自定义JDBCUtils工具类，要求如下：

1. 提供静态代码块加载配置文件，初始化连接池对象

2.获取连接方法：通过数据库连接池获取连接

3.获取连接池的方法

答案:

package cn.itcast.utils;

import com.alibaba.druid.pool.DruidDataSourceFactory;

import javax.sql.DataSource;

import java.io.IOException;

import java.sql.Connection;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.Properties;

/\*\*

\* Druid连接池的工具类

\*/

public class JDBCUtils {

//1.定义成员变量 DataSource

private static DataSource ds ;

static{

try {

//1.加载配置文件

Properties pro = new Properties();

pro.load(JDBCUtils.class.getClassLoader().getResourceAsStream("druid.properties"));

//2.获取DataSource

ds = DruidDataSourceFactory.createDataSource(pro);

} catch (IOException e) {

e.printStackTrace();

} catch (Exception e) {

e.printStackTrace();

}

}

/\*\*

\* 获取连接

\*/

public static Connection getConnection() throws SQLException {

return ds.getConnection();

}

/\*\*

\* 释放资源

\*/

public static void close(Statement stmt,Connection conn){

close(null,stmt,conn);

}

public static void close(ResultSet rs , Statement stmt, Connection conn){

if(rs != null){

try {

rs.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if(stmt != null){

try {

stmt.close();

} catch (SQLException e) {

e.printStackTrace();

}

}

if(conn != null){

try {

conn.close();//归还连接

} catch (SQLException e) {

e.printStackTrace();

}

}

}

/\*\*

\* 获取连接池方法

\*/

public static DataSource getDataSource(){

return ds;

}

}