

說明

1. 第①題：要使用 Scanner 物件接受輸入的資料，要載入 java.util.Scanner 套件，所以答案為(B)。程式碼請參考 Ex02_T10.java。
2. 第②題：要用 new 建立 Scanner 物件實體 sc，所以答案為(B)。
3. 第③題：使用 sc 物件的 next 方法取得輸入的字串，所以答案為(B)。
4. 第④題：不再使用 Scanner 物件時要用 close 方法關閉，所以答案為(B)。

 題目

十一. 根據下列程式請回答下列問題？(行號僅供參考)

```
01 int x = 5, y = 10;
02 int i = ++x * y--;
03 System.out.println(i);
04 int j = x-- + ++y;
05 System.out.println(j);
```

- ① 第 03 行的輸出結果為何？ (A) 15 (B) 45 (C) 50 (D) 60
- ② 第 05 行的輸出結果為何？ (A) 13 (B) 14 (C) 15 (D) 16

說明

++為遞增運算子，--為遞減運算子。運算子若放在變數前面時就稱為前置式，此時會先遞增或遞減才做運算。若運算子放在變數後面時就稱為後置式，此時會先運算才做遞增或遞減。程式執行時各變數的變數值如下表所示，所以第①題答案為(D)，第②題答案為(D)，程式碼請參考 Ex02_T11.java。

行號	x 變數值	y 變數值	i 變數值	j 變數值
01	5	10	-	-
02	6(++x 前置遞增)	10	60(6*10)	-
03	6	9(y--後置遞減)	60	-
04	6	10(++y 前置遞增)	60	16(6+10)
05	5(x--後置遞減)	10	60	16



說明

- 第 8~18 行：為巢狀結構，外層用 if...else 選擇結構來判斷 score 是否介於 0~100，正確時執行 9~15 行敘述其中有內層的 switch 選擇結構；不正確時執行第 17 行敘述。
- 第 10~14 行：為 switch 選擇結構根據(score / 10)運算式的值，分別執行不同的 case 值敘述區段。因為採用新語法，所以允許多個 case 值共用一個敘述區段。
- 第 9~14 行：此段敘述使用賦值方式可以改寫如下：

```
int prize = switch (score / 10) {
    case 9, 10 -> 500;
    case 8 -> 200;
    default -> 0;
};
```

3.4 認證實例演練



題目

- 某俱樂部加入的資格如下，請使用下列選項(可不用或多次使用)來完成 Java 程式。

- 年齡若大於等於 65 歲至少需捐助 10000 元
- 年齡若大於等於 21 歲至少需捐助 25000 元

(A) && (B) || (C) < (D) <= (E) > (F) >= (G) == (H) !=

```
if(ages ____①____ 65 ____②____ money ____③____ 10000 ____④____ )
    ages ____⑤____ 21 ____⑥____ money ____⑦____ 25000 ){
    System.out.println("資格符合");
}
else{
    System.out.println("資格不符合");
}
```

說明

根據題目 if 敘述寫法為：if (ages >= 65 && money >= 10000 || ages >= 21 && money >= 25000)，所以答案①(F)、②(A)、③(F)、④(B)、⑤(F)、⑥(A)、⑦(F)，程式碼請參考 Ex03_T01.java。

- ① 填入的敘述為：(A) if (ages >= 65) (B) if (ages > 65) (C) if (ages != 65)
(D) if (ages > 65 || ages != 20)
- ② 填入的敘述為：(A) if (ages > 19) (B) if (ages < 20)
(C) else if (ages >= 19) (D) else if (ages >= 20)
- ③ 填入的敘述為：(A) else if (ages != 20) (B) if (ages <= 20) (C) else
(D) default

說明

根據 if 多重判斷式的語法答案為①(A)、②(D)、③(C)，程式碼請參考 Ex03_T06.java。

 **題目**

七. 根據下列 pay()方法，回答下列問題？

```
public static int pay(char grade) {  
    int price = 0;  
    switch (grade) {  
        case 'A':price = 500; break;  
        case 'B':price = 200;  
        case 'C':price = 50; break;  
        default: price = 1000; break;  
    }  
    return price;  
}
```

