

Lab 04 Assignment

班级: 202111

学号: 19241027

姓名: 胡峻诚

Question1. Package

在 A.java 添加:

```
1 | package com.oo.aa;
```

在 B.java 添加:

```
1 | package com.oo.bb;
2 | import com.oo.aa.A;
```

在 C.java 添加:

```
1 | package com.oo.cc;
2 | import com.oo.bb.B;
```

在 Main.java 添加:

```
1 | package com.oo;
2 | import com.oo.cc.C;
```

在 BTest.java 添加:

```
1 | package test;
2 | import com.oo.bb.B;
```

运行效果:



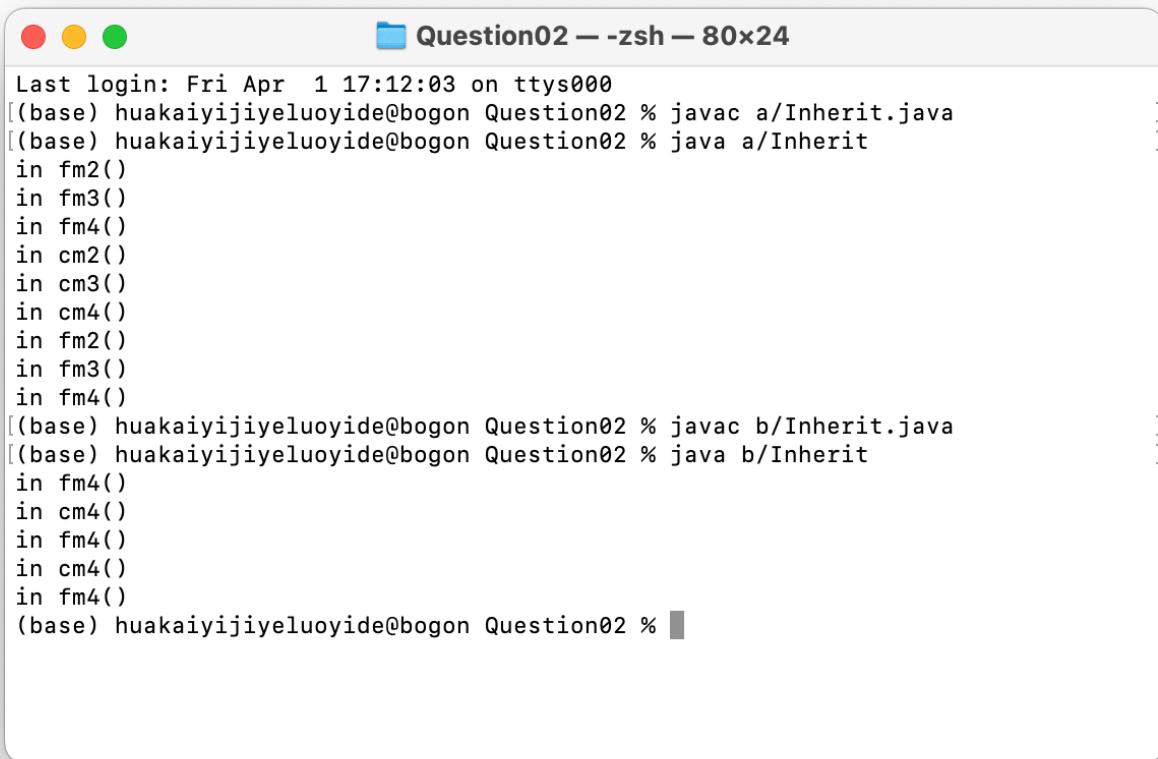
Question01 — -zsh — 80x24

```
Last login: Fri Apr  1 17:01:38 on ttys000
(base) huakaiyijiye@bogon Question01 % javac -cp ./src/ -d ./bin src/com/
oo/Main.java
[java -cp ./bin/ com.oo.Main
main starts
A1
B1
C1
yeah~
(base) huakaiyijiye@bogon Question01 % javac -cp ./src/ -d ./bin src/test
/BTest.java
[java -ea -cp ./bin/ test.BTest
b test starts
A1
B1
yeah~
(base) huakaiyijiye@bogon Question01 % ]
```

