

# 隨堂練習卷

建議作答 50 分鐘 · 總分 118

姓名：\_\_\_\_\_ 學號：\_\_\_\_\_ 班級：\_\_\_\_\_

得分：\_\_\_\_\_

本章為自學延伸、非課程考試範圍，本卷供自修自測用。作答說明：紙筆測驗，請勿使用電腦。程式題寫出完整可編譯程式碼，注意大小寫、分號與大括號。

## § 1 一、選擇題 (每題 6 分，共 30 分)

1. 「同一個方法呼叫，依物件實際型別跑出不同版本的行為」稱為？ (A) 多載 (B) 多型 (C) 封裝 (D) 組合
2. `Shape s = new Circle();` 中，`s.area()` 實際跑哪個版本由什麼決定？ (A) 變數 `s` 的宣告型別 `Shape` (B) 物件的實際型別 `Circle` (C) 編譯期隨機 (D) 一定跑 `Shape` 版
3. 下列關於**抽象類別**的敘述，何者**正確**？ (A) 可以直接 `new` (B) 不能有建構子 (C) 不能 `new`、只能被繼承 (D) 不能有一般方法
4. 一個類別與介面的關係，下列何者**正確**？ (A) 類別只能 `implements` 一個介面 (B) 類別能 `implements` 多個介面、但只能 `extends` 一個父類別 (C) 介面可以被 `new` (D) 類別用 `extends` 履行介面
5. 把 `superclass` 型別的變數轉回 `subclass` 型別前，應先用什麼確認實際型別以免 `ClassCastException`？ (A) `instanceof` (B) `==` (C) `equals` (D) `getClass` 強制

答：1.    2.    3.    4.    5.   

## § 2 二、概念追蹤 (共 16 分)

1. 追蹤下列程式，寫出每行 `println` 的輸出 (8 分)：

```
public class Demo {
    static class Animal { public String speak() { return "..."; } }
    static class Dog extends Animal { public String speak() { return "汪"; } }
    static class Cat extends Animal { public String speak() { return "喵"; } }
    public static void main(String[] args) {
        Animal a = new Dog();
        System.out.println(a.speak());
        a = new Cat();
        System.out.println(a.speak());
        Animal b = new Animal();
        System.out.println(b.speak());
    }
}
```

行	輸出
第 1 個 <code>a.speak()</code>	
第 2 個 <code>a.speak()</code>	
<code>b.speak()</code>	

1. 下列程式哪一行會 `ClassCastException`？說明為什麼，並寫出怎麼用 `instanceof` 避免 (8 分)：

```
Shape[] shapes = { new Circle(2), new Square(3) };
Circle c = (Circle) shapes[1]; // shapes[1] 實際是 Square
System.out.println(c.getRadius());
```


### § 3 三、改錯 (每題 6 分，共 18 分)

各有一個錯誤，指出並改正。

1. 想實例化抽象類別：

```
abstract class Shape { abstract double area(); } ... Shape s = new Shape();
```

- 類別想履行介面卻用錯關鍵字：`class Circle extends Drawable { public void draw(){} }`  
(`Drawable` 是 interface)
- 子類別想覆寫父類別的 `final` 方法：`class Account { public final double get(){return 0;} }` `class Saving extends Account { public double get(){return 1;} }`


## § 4 四、程式設計 (共 54 分)

- (18 分) 設計抽象類別 `Shape`，含抽象方法 `double area()`。寫兩個具象子類別 `Circle` (半徑) 與 `Square` (邊長)，各 `@Override` 實作 `area()`。在 `main` 用 `Shape[]` 裝一個 `Circle(2)` 與一個 `Square(3)`，用增強 `for` 印出各自面積 (保留兩位，預期 `12.57`、`9.00`)。


- (18 分) 定義介面 `Payable { double getPaymentAmount(); }`，由 `Invoice` (含 `int qty`、`double price`，金額 = `qty × price`) 與 `SalariedEmployee` (含 `double salary`，金額 = `salary`)

各實作。在 `main` 用 `Payable[]` 裝兩者，迴圈印出每個的應付金額（保留兩位，預期 `59.97`、`50000.00`）。


13. (18 分) 定義 `Flyable { void fly(); }` 與 `Swimmable { void swim(); }`，讓 `Duck` 同時 `implements` 兩者。在 `main` 建一隻 `Duck`，各呼叫 `fly()`（印 鴨子拍翅飛）與 `swim()`（印 鴨子划水游）。
