

MTA 98-381 考綱說明

MTA 98-381 Introduction to Programming Using Python

一、使用資料類型和運算子執行操作 (20-25%)

對一個運算式進行評估，以辨識 Python 會指派給每個變數的資料類型

- 辨識 STR、INT、浮點及布林資料類型

執行資料和資料類型操作

- 將一種資料類型轉換為另一種類型；建構資料結構；執行索引和切割操作

確定基於運算子優先順序的執行序列

- 指派；比較；邏輯；運算；辨識 (IS)；內含項目 (IN)

選擇適當的運算子以獲取預期的結果

- 指派；比較；邏輯；運算；辨識 (IS)；內含項目 (IN)

二、控制帶有決策和迴圈的流程 (25-30%)

建構和分析使用分支陳述式的程式碼區段

- IF；ELIF；ELSE；巢狀及複合條件運算式

建構和分析執行反覆運算的程式碼區段

- WHILE；FOR；中斷；繼續；通過；巢狀迴圈和包含複合條件運算式的迴圈

三、執行輸入和輸出操作 (20-25%)

建構並分析執行檔案輸入和輸出操作的程式碼區段

- 陳述式的打開；關閉；讀取；寫入；附加；檢查是否存在以及刪除

建構並分析執行主控臺輸入和輸出操作的程式碼區段

- 從主控臺讀取輸入；列印格式化文字；使用指令行引數

四、文件和結構代碼 (15-20%)

使用註解和文件字串的文件程式碼區段

- 使用縮排、空格、註解和文件字串；使用 Pydoc 產生文件

建構和分析包含功能定義的程式碼區段

- 呼叫簽章；預設值；返回；DEF；通過

五、執行疑難排解和錯誤處理 (5-10%)

分析、偵測及修復出現錯誤的程式碼區段

- 語法錯誤、邏輯錯誤、運行時間錯誤
- 分析與建構處理例外狀況的程式碼區段
- 嘗試；排除；其他；最終；改善

六、使用模組和工具執行操作 (1-5%)

使用內建模組執行基本操作

- 數學；日期時間；IO；SYS；OS 路徑；隨機
- 透過使用內建模組解決複雜運算問題
- 數學；日期時間；隨機

一、使用資料類型和 運算子執行操作

對一個運算式進行評估，以辨識 Python 會指派給每個變數的資料類型

- 辨識 STR、INT、浮點及布林資料類型

執行資料和資料類型操作

- 將一種資料類型轉換為另一種類型；建構資料結構；執行索引和切割操作

確定基於運算子優先順序的執行序列

- 指派；比較；邏輯；運算；辨識 (IS)；內含項目 (IN)

選擇適當的運算子以獲取預期的結果

- 指派；比較；邏輯；運算；辨識 (IS)；內含項目 (IN)

你編寫了以下的程式碼：

```
list_1 = [1, 2, 3]
list_2 = [4, 5, 6]
list_3 = list_1 + list_2
list_4 = list_3 * 2
print(list_4)
```

執行程式碼的輸出值是？

- A. `[[1, 2, 3], [4, 5, 6], [1, 2, 3], [4, 5, 6]]`
- B. `[4, 10, 18]`
- C. `[1, 2, 3, 4, 5, 6, 1, 2, 3, 4, 5, 6]`
- D. `[[1, 2, 3, 4, 5, 6], [1, 2, 3, 4, 5, 6]]`

你是運動App的程式設計師。你必須製作一個函式為跑者計算步速，所謂步速就是每公里所花的時間就是步速。輸出結果必須盡可能精準。要如何完成程式碼？

回答時請在回答區中選擇適當的程式碼片段。其中距離轉換為浮點數，分秒的輸入值都要轉換為整數。

步速計算器

```
distance = ___(1)___ (input("請輸入以公尺為單位的行駛距離"))
distance_kms = distance / 1000 # 轉換為公里
time_minute = ___(2)___ (input("請輸入經過分鐘"))
time_sec = ___(3)___ (input("請輸入經過秒數"))
time = time_minute*60 + time_sec
pace = time / distance_kms
print("步速是： " , str((pace//60)) + ":" + str((pace%60)))
```

以上空格分別要填入的函式名為：

- A. int B. string C. float
- A. int B. string C. float
- A. int B. string C. float