Question2. 权限

1. private 表示私有，只有自己类能访问
2. default 表示没有修饰符修饰，只有同一个包的类能访问
3. protected 表示可以被同一个包的类以及其他包中的子类访问
4. public 表示可以被该项目的所有包中的所有类访问

运行效果：



```
Last login: Fri Apr  1 17:12:03 on ttys000
[(base) huakaiyijiye@bogon Question02 % javac a/Inherit.java
[(base) huakaiyijiye@bogon Question02 % java a/Inherit
in fm2()
in fm3()
in fm4()
in cm2()
in cm3()
in cm4()
in fm2()
in fm3()
in fm4()
[(base) huakaiyijiye@bogon Question02 % javac b/Inherit.java
[(base) huakaiyijiye@bogon Question02 % java b/Inherit
in fm4()
in cm4()
in fm4()
in cm4()
in fm4()
(base) huakaiyijiye@bogon Question02 %
```

Question3. Hide & Override

1. 运行结果：

```

public class Test {
    public static void main(String[] args) {
        Parent f1 = new Parent();
        System.out.println(f1.num);

        Parent f2 = new Child();
        System.out.println(f2.num);

        Child c = new Child();
        System.out.println(c.num);

        f1.foo();
        f2.foo();
        c.foo();

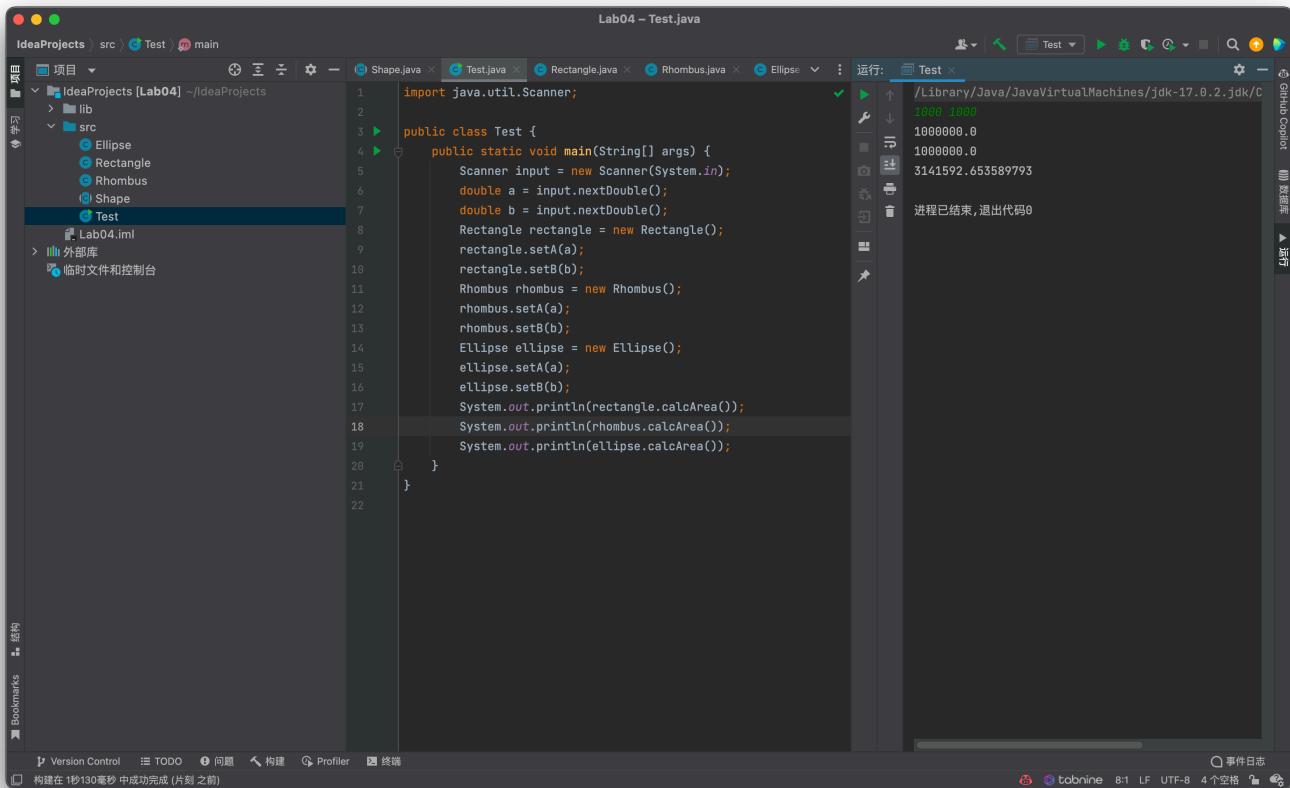
        f1.bar();
        f2.bar();
        c.bar();
    }
}

