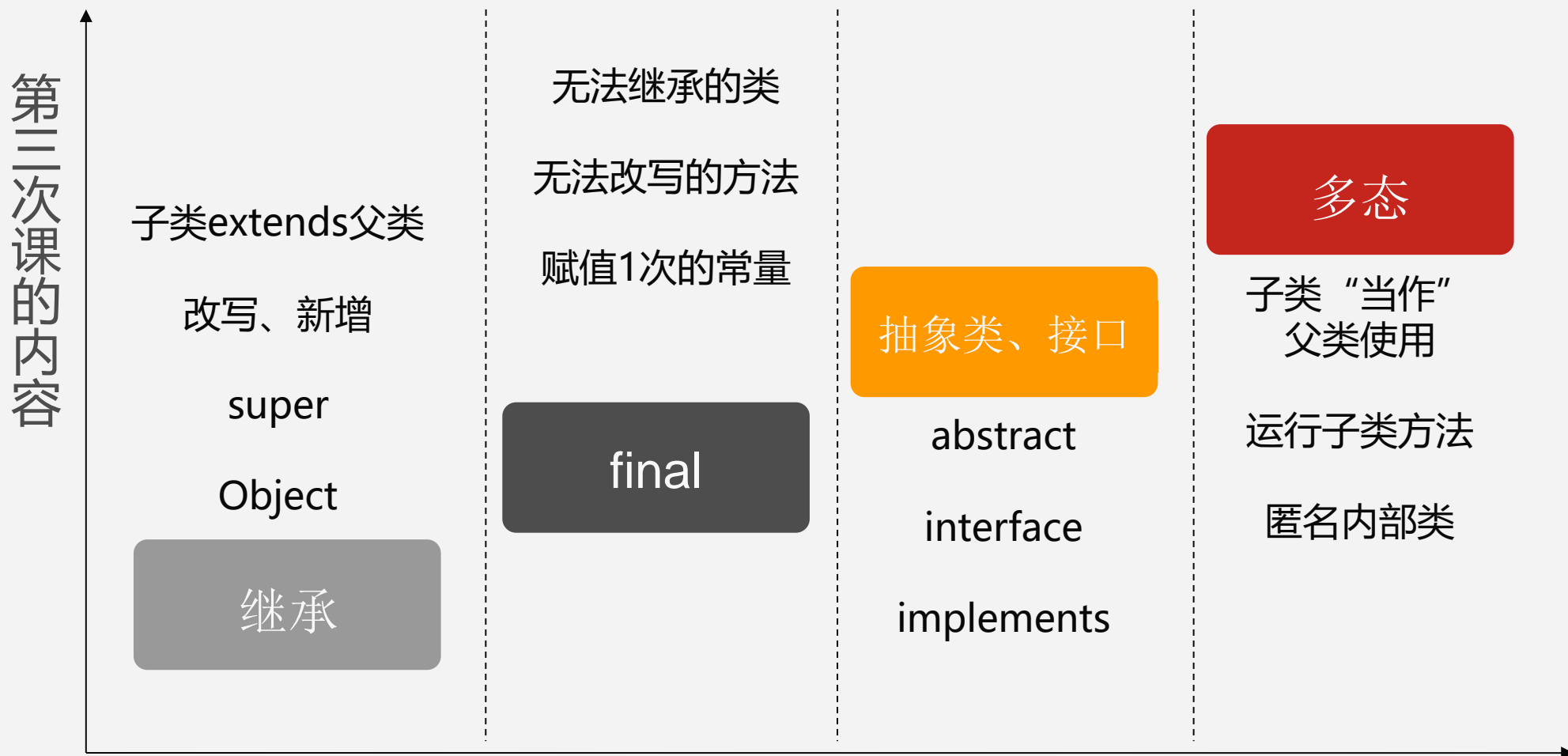




Java

面向对象程序设计

软件学院 贾伟峰



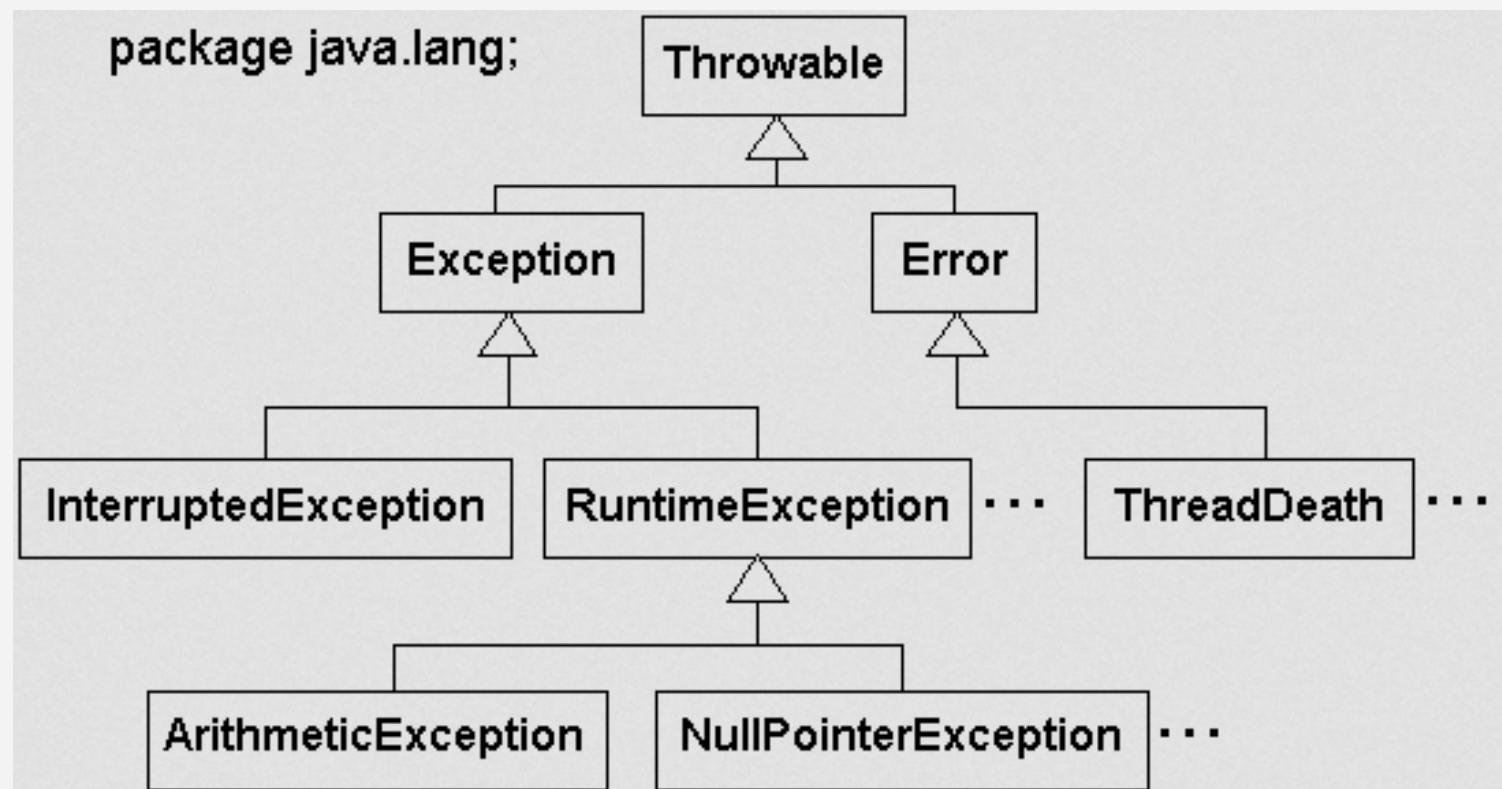
编程有时候会“出错”，比如下面的程序。

```
1 public class Example20{
2     public static void main(String[] args){
3         int result = divide(4, 0);
4         System.out.println(result);
5     }
6     public static int divide(int x, int y){
7         int result = x/y;
8         return result;
9     }
10 }
```