- ① 當 grade 為'A'傳回值為何？(A) 50 (B) 200 (C) 500 (D) 1000
- ② 當 grade 為'B'傳回值為何？(A) 50 (B) 200 (C) 500 (D) 1000
- ③ 當 grade 為'C'傳回值為何？(A) 50 (B) 200 (C) 500 (D) 1000
- ④ 當 grade 為'D'傳回值為何？(A) 50 (B) 200 (C) 500 (D) 1000

說明

- 第①題：當 grade 為'A'時，根據 switch 的語法有對應的 case 傳回值為 500，所以答案為 (C)，程式碼請參考 Ex03_T07.java。
- 第②題：當 grade 為'B'時有對應的 case，但是因為後面沒有 break 敘述，會繼續向下執行所以傳回值為 50，因此答案為 (A)。
- 第③題：當 grade 為'C'時，有對應的 case 傳回值為 50 所以答案為 (A)。
- 第④題：當 grade 為'D'時，因為沒有對應的 case，所以執行 default 傳回值為 1000，因此答案為 (D)。

由上面簡例中發現使用引數省略號非常簡便，但是使用省略號只能有一個而且必須置於最後面引數的限制。

```

✓ void add(int x, int... y){} //符合省略號只能一個和必須在最後的限制
✗ void add(int... x, int y){} //省略號在第一個引數，所以錯誤
✗ void add(int... x, int... y){} //省略號有兩個，所以錯誤

```

當有多個多載方法執行時，會以引數個數和資料型別都相同的方法優先執行，有明確個數的方法也比使用省略號方法優先。



實作

撰寫兩個名稱皆為 `max` 的靜態方法(方法多載)，第一個 `max()` 方法用來取得兩個整數中的最大值，第二個 `max()` 方法用來取得 `double` 陣列中的最大值。



結果

26 和 37 最大的數值為 37
陣列元素 [2.1, 5.3, 7.2, 4.8] 中最大的數值為 7.2



程式碼

檔名：\ex06\src\ex06\OverLoading.java

```

01 package ex06;
02 public class OverLoading {
03     static int max(int x, int y) {
04         if (x > y)
05             return x;
06         else
07             return y;
08     }
09
10     static double max(double[] vArray) {
11         double n = vArray[0];
12         for (int i = 1; i < vArray.length ; i++) {
13             if (vArray[i] > n)
14                 n = vArray[i];
15         }
16         return n;

```

`i < vArray.length - 1` 改成
`i < vArray.length`

- ① 要建立一個物件實體 millerS，使用下列哪一敘述？
- (A) student millerS = new student();
 - (B) student millerS = new Student();
 - (C) Student millerS = new student();
 - (D) Student millerS = new Student();
- ② 資料成員用什麼存取修飾子，而能從 main 主程式透過他們的 getter 和 setter 方法來存取？
- (A) private (B) protected (C) public
- ③ 在 main 主程式中，要如何透過 millerS 物件來呼叫一個無引數的 display 方法(函式)？
- (A) display(Student.millerS);
 - (B) display(millerS);
 - (C) millerS.display();
 - (D) millerS -> display();

說明

1. 答案：① (D)，② (A)，③ (C)。

 題目

六. Book 類別包含下列資料成員，及公有的 getter 和 setter 方法成員。

- private String title;
- private String author;
- private int year;
- private int isbn;

回答下列敘述，正確請選「是」，反之選「否」。

- ① 若方法成員只有設定資料成員的內容，則能傳出這些內容？
- (A) 是 (B) 否
- ② Book 類別的成員，只能在 main 主程式被 Book 類別實體化物件呼叫？
- (A) 是 (B) 否
- ③ 如果有一個沒有傳入引數的建構式，則只能有一個多載的建構式。
- (A) 是 (B) 否

- ① Account 類別含有一個建構式？ (A) 正確 (B) 錯誤
② 其他別的類別可以繼承 Account 類別？ (A) 正確 (B) 錯誤
③ 第 07 行的敘述也可以寫成 `this.balance = amount;` (A) 正確 (B) 錯誤

說明

1. 答案：① (B)，② (A)，③ (A)。

 題目

- 二. 你已經有一個 `InsurancePolicy` 類別。你需要定義一個常數資料成員 `Rate`，該常數是可被任何類別直接呼叫而不需建立實體物件。

請由選取下列的代碼來完成程式需求。(行號僅供參考)