```

进程已结束,退出代码0

2. 不能，子类重写父类的方法不能降低权限。
3. 不能，子类重写父类的方法不能降低权限。
4. 不一致，静态方法会被隐藏，非静态方法会被重写，因此 `f2.bar()` 调用父类函数，`f2.foo()` 便调用子类函数。
5. 只有非静态且非私有的方法可以重写，其余的均是隐藏。

Question4. shape



The screenshot shows the IntelliJ IDEA interface with the following details:

- Project Structure:** The project is named "Lab04". It contains a "src" directory which includes "lib", "Shape", "Rhombus", "Rectangle", and "Ellipse" classes, and a "Test" class.
- Code Editor:** The "Test.java" file is open, containing the following Java code:

```

import java.util.Scanner;

public class Test {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        double a = input.nextDouble();
        double b = input.nextDouble();
        Rectangle rectangle = new Rectangle();
        rectangle.setA(a);
        rectangle.setB(b);
        Rhombus rhombus = new Rhombus();
        rhombus.setA(a);
        rhombus.setB(b);
        Ellipse ellipse = new Ellipse();
        ellipse.setA(a);
        ellipse.setB(b);
        System.out.println(rectangle.calcArea());
        System.out.println(rhombus.calcArea());
        System.out.println(ellipse.calcArea());
    }
}

```
- Run Tab:** The "Test" tab is selected in the run configuration dropdown.
- Output Window:** The output shows the results of the program execution:

```

1000 1000
1000000.0
1000000.0
3141592.653589793
进程已结束,退出代码0

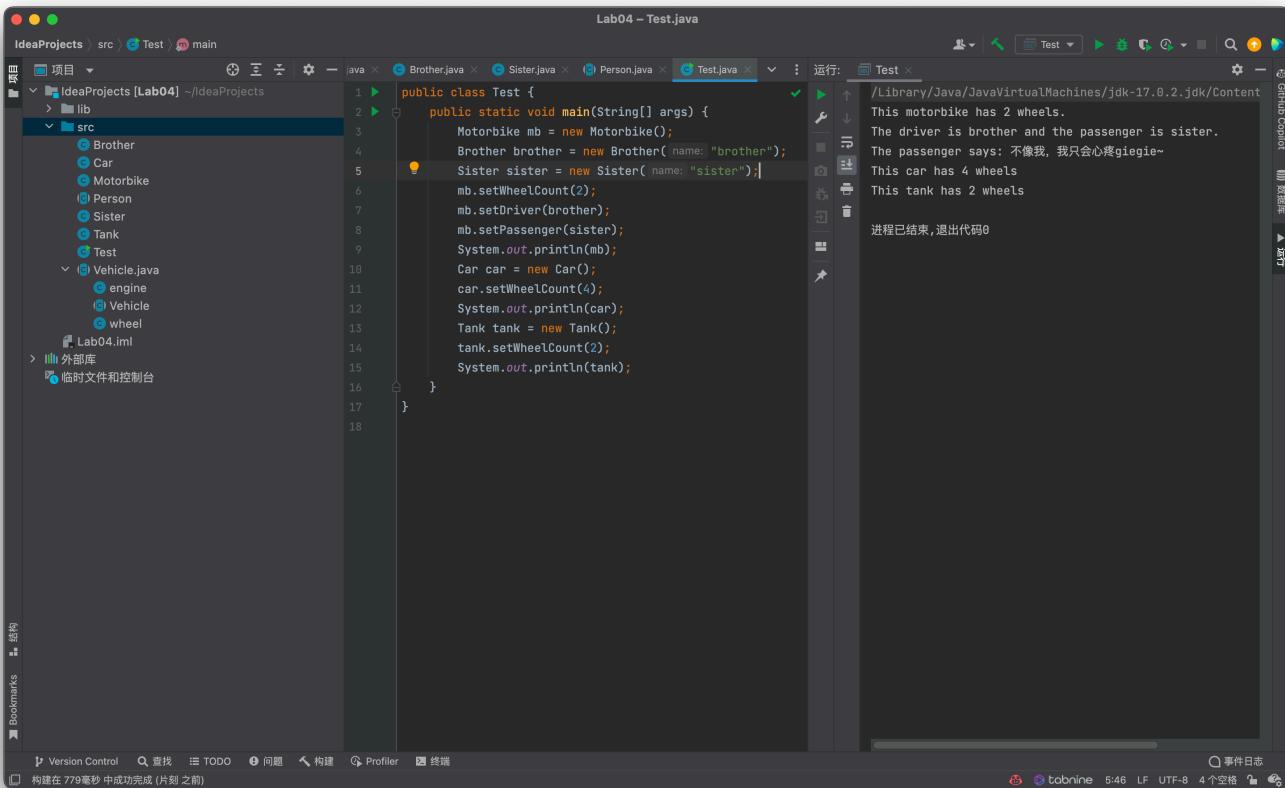
```
- Bottom Status Bar:** Shows "构建在 1秒130毫秒 中成功完成 (片刻之前)" and "tabnine 8:1 LF UTF-8 4个空格".