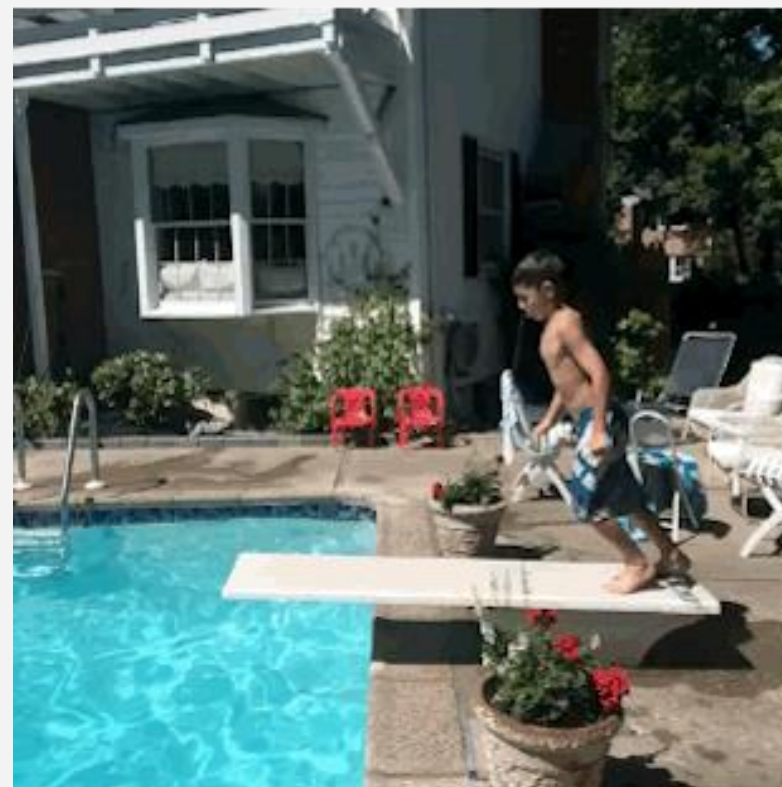
```
Exception in thread "main" java.lang.ArithmeticException: / by zero
    at Example20.divide(Example20.java:7)
    at Example20.main(Example20.java:3)
```

有 “异常” : *Exception*

Java中的异常(Exception)、错误(Error)

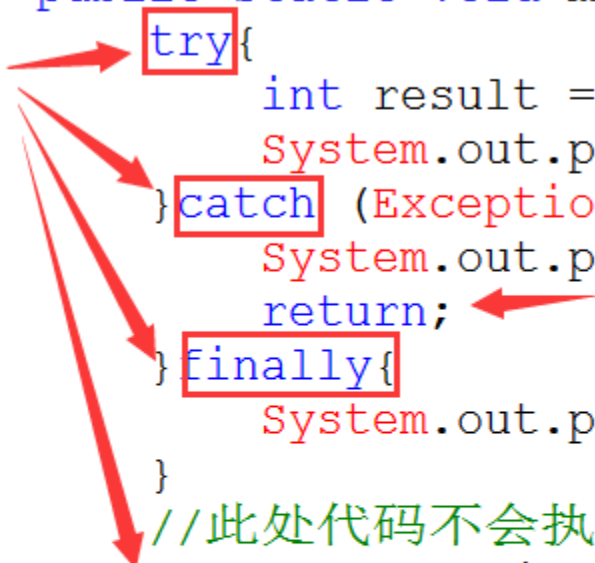


能否捕捉程序中的异常？如图所示。



```
1 public class Example20{
2     public static void main(String[] args){
3         try{
4             int result = divide(4, 0);
5             System.out.println(result);
6         }catch (Exception e){
7             System.out.println("捕捉的异常信息为: "+e.getMessage());
8         }
9     }
10    public static int divide(int x, int y){
11        int result = x/y;
12        return result;
13    }
14 }
```

```
1 public class Example20{
2     public static void main(String[] args){
3         try{
4             int result = divide(4, 0);
5             System.out.println(result);
6         } catch (Exception e){
7             System.out.println("捕捉的异常信息为: "+e.getMessage());
8             return;
9         } finally{
10             System.out.println("进入finally代码块");
11         }
12         //此处代码不会执行
13         System.out.println("程序继续向下执行.....");
14     }
15     public static int divide(int x, int y){
16         int result = x/y;
17         return result;
18     }
19 }
```