高年級的老師要製作一份報表來顯示這次考試班上所有學生的平均分數。報表必須去除平均分數的小數部分。

你應該使用哪兩個程式碼片段？每個正確的答案都提供了一個完整的解決方案。請選擇兩個。

- A. 平均分數 = `float(全班總分 // 全班人數)`
- B. 平均分數 = `int(全班總分 / 全班人數)`
- C. 平均分數 = `float(全班總分 ** 全班人數)`
- D. 平均分數 = `全班總分 // 全班人數`

你正在編寫一個Python程式用來記錄客戶資料並將其存儲在資料庫中。
這個程式處理各種各樣的資料。

以下的變數宣告後它們的資料類別是？

回答時，請將適合的程式碼片段連到正確的回答區。

程式碼片段

`int`

`bool`

`str`

`float`

回答區

<input type="checkbox"/>	<code>age = 12</code>	<code>int</code>
<input type="checkbox"/>	<code>minor = False</code>	<code>bool</code>
<input type="checkbox"/>	<code>name = "David"</code>	<code>str</code>
<input type="checkbox"/>	<code>weight = 64.5</code>	<code>float</code>
<input type="checkbox"/>	<code>zip = "545"</code>	<code>str</code>

你正在編寫一個計算使用者出生西元年轉民國年的程式。該程式詢問使用者的出生的西元年份，然後輸出使用者的出生的民國年份。你編寫以下程式碼，其中包含的行號只是做為參考。

```
01 year = input("請輸入你出生的西元年份：")
02
03 born = eval(year) - 1911
04 message = "你出生在民國 " + str(born) + " 年"
05 print(message)
```

請問下列何者是正確的？

- A. 在01行中year的資料類型是 str?
- B. 在03行中born的資料類型是 float?
- C. 在04行中message的資料類型是 bool?

在Python資料類型的課程中，你創建以下三個程式碼片段：

```
# 程式碼片段 1
```

```
x1 = "5"
```

```
y1 = 4
```

```
a = x1 * y1
```

```
# 程式碼片段2
```

```
x2 = 10
```

```
y2 = 4
```

```
b = x2 / y2
```

```
# 程式碼片段3
```

```
x3 = 5.5
```

```
y3 = 1
```

```
c = x3 / y3
```

你需要評估程式碼片段。請問下列何者是正確的？(可多選)

- A. 執行程式碼片段1後，變數 a 的資料類型為 str。
- B. 執行程式碼片段2後，變數 b 的資料類型是 float。
- C. 執行程式碼片段3後，變數 c 的資料類型為 int。

在Python程中我們利用`type()`查詢每個值的資料類別，以下的程式執行後出現的資料類別分別是：

```
type(+1E10)
```

```
type(5.0)
```

```
type("True")
```

```
type(False)
```

- A. int, int, bool, bool
- B. float, float, str, bool
- C. int, float, str, bool
- D. float, int, str, str

你設計了以下程式用座號來查詢學生的姓名。其中包含的行號只是做為參考。

```
01 students = {1: 'John', 2: 'Mary'}
02 id = input('輸入學生座號: ')
03 if not id in students:
04     print('該學生並不存在.')
05 else:
06     print("學生姓名為: " + students[id])
```

同事們報告說程式有時會產生不正確的結果。

你需要對程式進行故障排除。根據程式碼片段中提供的資訊，請選出正確答案。

(1) 在01行中有哪兩種資料類型存儲在students清單中？

- A. bool 與 string
- B. float 與 bool
- C. int 與 string
- D. float 與 int

(2) 在02行中id的資料類型是什麼？

- A. bool
- B. float
- C. int
- D. string

(3) 在03行中為什麼會在students清單中找不到資料？

- A. 語法不正確
- B. 資料類型不匹配
- C. 變數命名錯誤

你設計了一個函式，它會檢查傳入的值的資料類型再返回。程式碼如下：

```
01 def checkType(value):  
02     dataType = type(value)  
03     return dataType  
04 print(checkType(1))  
05 print(checkType(1.0))  
06 print(checkType("True"))  
07 print(checkType(True))
```

根據程式碼片段中提供的資訊，請選擇每個問題的答案選項。

(1) 在04行會印出什麼結果?

- A.<class'bool'>
- B.<class'float'>
- C.<class'int'>
- D.<class'str'>

(2) 在05行會印出什麼結果?

- A.<class'bool'>
- B.<class'float'>
- C.<class'int'>
- D.<class'str'>

(3) 在06行會印出什麼結果?

- A.<class'bool'>
- B.<class'float'>
- C.<class'int'>
- D.<class'str'>

(4) 在07行會印出什麼結果?