题外话:

我们知道属性不设置为 `public` 是为了保护数据, 如果 `a` 和 `b` 在 `Shape` 中的访问权限被设置为 `private`, 在不添加新的属性的情况下, 子类想要实现自己的 `calcArea()` 该怎么办?

父类设置 `getter` 方法, 为公开的, 子类调用父类的 `getter` 方法即可。

Question5. 车车



The screenshot shows the IntelliJ IDEA interface with the project 'Lab04' open. The 'src' directory contains several classes: Brother, Car, Motorbike, Person, Sister, Tank, Test, Vehicle.java (which includes engine, Vehicle, and wheel), and a file Lab04.iml. The 'Test.java' file is currently selected and contains the following code:

```

public class Test {
    public static void main(String[] args) {
        Motorbike mb = new Motorbike();
        Brother brother = new Brother(name: "brother");
        Sister sister = new Sister(name: "sister");
        mb.setWheelCount(2);
        mb.setDriver(brother);
        mb.setPassenger(sister);
        System.out.println(mb);
        Car car = new Car();
        car.setWheelCount(4);
        System.out.println(car);
        Tank tank = new Tank();
        tank.setWheelCount(2);
        System.out.println(tank);
    }
}

```

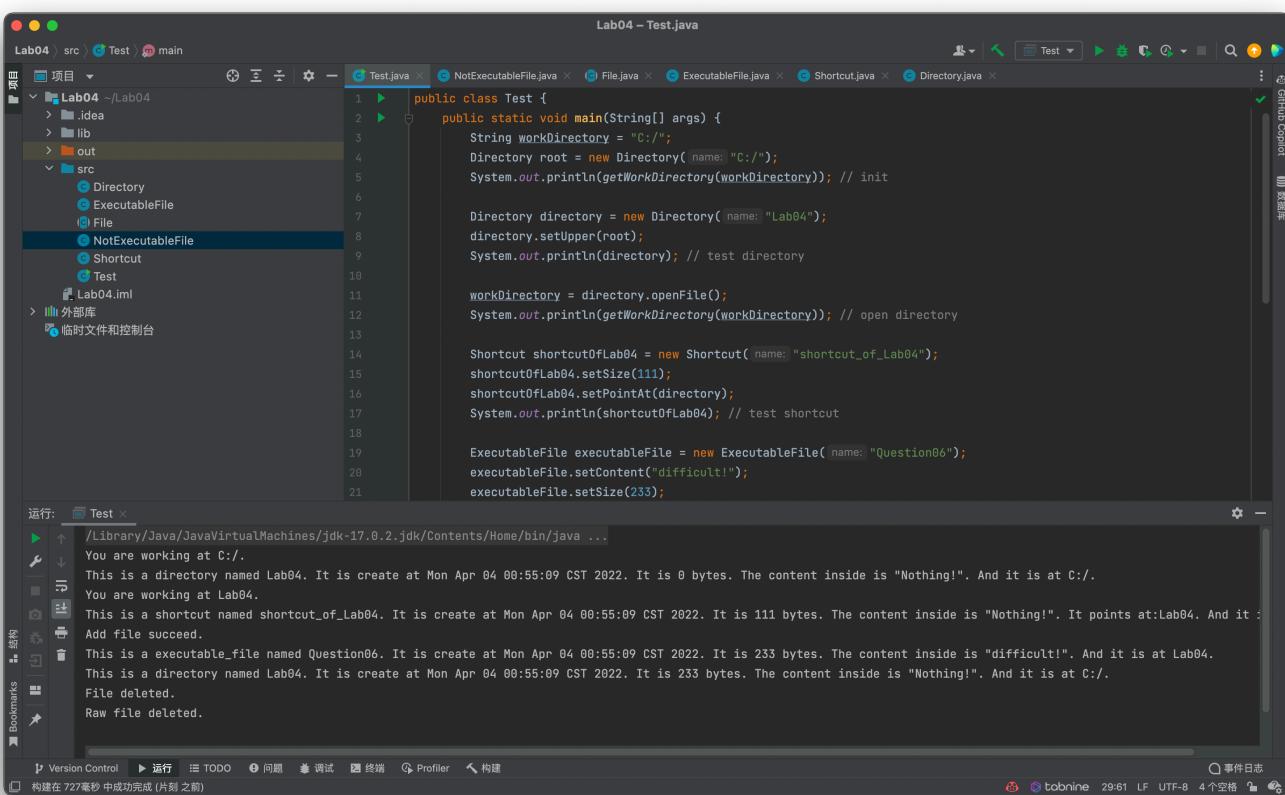
The 'Run' tool window shows the output of the program:

```

/Library/Java/JavaVirtualMachines/jdk-17.0.2.jdk/Content
This motorbike has 2 wheels.
The driver is brother and the passenger is sister.
The passenger says: 不像我, 我只会心疼giegie~
This car has 4 wheels
This tank has 2 wheels
进程已结束,退出代码0

```

Question6. 其实 OOP Lab 里面也可以写操作系统



The screenshot shows the IntelliJ IDEA interface with the project 'Lab04' open. The 'src' directory contains classes for Directory, ExecutableFile, File, NotExecutableFile, Shortcut, and Test, along with a file Lab04.iml. The 'Test.java' file is selected and contains the following code:

```

public class Test {
    public static void main(String[] args) {
