Java 核心技术

第七章 package, import和classpath

第三节 package和import—命令行

华东师范大学 陈良育

package和import(1)

- Java支持多个目录放置Java，并且通过package/import/classpath/jar等机制配合使用，可以支持多处地方放置和调用Java类。

- 利用Eclipse等IDE可以快速生成包、类和jar文件，可以快捷import所需要的class

package和import(2)

- 包名：和目录层次一样
  - cn.com.test.Man.java 必须放在cn\com\test目录下
- 类的完整名字：包名+类名
  - cn.com.test.Man
- 但是包具体放在什么位置不重要，编译和运行的时候通过classpath再指定。

package和import(3)

- 手动在c:\temp创建cn.com.test.Man.java
  - 即c:\temp\cn\com\test\Man.java
  - c:\temp可以替换成任何路径，后续命令同样替换
- 编译：
  - X:\>java c:\temp\cn\com\test\Man.java
- 运行
  - X:\>java -classpath .;c:\temp cn.com.test.Man
- X代表任意盘符或路径

package和import(4)

- java -classpath .;c:\temp cn.com.test.Man
- 第一部分：java，执行命令，是java.exe的简写。
- 第二部分：-classpath 固定格式参数，可以简写成-cp.
- 第三部分：是一个(Windows分号,Linux/Mac冒号连接起来的)字符串。按分隔符隔开，得到一个个子路径。当运行cn.com.test.Man类的过程中，如果需要用到其他的类，就会分裂第三部分的字符串，得到多个子路径，然后依次在每个路径下，再去寻找相应类（全称，包名以点隔开对应到目录）。
- 第四部分：主执行类的全称（含包名）

package和import(5)

- 编译和运行规则
  - 编译一个类，需要java文件的全路径，包括扩展名。
  - 运行一个类，需写类名全称（非文件路径），无须写扩展名。
  - 编译类的时候，需要给出这个类所依赖的类（包括依赖的类再次依赖的所有其他类）的所在路径。
  - 运行类的时候，需要给出这个类，以及被依赖类的路径总和。
  - classpath参数也可以包含jar包。如果路径内有空格，请将classpath参数整体加双引号。
  - java -classpath “.;c:\test.jar;c:\temp;c:\a bc” cn.com.test.Man

package和import(6)

类名（全称）	文件名	文件全路径
A.B.C	C.java	c:\m1\A\B\C.java
D.E.F(引用了 A.B.C)	F.java	c:\m2\D\E\F.java
G.H(引用了 D.E.F，没有直接但是间接引用了 A.B.C，因此仍然需要给出 A.B.C 的路径)	H.java	c:\m3\G\H.java

A.B.C	编译 X>javac c:\m1\A\B\C.java 运行 X>java –classpath .;c:\m1 A.B.C
D.E.F	编译 X>javac –classpath .;c:\m1 c:\m2\D\E\F.java 运行 X>java –classpath .;c:\m1;c:\m2 D.E.F
G.H	编译 X>javac –classpath .;c:\m1;c:\m2 c:\m3\G\H.java 运行 X>java –classpath .;c:\m1;c:\m2;c:\m3 G.H

package和import(7)

- 总结
  - 包名和类所在的目录必须严格一致
  - 在命令行中，必须依靠classpath来指引所需要的类
  - 编译需要文件的全路径，运行需要类的完整名字

代码(1)


```java
package A.B;
//一般来说，包名都是纯小写的。这里大写仅仅和类名统一

public class C {

    public static void main(String[] args) {
        new C().cook();
    }
    public void cook()
    {
        System.out.println("C: I can cook very well");
    }

}
```

代码(2) F类，依赖C类


```
package D.E;
//一般来说，包名都是纯小写的。这里大写仅仅和类名统一

import A.B.C;

public class F {

    public static void main(String[] args) {
        System.out.println("I can call C for cooking");
        new C().cook();
    }

    public void cook()
    {
        System.out.println("I can call C");
        new C().cook();
    }

}
```

代码(3) H类，依赖F类，间接依赖C类


```
package G;
//一般来说，包名都是纯小写的。这里大写仅仅和类名统一

import D.E.F;

public class H {

    public static void main(String[] args) {
        System.out.println("I can call F for cooking");
        new F().cook();
    }

    public void cook()
    {
        System.out.println("I can call F");
        new F().cook();
    }

}
```

谢 谢！