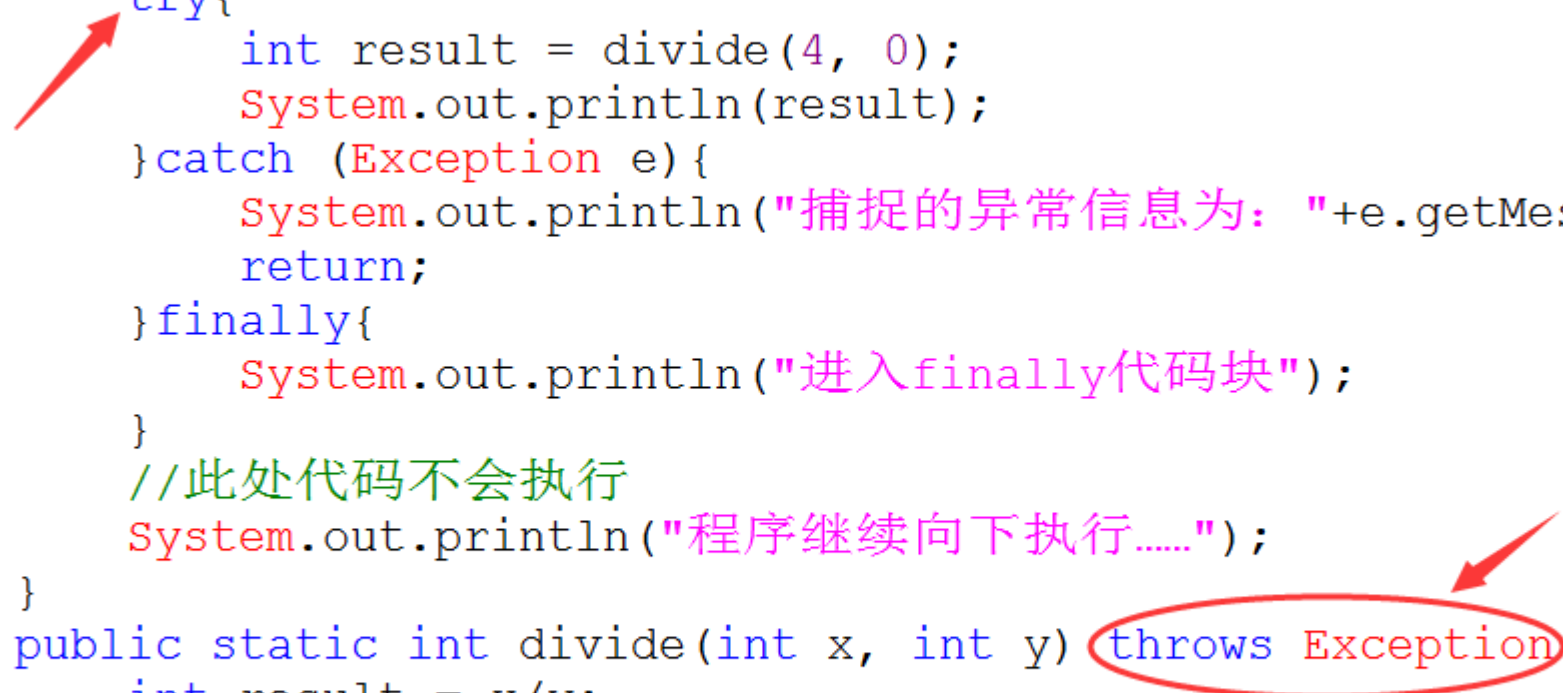


程序员就是不写异常捕捉（处理）代码怎么办？



throws

```
1 public class Example20{
2     public static void main(String[] args){
3         try{
4             int result = divide(4, 0);
5             System.out.println(result);
6         }catch (Exception e){
7             System.out.println("捕捉的异常信息为: "+e.getMessage());
8             return;
9         }finally{
10             System.out.println("进入finally代码块");
11         }
12         //此处代码不会执行
13         System.out.println("程序继续向下执行.....");
14     }
15     public static int divide(int x, int y) throws Exception{
16         int result = x/y;
17         return result;
18     }
19 }
```