- A.<class'bool'>
- B.<class'float'>
- C.<class'int'>
- D.<class'str'>

你加入了一個電子商務公司成為程式開發部門的實習生。你的程式中有一個地方要讓使用者提供一個數值。即使使用者輸入了小數，該值也必須轉換為整數來進行計算。

你應該使用哪個程式碼片段？

- A. `totalNums = input("總共有幾筆資料？")`
- B. `totalNums = int(input("總共有幾筆資料？"))`
- C. `totalNums = str(input("總共有幾筆資料？"))`
- D. `totalNums = float(input("總共有幾筆資料？"))`

同事開發一個將產品名稱輸入到資料庫的程式，但是其中發生了錯誤，讓每個存入的名稱都顛倒了字母順序。請你開發一個Python函式，將每個產品名稱以正確的順序輸出。

請選擇適當的程式碼片段來完成程式：

```
# 函式會反轉字串中的字元。  
# 以相反的順序返回新字串。  
def reverse_pname(backwards_pname):  
    forward_pname = ""  
    for index in ____(1)__:  
        forward_pname += ____(2)__  
    return forward_pname  
print(reverse_pname("klim")) # 測試範例
```

(1)

- A. backwards_pname
- B. len(backwards_pname)
- C. range(0, len(backwards_pname), -1)
- D. range(len(backwards_pname)-1, -1, -1)

(2)

- A. backwards_name[index-1]
- B. backwards_name[len(forward_name)-1]
- C. backwards_name[len(backward_name)-len(forward_name)]
- D. backwards_name[index]

你設計了一個 Python 程式用來顯示每個員工每天工作到現在的小時數。

你需要計算工作時數並顯示訊息。程式碼如下：

```
01 start = input("你今天幾點開始工作?")
```

```
02 end = input("現在幾點?")
```

```
03
```

如果要完成這個程式，在03行使用哪個程式碼？

- A. `print("你已經工作了 " + str(int(end) - int(start)) + " 小時!")`
- B. `print("你已經工作了 " + (int(end) - int(start)) + " 小時!")`
- C. `print("你已經工作了 " + str(end - start) + " 小時!")`
- D. `print("你已經工作了 " + int(end - start) + " 小時!")`

你有以下清單結構：

```
alph = "abcdefghijklmnopqrstuvwxyz"
```

以下各個程式片段的結果各是如何？

(1) `alph[3:15]`

(2) `alph[3:15:3]`

(3) `alph[15:3:-3]`

(4) `alph[::-3]`

C F B A

A. `zwtqnkheb`

B. `pmjg`

C. `defghijklmno`

D. `ponmlkjihgfe`

E. `defghijklmnop`

F. `dgjm`

G. `Olif`

你為學校設計了一個Python應用程式，在classroom的清單中包含了60位同學的姓名，最後3名是班上的幹部。

你需要分割清單內容顯示除了幹部以外的所有同學，你可以利用以下哪二個程式碼達成？

- A. `classroom[0:-2]`
- B. `classroom[0:-3]`
- C. `classroom[1:-3]`
- D. `classroom[:-3]`
- E. `classroom[1:-3]`

你開發一個Python應用程式，其中有一個名為month的清單儲存所有的月份的英文。

你要分割這個清單，取得由第二個月份開始，每間隔一個值的月份名稱，你應該使用哪個程式碼？

- A. `month[2:2]`
- B. `month[::2]`
- C. `month[1::2]`
- D. `month[1:2]`

你正在編寫Python程式用於計算一個數學公式。

公式內容為 b 等於 a 加上 5，然後再平方，其中 a 是輸入的值， b 是結果。

你設計了以下的程式碼片段。

```
01 a = eval(input("輸入一個數字: "))
```

```
02 b =
```

如何完成 02 行的程式碼？

- A. $b = (a + 5)**2$
- B. $b = a + 5 **2$
- C. $b = a + 5 * 2$
- D. $b = a + (5 **2)$

你正在開發一個補習班的Python的函式來計算折扣，補習班希望鼓勵小朋友和老年人報名，只要是小朋友和老年人報名相關課程就會獲得**10%**折扣。

你編寫以下程式碼。

```
01 def get_discount(kid, senior):  
02     discount = 0.1  
03  
04     discount = 0  
05     return discount
```

你必須完程這個程式碼。

在03行中你將加入什麼程式碼？

- A. `if not (kid or senior):`
- B. `if (not kid) or senior:`
- C. `if not (kid and senior):`
- D. `if (not kid) and senior:`