- (A) final (B) finally (C) private (D) protected (E) public
(F) static (G) super (H) void

```
01 public class InsurancePolicy
02 {
03     _____ ① _____ ② _____ ③ double RATE = .0642;
04 }
```

說明

1. 答案：① (E)，② (F)，③ (A)。
2. 第 03 行：`public static final` 用來修飾成員變數和成員方法，為 "全域性常數"。對於變數，表示一旦給值就不可修改，並且透過類別名稱就可以呼叫。

 題目

- 三. 你已定義有一個 `Account` 類別，其建構式含有一個字串引數。你需要建立一個 `SavingsAccount` 類別來繼承 `Account` 類別。這 `SavingsAccount` 類別的建構式必須執行下列任務：

- 建構式含有一個 `name` 引數。
- 傳遞一個字串 "SavingsAccount" 至 `Account` 類別的建構式。
- 由建構式的 `name` 引數所傳入的值來設定類別 `name` 成員的變數值。

請由選取下列的代碼來完成程式需求。(行號僅供參考)

- (A) Account (B) extends (C) implements (D) super (E) this

5. 第 34 行：產生 `MyException` 例外類別物件實體，並將例外物件拋出，返回原呼叫點，向下執行第 53 行對應的 `catch` 作輸出。
6. 第 54 行：呼叫 `MyException` 類別的 `toString()` 方法。
7. 第 55 行：呼叫 `MyException` 類別的 `getMessage()` 方法。
8. 第 56 行：呼叫 `MyException` 類別的 `getLocalizedMessage()` 方法。
9. 第 57 行：呼叫第 17~19 行自定的 `showError()` 方法。

10.6 認證實例演練

題目

- 一. 請檢視下面 Java 程式執行時的錯誤訊息。

```
Exception in thread "main" java.lang.ArrayIndexOutOfBoundsException: Index 3
out of bounds for length 3
at exam.test5.func1(test5.java:8)
at exam.test5.run(test5.java:17)
at exam.test5.main(test5.java:30)
```

對於下列敘述，請選取正確或錯誤。

- ① 是 `run()` 方法內的敘述產生錯誤 (A) 正確 (B) 錯誤
- ② 是第 8 行敘述產生錯誤 (A) 正確 (B) 錯誤
- ③ 在產生錯誤之前，呼叫了 3 個方法 (A) 正確 (B) 錯誤
- ④ 這個錯誤是屬於語法錯誤 (A) 正確 (B) 錯誤

說明

1. 答案：① (B)、② (A)、③ (A)、④ (B)。
2. 第①題，程式執行流程是 `main()` 呼叫 `run()`，`run()` 呼叫 `func1()`，在 `func1()` 方法中發生錯誤，所以是 `func1()` 方法內的第 8 行敘述產生錯誤。
3. 第③題，在發生錯誤之前呼叫了 `main()`、`run()` 及 `func1()` 兩個方法。
4. 第④題，`ArrayIndexOutOfBoundsException` 是存取陣列時，超出陣列元素的註標範圍，是屬於執行階段的錯誤。

 題目

- 六. 試開發一個能夠讀寫檔案的應用程式。此程式必須符合以下條件：
1. 如果因為檔案處理錯誤而出現例外狀況，此例外狀況的細節必須顯示出來。
 2. 如果有任何其他例外狀況出現，則顯示出堆疊資料。
- 請問要如何完成此程式碼？請在作答區中選擇正確的敘述片段。

```
try
{
    /* 檔案處理程式區塊 */
}
catch ( _____ ① ex1) {
    System.out.println(ex1 _____ ② );
}
catch ( _____ ③ ex2) {
    System.out.println(ex2 _____ ④ );
}
```

- ① 填入的敘述為：
(A) IllegalArgumentException (B) IOException (C) RuntimeException
- ② 填入的敘述為：
(A) getMessage() (B) getCause() (C) printStackTrace()
- ③ 填入的敘述為：(A) Error (B) Exception
(C) IllegalArgumentException (D) RuntimeException
- ④ 填入的敘述為：
(A) fillInStackTrace() (B) getCause () (C) getStackTrace()