或者，继续*throws*，就像p146那样

编译时异常

01

02

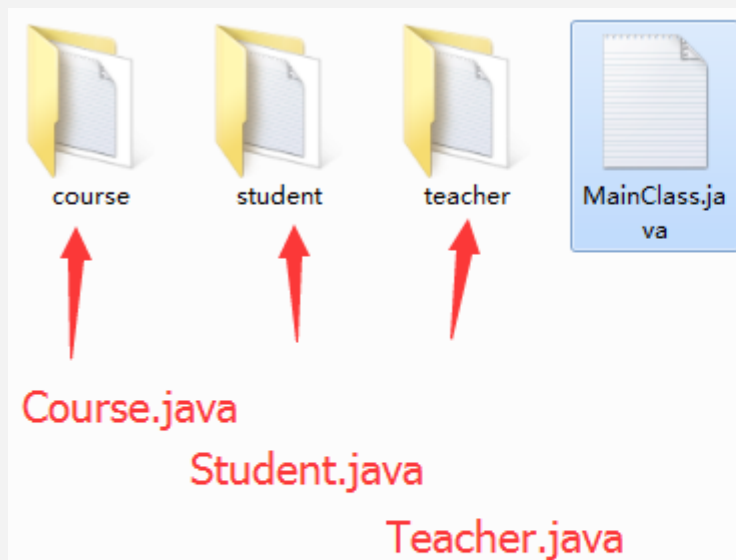
运行时异常

还可以自定义异常！ P147-149

太多文件，如何更合适地管理？用package



java文件所在目录src



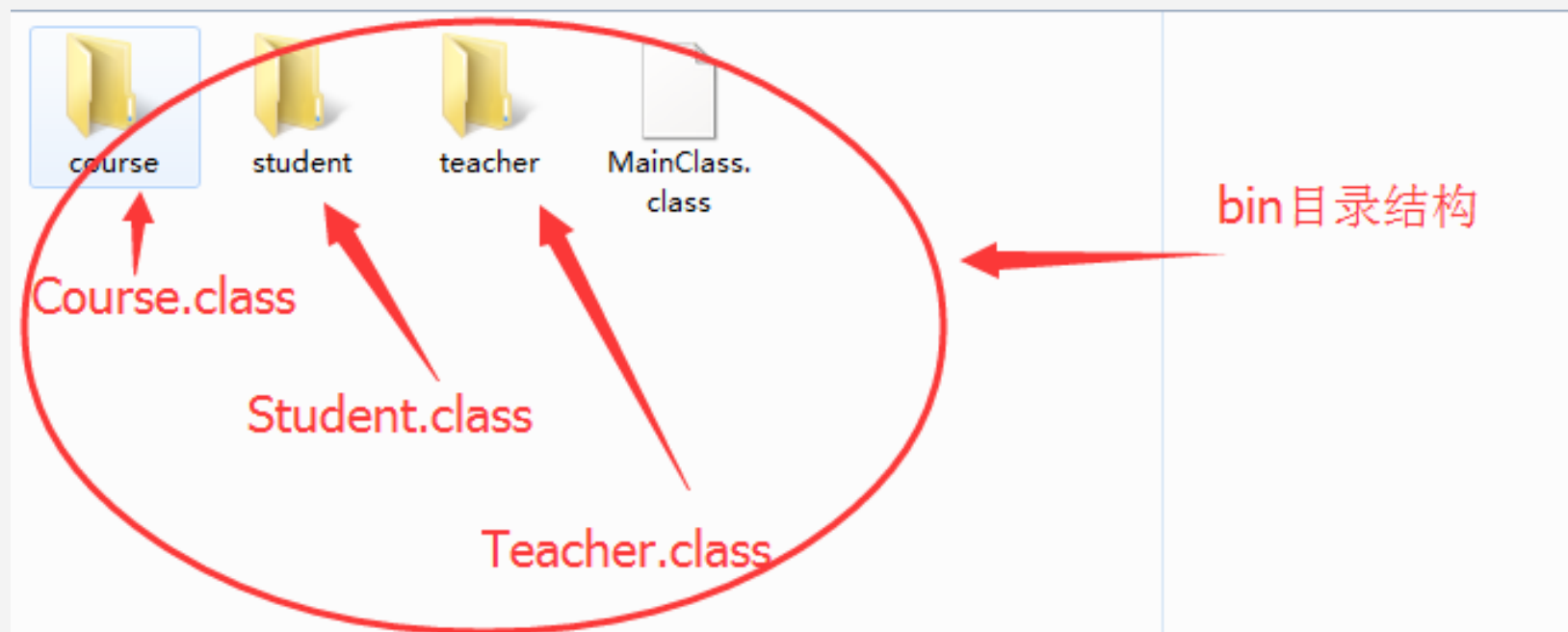
```
import course.*;  
import student.*;  
import teacher.*;
```

```
public class MainClass  
{  
    public static void main(String[] args) {  
        Course cs = new Course();  
        Student st = new Student();  
        Teacher te = new Teacher();  
    }  
}
```

