" />
    <property name="password" value="root" />
    <property name="removeAbandoned" value="true"/>
    <property name="initialSize" value="20" />
    <property name="maxActive" value="30" />
</bean>

//Dao class configuration in spring
<bean id="EmployeeDatabaseBean" class="com.test.EmployeeDAOImpl">
    <property name="dataSource" ref="springDataSource"/>
</bean>
```

Below configuration of DBCP connection pool will create 20 database connection as `initialSize` is 20 and goes up to 30 Database connection if required as `maxActive` is 30. you can customize your database connection pool by using different properties provided by Apache DBCP library. Above example is creating connection pool with Oracle 11g database and we are

using `oracle.jdbc.driver.OracleDriver` comes along with `ojdbc6.jar` or `ojdbc6_g.jar`, to learn more about [how to connect Oracle database from Java program](#) see the link.

Java Code for using Connection pool in Spring



Below is complete code example of DAO class which uses Spring `JdbcTemplate` to execute SELECT query against database using database connection from Connection pool. If you are not initializing Database connection pool on start-up then it may take a while when you execute your first query because it needs to create certain number of SQL connection and then it execute query but once connection pool is created subsequent queries will execute faster.

```
//Code for DAO Class using Spring JdbcTemplate

package com.test;

import javax.sql.DataSource;
import org.log4j.Logger;
import org.log4j.LoggerFactory;
import org.springframework.jdbc.core.JdbcTemplate;

/**
 * Java Program example to use DBCP connection pool with Spring framework
 * @author Javin Paul
 */
public class EmployeeDAOImpl implements EmployeeDAO {

    private Logger logger = LoggerFactory.getLogger(EmployeeDAOImpl.class);
    private JdbcTemplate jdbcTemplate;

    public void setDataSource(DataSource dataSource) {
        this.jdbcTemplate = new JdbcTemplate(dataSource);
    }

    @Override
    public boolean isEmployeeExists(String emp_id) {
        try {
            logger.debug("Checking Employee in EMP table using Spring Jdbc Template");
            int number = this.jdbcTemplate.queryForInt("select count(*) from EMP where emp_id=?", emp_id);
            if (number > 0) {
                return true;
            }
        } catch (Exception exception) {
            exception.printStackTrace();
        }
        return false;
    }
}
```

Dependency:

1. you need to include oracle driver jar like `ojdbc6.jar` in you classpath.
2. Apache DBCP and commons pool jar in application classpath.

That's all on [how to configure JDBC Database connection pool in Spring framework](#). As I said its pretty easy using Apache DBCP library. Just matter of few configuration in spring `applicationContext.xml` and you are ready. If you want to configure *JDBC Connection pool on tomcat (JNDI connection pool)* and want to use in spring than see [here](#).

Further Reading

[Spring Framework 5: Beginner to Guru](#)
[Spring Master Class - Beginner to Expert](#)
[Introduction to Spring MVC 4 By Bryan Hansen](#)
[Spring in Action 4th edition by Craig Walls](#)

Other **Java JDBC** tutorials you may like

[4 JDBC performance tips for Java application](#)
[Why use prepared statement in Java](#)
[What is default bean scope in Spring framework](#)
[Difference between truncate and delete in SQL](#)
[10 Example of SQL select command for programmers](#)
[Difference between type 2 and type 4 JDBC drivers](#)

P.S. - If you are an experienced Java/JEE Program and want to learn Spring Security end-to-end, I recommend [Learn Spring Security](#) course by Eugen Paraschiv, The definitive guide to secure your Java application. It's useful for both junior and experienced Java Web developers.

He is also author of [REST with Spring](#) course, one of the best online course to learn RESTful WebServices using Spring framework.

You May Like

Find Cheap Flights From Krishna

Trips Shop

Hotels In India At Ridiculously Low Prices

Trips Insider

Sponsored Links

Javarevisited Spring Security Concurrent Session Control Example Tutorial - How to Limit Number of User Session in Java JEE Web...

If you don't know, Spring security can limit the number of sessions a user can have in a Java web application. If you are developing a we...

You May Like


Have You Met Ted? Watch All The 9 Seasons Of HIMYM On Your Phone Anywhere

Hotstar Premium

Buy DC get DR Free

Tech Data and DELL EMC

Sponsored Links

By Javin Paul at [June 15, 2017](#) 



Labels: [core java](#), [spring](#)

Location: [United States](#)

7 comments :

[Jirka Pinkas](#) said...

I use C3P0 instead of DBCP. DBCP has memory leaks (maybe they fixed it, but I don't know that for sure because I already jumped the ship) :)

And for Oracle database it's best to use DataSource from their JDBC driver, it has also connection pooling and better performance.

[June 16, 2012 at 2:17 AM](#)

Anonymous said...

ComboPooledDataSource from c3po jar is my preferred data source. c3po has proper default values for many configuration requires for connection pool as well. you just need to change your bean class to com.mchange.v2.c3p0.ComboPooledDataSource and include c3p0-0.9.1.jar

[June 18, 2012 at 10:25 PM](#)

[Yogesh](#) said...

nice info...

[August 1, 2013 at 12:10 AM](#)

Anonymous said...

can we use transaction manager if we are using c3po i spring. Because c3po is not closing my connections after i say trsansionmanager.commit(trsansionstatus)

[March 23, 2015 at 4:33 AM](#)

[Harsh](#) said...

java.lang.UnsupportedOperationException: Not supported by BasicDataSource at org.apache.tomcat.dbcp.dbcp.BasicDataSource.getConnection(BasicDataSource.java:899)

I am trying to save data thorough a form(index.jsp) but this exception continuously coming, do you have a solution for this ?? Thanks !!

[May 29, 2016 at 5:02 AM](#)

[Javin Paul](#) said...

Hello @Harsh, I am not sure which version of DBCP you are using, but in latest version getConnection(User,Password) method is not supported and that's why it's throwing java.lang.UnsupportedOperationException. It's better now to move to another connection pool like C3PO or tomcat connection pool. If you decides to use tomcat connection pool, then use org.apache.tomcat.jdbc.pool.DataSource class and tomcat-jdbc JAR as dependency.

See this for more details <https://tomcat.apache.org/tomcat-7.0-doc/jdbc-pool.html>

[May 29, 2016 at 5:35 AM](#)

Anonymous said...

HikariCP is the best JDBC connection pool out there.

[June 17, 2017 at 12:29 AM](#)

Post a Comment

Enter your comment...

Comment as:

Google Accou ▼

Publish

Preview

Javarevisited Does Standard Column Width of 80 On Coding Make Sense in 2018?

One of the oldest coding practice is to keep line width 80, and many of us follow it blindly but have you ever thought why we have this pra...

[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom \)](#)