

# 隨堂練習卷 (自學延伸)

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，本卷供想多練的人自測。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. 繼承表達的是哪一種關係？ (A) has-a (內含) (B) is-a (是一種) (C) use-a (使用) (D) 沒有關係
2. 子類別要繼承父類別，用哪個關鍵字？ (A) `implements` (B) `instanceof` (C) `extends` (D) `super`
3. 下列關於 `protected` 成員的敘述，何者**正確**？ (A) 只有自己類別可見 (B) 任何類別都可見 (C) 自己、子類別與同套件可見 (D) 只有子類別可見、同套件不行
4. 子類別重新定義父類別「同名、同參數列、同回傳型別」的方法，稱為？ (A) 多載 (overloading) (B) 覆寫 (overriding) (C) 封裝 (D) 組合
5. 下列三層繼承，`new C()` 的輸出順序是？

```
class A { A(){ System.out.println("A"); } }  
class B extends A { B(){ System.out.println("B"); } }  
class C extends B { C(){ System.out.println("C"); } }
```

(A) C B A (B) A B C (C) C A B (D) 只印 C

答：1. \_ 2. \_ 3. \_ 4. \_ 5. \_\_\_\_\_

## § 2 二、概念追蹤 (共 16 分)

1. 說明下列為何能編譯並印出 `旺財`，`Dog` 明明沒寫 `getName()` (8 分)：

```
class Animal {
    private String name;
    public Animal(String name) { this.name = name; }
    public String getName() { return name; }
}
class Dog extends Animal {
    public Dog(String name) { super(name); }
}
// new Dog("旺財").getName()
```


1. 寫出下列程式的輸出，並說明 `super.speak()` 的作用 (8分)：

```
class Animal { public String speak() { return " (某種聲音) "; } }
class Dog extends Animal {
    @Override public String speak() { return super.speak() + " → 汪汪"; }
}
// System.out.println(new Dog().speak());
```


### § 3 三、改錯 (每題 6 分，共 18 分)

各有一個錯誤，指出並改正。

1. 想讓子類別覆寫父方法 `speak()`，卻沒被呼叫到：`class Dog extends Animal { @Override public String Speak() { return "汪汪"; } }` (父類別有 `public String speak()`)。

- 父類別只有帶參數建構子，子類別沒寫 `super(...)`：

```
class Animal { Animal(String n){} }  
class Dog extends Animal { Dog(){ } }
```
- 覆寫 `equals` 卻寫成多載：

```
class Point { int x; @Override public boolean equals(Point  
p){ return x==p.x; } }
```


## § 4 四、程式設計 (共 54 分)

- (18 分) 設計父類別 `Vehicle` (`protected String name`、建構子、`move()` 回 `name + " 移動中"`) 與子類別 `Car` (建構子用 `super(name)`，覆寫 `move()` 回 `name + " 在路上行駛"`)。在 `main` 各建一個 `Vehicle("交通工具")` 與 `Car("轎車")`，分別呼叫 `move()` 印出。


- (18 分) 設計三層繼承：`Account` (`protected double balance`、建構子設初始餘額、`toString()` 回 `餘額:balance`) ← `SavingsAccount` (多 `interestRate`，覆寫 `toString()` 用

`super.toString()` 串接利率) 。在 `main` 建一個 `SavingsAccount(1000, 0.02)` 並印出 (預期 餘額：1000.0，利率：0.02) 。


13. (18 分) 設計 `Point (private int x, y)`，覆寫 `toString()` 回 `(x, y)`、覆寫 `equals(Object o)` 比座標是否相同 (先 `instanceof` 再轉型) 。在 `main` 建兩個 `(3,5)` 的點，印出 `==` 與 `equals` 的結果 (預期 `==` 為 `false`、`equals` 為 `true`) 。