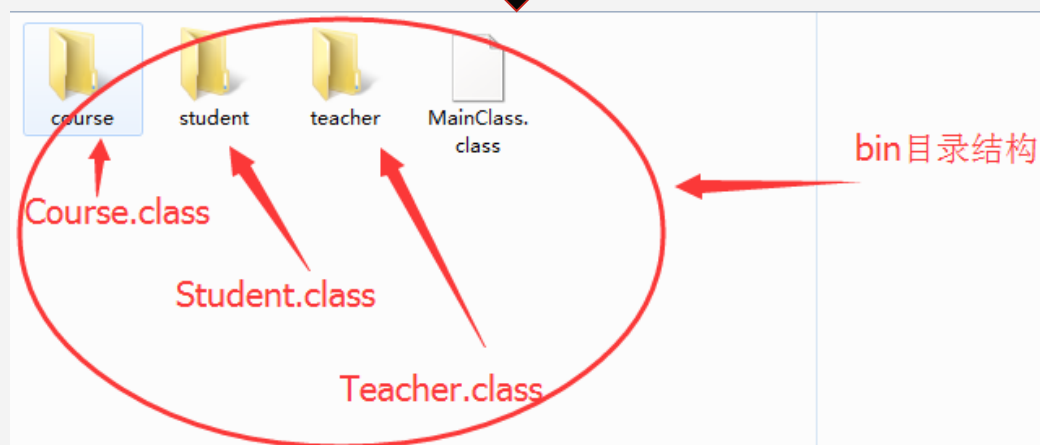
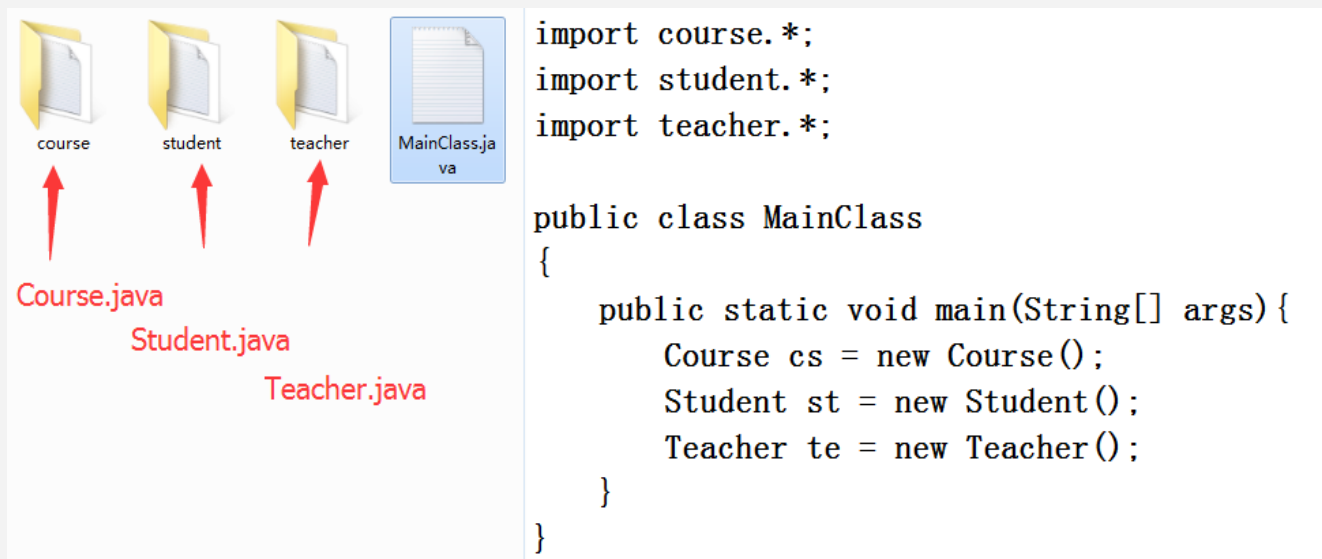
```
1 package teacher; ←  
2 public class Teacher  
3 {  
4     public Teacher() {  
5         System.out.println("This is a teacher.");  
6     }  
7 }
```

```
1 package course; ←  
2 public class Course  
3 {  
4     public Course() {  
5         System.out.println("This is a course.");  
6     }  
7 }
```

```
1 package student; ←  
2 public class Student  
3 {  
4     public Student()  
5     {  
6         System.out.println("This is a student.");  
7     }  
8 }
```