        String workDirectory = "C:/";
        Directory root = new Directory(name: "C/");
        System.out.println(getWorkDirectory(workDirectory)); // init

        Directory directory = new Directory(name: "Lab04");
        directory.setParent(root);
        System.out.println(directory); // test directory

        workDirectory = directory.openFile();
        System.out.println(getWorkDirectory(workDirectory)); // open directory

        Shortcut shortcutOfLab04 = new Shortcut(name: "shortcut_of_Lab04");
        shortcutOfLab04.setSize(111);
        shortcutOfLab04.setPointAt(directory);
        System.out.println(shortcutOfLab04); // test shortcut

        ExecutableFile executableFile = new ExecutableFile(name: "Question06");
        executableFile.setContent("difficult!");
        executableFile.setSize(233);
    }
}

```

The 'Run' tool window shows the output of the program, demonstrating the creation and manipulation of files and directories:

```

/ Library/Java/JavaVirtualMachines/jdk-17.0.2.jdk/Contents/Home/bin/java ...
You are working at C:/.
This is a directory named Lab04. It is create at Mon Apr 04 00:55:09 CST 2022. It is 0 bytes. The content inside is "Nothing!". And it is at C:/.
You are working at Lab04.
This is a shortcut named shortcut_of_Lab04. It is create at Mon Apr 04 00:55:09 CST 2022. It is 111 bytes. The content inside is "Nothing!". It points at:Lab04. And it :
Add file succeed.
This is a executable_file named Question06. It is create at Mon Apr 04 00:55:09 CST 2022. It is 233 bytes. The content inside is "difficult!". And it is at Lab04.
This is a directory named Lab04. It is create at Mon Apr 04 00:55:09 CST 2022. It is 233 bytes. The content inside is "Nothing!". And it is at C:/.
File deleted.
Raw file deleted.

```