

mysql8.0连接方式:

<!-- 配置数据库连接的驱动器 -->

```
<property name="hibernate.connection.driver_class">
```

```
com.mysql.cj.jdbc.Driver
```

```
</property>
```

<!-- 配置数据库连接的url -->

```
<property name="hibernate.connection.url">
```

```
jdbc:mysql://localhost:3306/hb2?
```

```
useSSL=false&serverTimezone=UTC&characterEncoding=utf8
```

```
</property>
```

<!-- 配置登录数据库的用户名 -->

```
<property name="hibernate.connection.username">root</property>
```

<!-- 配置登录数据库的密码 -->

```
<property
```

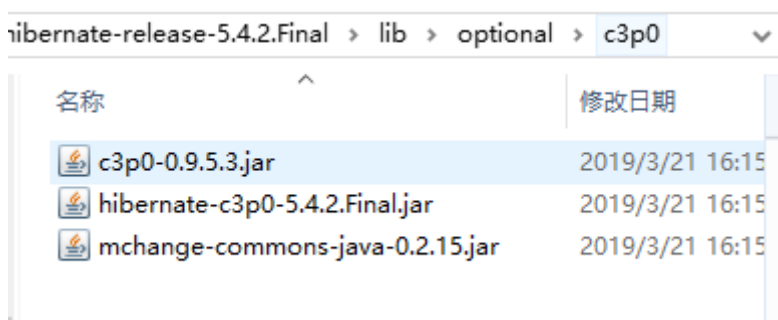
```
name="hibernate.connection.password">AQL271422</property>
```

虽然hibernate本身内置就有一个连接池，但是由于hibernate自身的原因，导致hibernate自身的连接池无论从性能还是其他的方面，都不是很好，这就需要我们整合一些其他的数据库连接池。

hibernate官方推荐的连接池是c3p0连接池：

整合步骤：

(1) 导入c3p0的整合相关导入，在hibernate解压后的文件夹下，lib/optional/c3p0下的三个jar包。



c3p0-0.9.5.3: c3p0的核心包

hibernate-c3p0-5.4.2.Final: hibernate与c3p0的整合包

mchange-commons-java-0.2.15: c3p0依赖包

(2) 在hibernate.cfg.xml文件中整合c3p0（其中值为hibernate-c3p0-5.4.2.Final包下org.hibernate.c3p0.internal.C3P0ConnectionProvider.class然后复制该文件名称）

```
<property name="hibernate.connection.provider_class">
    值
```

```
</property>
```

如：

```
<property name="hibernate.connection.provider_class">
    org.hibernate.c3p0.internal.C3P0ConnectionProvider
</property>
```

当然也可以根据实际项目的需求对其中的一些参数进行修改：

```
<!-- c3p0的详细配置 -->
```

```
<property name="c3p0.min_size">15</property>
```

```
<property name="c3p0.max_size">30</property>
```

查看配置成功的标志：

```
Session session = hibernateUtils.getSession();
    session.doWork(new Work() {
        @Override
        public void execute(Connection connection) throws
SQLException {
                                // TODO Auto-generated method stub
                                System.out.println(connection);
        }
    });

    session.close();
```

最终在控制台输出：

```
com.mchange.v2.c3p0.impl.NewProxyConnection@3eee3e2b [wrapping:
com.mysql.cj.jdbc.ConnectionImpl@4f63e3c7]
```

如果没有配置成功，则输出：

```
com.jdbc.....
```