# 环境说明

|  |  |
| --- | --- |
|  | 版本 |
| **Berkeley DB** | Berkeley DB Java Edition 5.0.97.zip (15M) md5 |

1. **安装数据库**

Berkeley DB Java Edition (JE)是一个完全用JAVA写的，它适合于管理海量的，简单的数据。

JE下载地址：

<http://www.oracle.com/technology/software/products/berkeley-db/je/index.html>

进入下载页面之后点击下载Berkeley DB Java Edition，具体情形如下图所示。



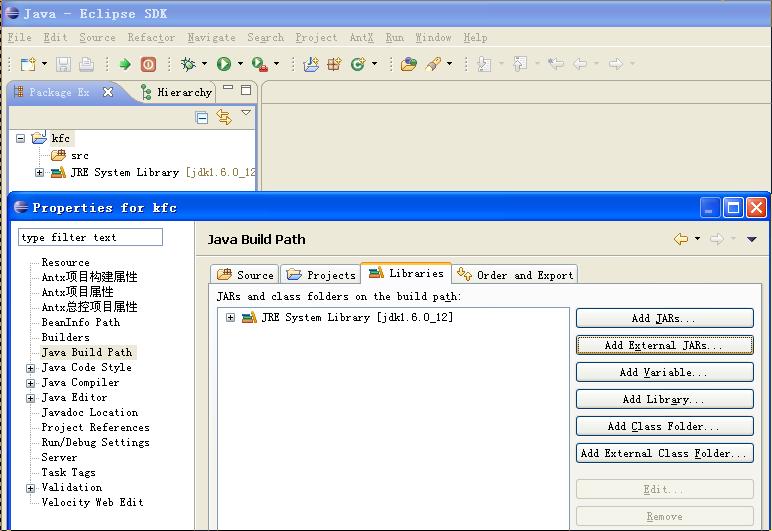
解压压缩包之后，把JE\_HOME/lib/je-<version>.jar中的jar文件添加到你的MyEclips中。

具体步骤:

1. 右键你的工程目录,点击最后一项属性Properties。
2. 选择第三项Java Build Path，再选择右边选项卡第三项Libraries。
3. 再选择第二个Add External JARS，然后找到你的jar包所在路径，然后添加进去。如下图所示。

相关帮助文档可以参考 *JE\_HOME*/docs/index.html

源代码见*JE\_HOME*/src/\*.\*



1. **创建数据库环境**

JE要求在任何DATABASE操作前，要先打开数据库环境，就像我们要使用数据库的话必须得先建立连接一样。你可以通过数据库环境来创建和打开database，或者更改database名称和删除database.

可以通过Environments对象来打开环境，打开环境的时候设置的目录必须是已经存在的目录，否则会出错误。默认情况下，如果指定的database不存在则不会自动创建一个新的database,但可以通过设置setAllowCreate来改变这一情况。

1. 创建database的环境

示例：

import com.sleepycat.je.DatabaseException;

import com.sleepycat.je.Environment;

import com.sleepycat.je.EnvironmentConfig;

import java.io.File;

...

Environment myDbEnvironment = null;

try {

    EnvironmentConfig envConfig = new EnvironmentConfig();

    envConfig.setAllowCreate(true);//如果不存在则创建一个

    myDbEnvironment = new Environment(new File("路径名"), envConfig);

} catch (DatabaseException dbe) {

// 错误处理

dbe.printStackTrace();

}

1. 关闭database的环境

可以通过Environment.close()这个方法来关闭database环境，当完成数据库操作后一定要关闭数据库环境，如有必要，在关闭数据库环境之前，需清理一下日志文件。

示例：

import com.sleepycat.je.DatabaseException;

import com.sleepycat.je.Environment;

...

try {

    if (myDbEnvironment != null) {

myDbEnvironment.cleanLog(); // 在关闭环境前清理下日志

      myDbEnvironment.close();

    }

} catch (DatabaseException dbe) {

    // 错误处理

dbe.printStackTrace();

}

1. 打开database

可以通过environment.openDatabase()方法打开一个database,在调用这个方法的时候必须指定database的名称。和databaseConfig() (注：数据库设置)

示例：

package je.gettingStarted;

import com.sleepycat.je.Database;

import com.sleepycat.je.DatabaseConfig;

import com.sleepycat.je.DatabaseException;

import com.sleepycat.je.Environment;

import com.sleepycat.je.EnvironmentConfig;

import java.io.File;

...

Environment myDbEnvironment = null;

Database myDatabase = null;

...

try {

    // 打开一个环境，如果不存在则创建一个

    EnvironmentConfig envConfig = new EnvironmentConfig();

    envConfig.setAllowCreate(true);

    myDbEnvironment = new Environment(new File("/export/dbEnv"), envConfig);

    // 打开一个数据库，如果数据库不存在则创建一个

    DatabaseConfig dbConfig = new DatabaseConfig();

    dbConfig.setAllowCreate(true);

    myDatabase = myDbEnvironment.openDatabase(null,

"sampleDatabase", dbConfig); //打开一个数据库，数据库名为

                                   //sampleDatabase,数据库的配置为dbConfig

} catch (DatabaseException dbe) {

    // 错误处理

}

1. 关闭database

通过调用Database.close()方法来关闭数据库，但要注意，在关闭数据库前必须得先把游标先关闭。

使用示例：

import com.sleepycat.je.DatabaseException;

import com.sleepycat.je.Database;

import com.sleepycat.je.Environment;

...

try {

        if (myDatabase != null) {

            myDatabase.close();

        }

        if (myDbEnvironment != null) {

            myDbEnvironment.close();

        }

} catch (DatabaseException dbe) {  // 错误处理

}