你正在開發一個比較數字的Python程式，下列何者的值是True？

- A. `0 or 5`
- B. `bool(0)`
- C. `None is None`
- D. `-5 < 0 < 5`

計算以下的 Python 數學運算式

$(3 * (1 + 2) ** 2 - 2 ** 2 * 3)$

結果為何？

- A. 3
- B. 13
- C. 15
- D. 69

你編寫了以下的程式碼：

```
a = 'Test1'  
print(a)  
  
b = 'Test2'  
a += b  
print(a)  
print(b)
```

根據程式碼片段中提供的資訊選擇每個問題的答案選項。

(1) 在第一次print後會顯示什麼？

- A. Test1
- B. Test1 Test2
- C. Test2

(2) 在第二次print後會顯示什麼？

- A. Test1
- B. Test1 Test2
- C. Test2

(3) 在第三次print後會顯示什麼？

- A. Test1
- B. Test1 Test2
- C. Test2

你為你的公司開發一個Python應用程式有以下的程式碼：

```
01 def test(a, b, c, d):  
02     value = (a + b) * c - d  
03     return value
```

根據程式碼片段中提供的資訊選擇每個問題的答案選項。

(1) 運算式的哪個部份將第一個進行計算？

- A. $a+b$
- B. $b*c$
- C. $c-d$

(2) 運算式的哪個部份將第二個進行計算？

- A. 加(+)
- B. 減(-)
- C. 乘(*)

(3) 哪個運算式等於題目函式中的運算式？

- A. $(a+b) * (c-d)$
- B. $(a + (b*c)) - d$
- C. $((a + b) * c) - d$

請按先後順序從頭至尾排列這六類運算的正確順序：

加法和減法 (+, -)

乘法和除法 (*, /)

正數(+)、負數(-)與反位元(not)

括弧

指數 (**)

且(And)

- A. 加法和減法->乘法和除法->正數、負數與反位元->括弧->指數->且
- B. 括弧->指數->正數、負數與反位元->乘法和除法->加法和減法->且
- C. 指數->乘法和除法->正數、負數與反位元->括弧->且->加法和減法
- D. 乘法和除法->括弧->正數、負數與反位元->指數->且->加法和減法

出版社想要檢查出版物中特定字母的數量，你需要設計一個函式以符合要求。你要如何完成這段程式碼？

回答區

```
# 函式接受檔案中的文字清單，
# 以及要搜索的字母。
# 返回該清單中特定字母的數量。

def count_letter(letter, letter_list):
    count = 0
    for ___(1)___
        if ___(2)___
            count += 1
    return count

letter_list = []
# letter_list 是由檔案中取得。程式碼中並未顯示。
letter = input("你想要計算哪個字母的數量?")
letter_count = count_letter(letter, letter_list)
print("一共有：", letter_count, " 個字母 " + letter)
```

(1)

- A. letter_list in word:
- B. word in letter_list:
- C. word == letter_list:
- D. word is letter_list:

(2)

- A. word is letter:
- B. letter is word:
- C. word in letter:
- D. letter in word:

租車公司需要一種方法來決定客戶租用車輛的費用。該費用取決於車輛歸還的時間。然而，週四和周日也有特別的費率。

費用結構如下所示：

- 費用是每天**100**美元。
- 如果車輛在晚上**11**點後返還，則客戶將被收取額外一天的費用。
- 如果車輛是在星期天租的，那麼客戶即可享受**10%**的折扣。
- 如果車輛是在星期四租的，那麼客戶即可以享受**20%**的折扣。

你需要撰寫程式碼去符合這個需求。

你要如何完成這段程式碼？

回答區

```
# 車輛出租計算機
```

```
ontime = input("車子是在晚上11點前返還的嗎? y 或 n").lower()
```

```
days_rented = int(input("車子出租了幾天?"))
```

```
day_rented = input("車子是在星期幾出租?").capitalize()
```

```
cost_per_day = 100
```

```
if ontime ____(1)____
```

```
    days_rented += 1
```

```
if day_rented ____(2)____
```

```
    total = (days_rented * cost_per_day) * 0.9
```

```
elif day_rented ____(3)____
```

```
    total = (days_rented * cost_per_day) * 0.8
```

```
else:
```

```
    total = days_rented * cost_per_day
```

```
print("車輛的租借費用為：$", total)
```

(1)

A. != "n": B. == "n": C. == "y":

(2)

A. == "Sunday ": B. >= "Sunday ": C. is "Sunday ":

(3)