說明

1. 第①題：答案為(B)。
2. 第②題：答案為(C)。
3. 第③題：答案為(B)。
4. 第④題：答案為(C)。

		<p>A. ① <= ② >= ③ != ④ ==</p> <p>B. ① ② && ③ != ④ &=</p> <p>C. ① <= ② >= ③ > ④ <</p> <p>if(y ___ A ___ x ___ B ___ z ___ C ___ y)</p>
20.	<p>A: ①</p> <p>B: ②</p>	<p>請審視下列 Java 程式碼片段。</p> <pre>displayMenu(); for(;;) { boolean b = isOK(); if(!b) break; displayMenu(); }</pre> <p>在保留原有功能的前題下，您將重構此下列程式碼片段。請由下列選項中選取正確程式碼。</p> <p>程式碼片段：</p> <p>A. ① do ② for ③ while (isOK()) ④ while</p> <p>B. ① } do; ② } while (isOK()); ③ } ④ } continur;</p> <p>___ A ___</p> <pre>{ displayMenu(); ___ B ___</pre>
21.	A: ③	<p>您必須在 intList 二維陣列中列印每一個元素。</p> <p>請從以下選取正確的選項以完成程式碼。</p> <p>A. ① column < intList.length; column++)</p> <p>② column < intList[column].length; column++)</p> <p>③ column < intList[row].length; column++)</p> <p>④ column < intList[row][column].length; column++)</p> <pre>for (int row = 0; row < intList.length; row++) for(int column = 0; ___ A ___ System.out.println(intList[row][column]);</pre>

		<pre> <u> A </u> <u> B </u> isInGroup <u> C </u> { for(int i = 0; i < numMember; i++) { if (members[i].equals member) <u> D </u> } <u> E </u> } </pre>
34.	<p>A: ③ B: ② C: ② D: ①</p>	<p>若要開發可讀取及寫入檔案的應用程式。 此應用程式必須執行下列工作：</p> <ul style="list-style-type: none"> ● 顯示因檔案處理錯誤而引起的例外狀況。 ● 顯示發生任何其他例外狀況時的堆疊資訊。 <p>請由下列選項中選取正確的程式碼片段以完成程式碼。 程式碼片段：</p> <p>A. ① ArrayIndexOutOfBoundsException ② RuntimeException ③ IOException ④ IllegalArgumentException B. ① printMessage() ② getMessage() ③ printStackTrace() C. ① Error ② Exception ③ RuntimeException ④ IllegalArgumentException ⑤ ArrayIndexOutOfBoundsException D. ① getStackTrace() ② printStackTrace() ③ fillStackTrace() ④ errorStackTrace()</p> <pre> try { /* add procedure */ } catch (<u> A </u> e1) { System.out.println(e1.<u> B </u>); } catch (<u> C </u> e2) { System.out.println(e2.<u> D </u>); } </pre>
35.	<p>A: ② B: ③ C: ①</p>	<p>請選取正確的選項，以完成有關在 Java 中使用控制流程的敘述。</p> <p>A. 若要從巢狀迴圈的內部迴圈跳離，則需使用何種敘述？ ① continue ② 已標記 break ③ 未標記 break ④ return</p> <p>B. 若要終止巢狀迴圈的外部迴圈，則需使用何種敘述？ ① continue ② 已標記 break ③ 未標記 break ④ return</p>

		<pre>class Game { ___A___ ___B___ void playNumber(String gameNumber) { } }</pre>
40.	<p>A: ② B: ① C: ④ D: ④</p>	<p>Book 類別包含下列私有資料成員：</p> <ul style="list-style-type: none"> ● String title; ● String author; ● int year; ● int isbn; <p>建構式預設會依此指派空字串及零，而多載的建構式則會以上列順序接受引數。</p> <p>您需要完成程式碼，將名稱為 book 的 Book 類別物件實體化，以便使用者可以在之後的函式進行更新。</p> <p>請選取正確的以完成程式碼。</p> <p>程式碼片段：</p> <p>A. ① book ② Book ③ int ④ new B. ① book ② Book ③ int ④ new C. ① book ② Book ③ int ④ new D. ① book() ② book(title, author, year, isbn) ③ Book() ④ Book(title, author, year, isbn)</p> <pre>public static void main(String args[]){ ___A___ ___B___ = ___C___ ___D___ ; }</pre>