如何从src的java文件，变成bin中的class文件？



```
pkgdaemon\src>javac -d ../bin course/Course.java
```

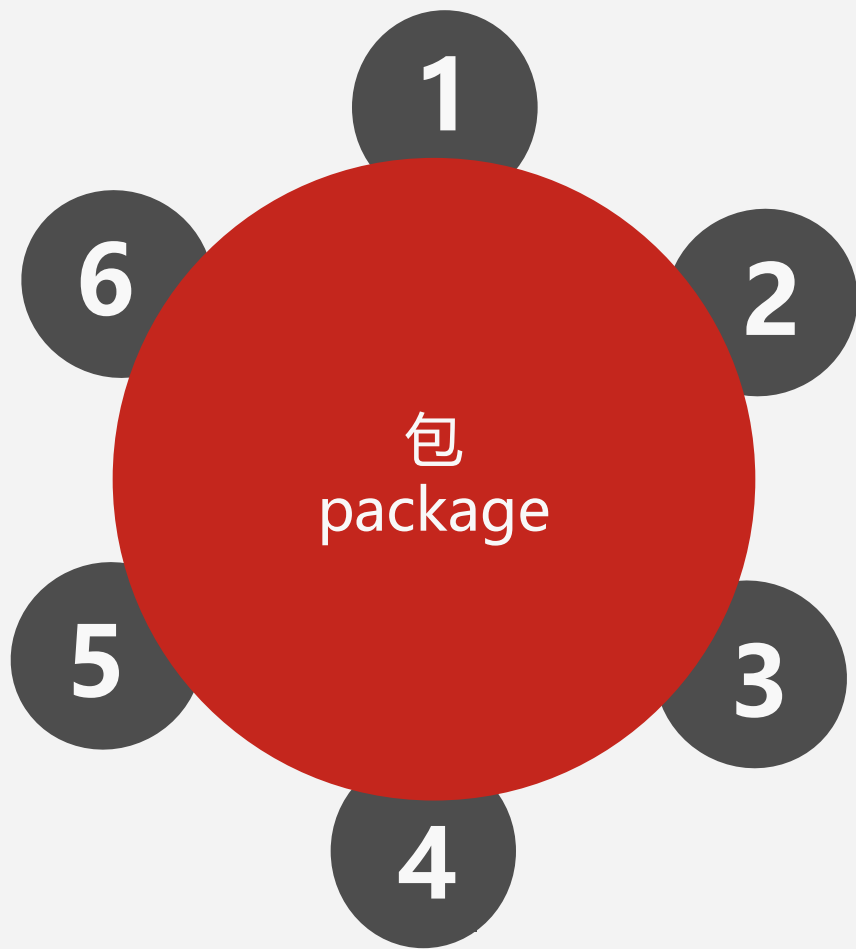
```
pkgdaemon\src>javac -d ../bin student/Student.java
```

```
pkgdaemon\src>javac -d ../bin teacher/Teacher.java
```

```
pkgdaemon\src>javac -d ../bin MainClass.java
```

```
pkgdaemon\bin>java MainClass
```

java中自带的包



java.lang

java.util

java.net

java.io

java.awt

java.sql



这么多的class，如何打包成1个供使用？

jar

```
\pkgdaemon\bin>jar -cvf MainClass.jar .
```

↑
在有**Main**方法的那个类所在的文件夹中执行。

改MANIFEST.MF, 加入口类 (P156) 。

```
C:\>java -jar MainClass.jar
```

运行打包后的文件



访问控制

■private: 类内访问

■default: 包内访问

■protect: 包内访问+继承（子类）访问

■public: 不受限