A. == "Thursday": B. <= "Thursday" C. is "Thursday":

你設計了一個函式來執行除法，因為除法的除數不能為零，所以在函式中必須要針對這個重點進行檢查。你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

```
def safe_divide(numerator, denominator):  
    ____ (1) ____  
    print("你少填了被除數或除數")  
    ____ (2) ____  
    print("除數為零會產生錯誤")  
else:  
    return numerator / denominator
```

(1)

- A. if numerator is None or denominator is None:
- B. if numerator is None and denominator is None:
- C. if numerator = None or denominator = None:
- D. if numerator = None and denominator = None:

(2)

- A. elif denominator == 0:
- B. elif denominator = 0:
- C. elif denominator != 0:
- D. elif denominator in 0:

你設計了一個數學運算的Python程式，程式碼如下：

```
a = 11
```

```
b = 5
```

何者是下列每個數學運算式的結果？回答時，請由左方回答區的答案連到右方的結果區的答案。

回答區

1

2

2.2

結果

print(a / b)

print(a // b)

print(a % b)

2.2 , 2 , 1

老闆要求你對以下程式碼除錯：

```
x = 0
while x < 4:
    if x % 4 == 0:
        print("party")
    elif x - 2 < 0:
        print("cake")
    elif x / 3 == 0:
        print("greeting")
    else:
        print("birthday")
    x = x + 1
```

什麼會將會輸出列印到螢幕上？

A.

party
greeting
birthday
cake

B.

party
cake
birthday
birthday

C.

birthday
party
greeting
cake

D.

birthday
greeting
party
cake

在下列的程式碼中：

```
aList = [0, 1, 2, 3, 4]  
print(4 in aList)
```

會輸出列印的內容？

- A.4
- B.5
- C.True
- D.False

你為你的公司開發一個Python應用程式。

你設計了以下的程式碼：

```
aList = ["a", "b", "c", "d", "e"]
bList = [1, 2, 3, 4, 5]
print(aList is bList)
print(aList == bList)
aList = bList
print(aList is bList)
print(aList == bList)
```

根據程式碼片段中提供的資訊，使用下拉式功能表選擇每個問題的答案選項。

回答區

- | | | |
|----------------------|---|--|
| (1) 在第一次print後會顯示什麼? | <input type="radio"/> A.True | <input checked="" type="radio"/> B.False |
| (2) 在第二次print後會顯示什麼? | <input type="radio"/> A.True | <input checked="" type="radio"/> B.False |
| (3) 在第三次print後會顯示什麼? | <input checked="" type="radio"/> A.True | <input type="radio"/> B.False |
| (4) 在第四次print後會顯示什麼? | <input checked="" type="radio"/> A.True | <input type="radio"/> B.False |

你設計了一個比較數字的Python程式，內容如下：

```
01 n1 = eval(input("請輸入第一個數字: "))
02 n2 = eval(input("請輸入第二個數字: "))
03 if n1 = n2:
04     print("這二個數字是相等的.")
05 if n1 <= n2:
06     print("數字1 小於 數字2.")
07 if n1 > n2:
08     print("數字1 大於 數字2.")
09 if n2 <> n1:
10     print("這二個數字是不相同的.")
```

針對下列每個敘述，如果正確的就選擇Yes。否則請選擇 No。

- A. 在 03 行的語法是不正確的比較。 Yes No
- B. 在 06 行的語法只有 num1 小於 num2 時才會列印出來。 Yes No
- C. 在 08 行的語法只有 num1 大於 num2 時才會列印出來。 Yes No
- D. 在 09 行的語法是不正確的比較。 Yes No

二、控制帶有決策 和迴圈的流程

建構和分析使用分支陳述式的程式碼區段

- IF; ELIF; ELSE; 巢狀及復合條件運算式

建構和分析執行反覆運算的程式碼區段

- WHILE; FOR; 中斷; 繼續; 通過; 巢狀迴圈和包含復合條件運算式的迴圈

你設計了一個程式要依學生的成績來顯示等級，它的規定如下：

例如，如果使用者輸入 90，則輸出應該是，"你的成績為甲等" 相同的，如果使用者輸入 89，則輸出應該為 "你的成績為乙等"

你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

分數	等級
100 到 90	甲等
89 到 80	乙等
79 到 70	丙等
69 到 60	丁等
60以下	不及格

回答區

```
# 字母等級轉換器
```

```
grade = int(input("請輸入數字等級"))
```

```
____(1)____
```

```
    letter_grade = '甲等'
```

```
____(2)____
```

```
    letter_grade = '乙等'
```

```
____(3)____
```

```
    letter_grade = '丙等'
```

```
____(4)____
```

```
    letter_grade = '丁等'
```

```
else:
```

```
    letter_grade = '不及格'
```

```
print("你的成績為 :", letter_grade)
```

(1)

A.if grade <= 90:

B.if grade >= 90:

C.elif grade > 90:

D.elif grade >= 90:

(2)

A.if grade > 80:

B.if grade >= 80:

C.elif grade > 80:

D.elif grade >= 80:

(3)

A.if grade > 70:

B.if grade >= 70:

C.elif grade > 70:

D.elif grade >= 70:

(4)

A.if grade > 60:

B.if grade >= 60:

C.elif grade > 60:

D.elif grade >= 60:

你要設計一款以使用者年齡進行電影分級的程式。這個函式必須符合以下要求：

- 任何18歲或以上的人會顯示 "限制級" 的訊息。
- 任何13歲或以上，但小於18歲的人都會顯示 "輔導級"的訊息。
- 任何12歲或更年輕的人都會顯示 "普通級"的訊息。
- 如果年齡未知，則會顯示 "未知"的訊息。

你需要完成程式碼以符合要求。你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

回答區

```
def get_rating(age):  
    rating = ""  
    if ___(1)___  
    elif ___(2)___  
    elif ___(3)___  
    else ___(4)___  
    return rating
```

(1)

- A. age < 13: rating = "普通級"
- B. age < 18: rating = "輔導級"
- C. : rating = "限制級"
- D. age == None: rating = "未知"

(2)

- A. age < 13: rating = "普通級"
- B. age < 18: rating = "輔導級"
- C. : rating = "限制級"
- D. age == None: rating = "未知"

(3)

- A. age < 13: rating = "普通級"
- B. age < 18: rating = "輔導級"
- C. : rating = "限制級"
- D. age == None: rating = "未知"

(4)

- A. age < 13: rating = "普通級"
- B. age < 18: rating = "輔導級"
- C. : rating = "限制級"
- D. age == None: rating = "未知"

你正在編寫一個函式來判別負數與非負數。

這個函式必須符合以下要求：

- 如果 a 是負數，則回傳 "值是負數"
- 如果 a 不是負數，則為非負數，再繼續判別。
- 如果 a 大於 0 ，則回傳 "值是正數"，否則回傳 "值是零"

你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

回答區

```
def reResult(a):  
    ___(1)___  
    answer = "值是負數"  
    ___(2)___  
    ___(3)___  
    answer = "值是正數"  
    ___(4)___  
    answer = "值是零"  
    return answer
```

(1)

- A. `if a < 0:`
- B. `if a > 0:`
- C. `else:`
- D. `elif:`

(2)

- A. `if a < 0:`
- B. `if a > 0:`
- C. `else:`
- D. `elif:`

(3)

- A. `if a < 0:`
- B. `if a > 0:`
- C. `else:`
- D. `elif:`

(4)

- A. `if a < 0:`
- B. `if a > 0:`
- C. `else:`
- D. `elif:`

你設計了一個電影票收費的函式，票價的規則如下：

- 5歲以下= 免費入場
- 5歲及以上的學生 = 60 元
- 5歲到17歲但不是學生 = 120元
- 17歲以上但不是學生 = 180元

你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

```
def ticket_fee(age, school):  
    fee = 0  
    ___(1)___  
    fee = 60  
    ___(2)___  
    ___(3)___  
    fee = 120  
    else:  
        fee = 180  
    return fee
```

(1)

- A. if age >= 5 and school == True:
- B. if age >= 5 and school == False:
- C. if age <= 17

(2)

- A. if age >= 5 and school == True:
- B. if age >= 5 and school == False:
- C. if age <= 17

(3)

- A. if age >= 5 and school == True:
- B. if age >= 5 and school == False:
- C. if age <= 17

你要設計一個Python程式來檢查使用者輸入的數字是是1位數，2位數還是2位數以上，其中規定輸入的值必須是正整數。

你要如何完成這段程式碼？

```
num = int(input("請輸入一個正整數: "))
digits = "0"
if num > 0:
    ___(1)___
    digits = "1"
    ___(2)___
    digits = "2"
    ___(3)___
    digits = ">2"
    print(digits + " 位數.")
else:
    print("輸入值並不是正整數")
```

(1)

- A. `if num < 10:`
- B. `if num < 10`
- C. `if num > -10 and num < 10:`
- D. `if num > -100 and num < 100:`

(2)

- A. `if num < 100:`
- B. `if num < 100`
- C. `elif num < 100:`
- D. `elif num > -100 and num < 100:`

(3)

- A. `elif num > 100`
- B. `if:`
- C. `else:`
- D. `elif:`

你在設計一個Python程式遊戲讓參加的人猜測1到100之間的數字，最多有三次機會。程式碼如下：

```
01 from random import randint
02 target = randint(1, 100)
03 chance = 1
04 print("猜一個從1到100的整數。你將有3機會。")
05
06     guess = int(input("請輸入一個整數: "))
07     if guess > target:
08         print("數字太大了，猜低一點")
09     elif guess < target:
10         print("數字太小了，猜高一點")
11     else:
12         print("猜對了!")
13
14
```

程式可以讓使用者猜三次，如果猜出正確號碼即停止程式。

你要如何完成行號05、13 及 14 的程式碼？回答時，請將適合的程式碼片段連到正確的回答區。

回答區

- A. 在 05 行你要使用哪個程式碼片段？ (1)
- B. 在 13 行你要使用哪個程式碼片段？ (3)
- C. 在 14 行你要使用哪個程式碼片段？ (5)

程式碼片段

- 1. while chance <= 3:
- 2. while chance < 3:
- 3. break
- 4. pass
- 5. chance += 1
- 6. while chance < 3
- 7. chance = 2

在以下的程式碼中，要加入哪些程式碼片段讓 `print` 語法可以正確執行？你要如何完成程式碼讓 `print` 語法是正確的？回答時，請在回答區中選擇適當的程式碼片段。

```
aList = [1, 2, 3]
bList = ["a", "b", "c"]
____(1)____
    print("aList 中的值等於 bList")
____(2)____
    print("aList 中的值不等於 bList")
```

(1)

- A. `if aList == bList :`
- B. `if aList == bList`
- C. `else :`
- D. `else`

(2)

- A. `if aList == bList :`
- B. `if aList == bList`
- C. `else :`
- D. `else`

你設計了一個Python程式來檢查輸入英文姓名的大小寫，你可以使用哪四個程式碼進行順序排列。

A.

```
name = input("輸入你的英文姓名: ")
```

B.

```
elif name.lower() == name:  
    print(name, "是全部小寫.")
```

C.

```
else:  
    print(name, "是大寫.")
```

D.

else:

```
    print(name, "是大小寫混合.")
```

E.

```
if name.upper() == name:  
    print(name, "是全部大寫.")
```

F.

```
else:  
    print(name, "是小寫.")
```

A、E、B、D

公司決定要幫所有年薪不到 50 萬的員工調升基本工資5%，並給予獎金1萬元，以下是計算公式：

新工資 = 目前工資 × 105% + \$10000 獎金。

程式會將每個人調整後的薪資料存入 salarylist 清單中。

你要如何完成這段程式碼？回答時，選擇右方適當的程式碼片段。

回答區

```
# 清單中的每個人的工資都根據增加而更新。  
# 年薪50萬元以上的員工將不會得到加薪。  
# salarylist 是由員工資料庫中取得，程式碼不會顯示。
```

____(1)____

```
if salaryList[index] >= 150000:
```

____(2)____

```
salaryList[index] = (salaryList[index] * 1.05) + 10000
```

(1)

- A. for index in range(len(salary_list)+1):
- B. for index in range(len(salary_list)-1):
- C. for index in range(len(salary_list)):
- D. for index in salary_list:

(2)

- A. exit()
- B. continue
- C. break
- D. end

你設計了一個函式計算並顯示從2到9的所有乘法組合的九九乘乘法表。
你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

```
# 九九乘法表函式
def times_tables():
    ____ (1) ____
        ____ (2) ____
            print(row * col, end = " ")
        print()
# 呼叫九九乘法表
times_tables()
```

(1)

- A. for col in range(9):
- B. for col in range(2,10):
- C. for col in range(2,9,1):
- D. for col in range(10):

(2)

- A. for row in range(9):
- B. for row in range(2,9,1):
- C. for row in range(2,10):
- D. for row in range(10):

你設計了一個Python程式來顯示 2 到 100 中的所有質數，請將左方的程式碼片段排列到右方的回答區的區塊正確位置中。

A.
n = 2
is_prime = True
while n <= 100:
B.
n = 2
while n <= 100:
 is_prime = True
C.
break
D.
continue

E.
n += 1
F.
for i in range(2, n):
 if n / i == 0:
 is_prime = False
G.
for i in range(2, n):
 if n % i == 0:
 is_prime = False

B C G E

```
n = 2
while n <= 100:
    is_prime = True
    for i in range(2, n):
        if n % i == 0:
            is_prime = False
            break
    if is_prime == True:
        print(n)
    n += 1
```

你用Python設計了一個比大小函式，必須符合以下的需求：

- 這個函式有二個參數：一個整數清單和一個整數作為參數。
- 該函式必須在清單中以整數進行比較。
- 如果在清單中整數字參數的整數大，則該函式必須列印一則消息，指示已找到的整數比參數整數大，然後停止查看清單的動作。
- 如果找不到比參數整數大的數，則該函式必須列印一則消息，指示在清單中找不到比參數整數大的數。

你應該如何安排這些程式碼片段的順序來開發解決方案？回答時，請將適合的程式碼片段從程式碼片段清單移動到回答區，並按正確的順序排列。

A.

```
for i in range(len(nums)):
```

B.

```
if nums[i] > num:
```

```
    print("{0}清單中找到比{1}的大值".format(nums, num))
```

C.

```
else:
```

```
    print("{0}清單中找不到比{1}的大值".format(nums, num))
```

D.

```
    break
```

E.

```
def search(nums, num):
```

E、A、B、C

你正在設計一個Python程式去驗證產品編號。

產品編號的格式必須為 `dd-dddd`，並且只包含數字和破折號。如果格式正確，則程式必須列印 `True`，如果格式不正確，則列印 `False`。

你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

```
____(1)____  
parts = ""  
____(2)____  
____(3)____  
product_no = input("輸入產品編號 (dd-dddd): ")  
parts = product_no.split('-')  
if len(parts) == 2:  
    if len(parts[0]) == 2 and len(parts[1]) == 4:  
        if parts[0].isdigit() and parts[1].isdigit():  
            ____ (4) ____  
print(valid)
```

(1)

- A. `product_no = ""`
- B. `product_no = "sentinel"`

(2)

- A. `while product_no != "":`
- B. `while product_no != "sentinel":`

(3)

- A. `valid = False`
- B. `valid = True`

(4)

- A. `valid = False`
- B. `valid = True`

你正設計Python應用程式。

你需要該程式逐一查看數字清單，並在找到找到 4 時進行跳脫的動作。你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

注意：每個正確的選擇都可獲得一分。

```
nList = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
index = 0
___(1)___(index < 10) :
    print(nList[index])
    if nList[index] == 4 :
        ___(2)___
    else:
        ___(3)___
```

(1)

- A. while
- B. for
- C. if
- D. break

(2)

- A. while
- B. for
- C. if
- D. break

(3)

- A. continue
- B. break
- C. index += 1
- D. index = 1

三、執行輸入和輸出操作

建構並分析執行檔案輸入和輸出操作的程式碼區段

- 陳述式的打開；關閉；讀取；寫入；附加；檢查是否存在以及刪除

建構並分析執行主控臺輸入和輸出操作的程式碼區段

- 從主控臺讀取輸入；列印格式化文字；使用指令行引數

你必須開發一個簡單的Python檔案程式來執行以下的動作：

- 檢查檔案是否存在。
- 如果該檔案存在，就顯示檔案內容。
- 如果該檔案不存在，就使用指定的名稱新增檔案。
- 在檔案最後加入文字："這是檔案結尾"。

你需要完成程式碼以符合要求。

你要如何完成這段程式碼？回答時，在回答區選擇適當的程式碼片段。

注意：每個正確的選擇都可獲得一分。

```
import os
if __ (1) __
    file = open('theFile.txt')
    __ (2) __
    file.close()
file = __ (3) __
__ (4) __ ("這是檔案結尾")
file.close()
```

(1)

- A. `isfile('theFile.txt')`
- B. `os.exist('theFile.txt')`
- C. `os.find('theFile.txt')`
- D. `os.path.isfile('theFile.txt')`

(2)

- A. `output('theFile.txt')`
- B. `print(file.get('theFile.txt'))`
- C. `print(file.read())`
- D. `print('theFile.txt')`

(3)

- A. `open('theFile.txt', 'a')`
- B. `open('theFile.txt', 'a+')`
- C. `open('theFile.txt', 'w')`
- D. `open('theFile.txt', 'w+')`

(4)

- A. `Append`
- B. `file.add`
- C. `file.write`
- D. `write`

你正在設計一個檔案的函式。

如果檔案不存在，則返回 "檔案不存在"。如果該檔案存在，則該函式返回第一行的內容。請完成以下程式碼：

```
import os
```

```
def get_file_message(file):
```

你應該如何安排這些程式碼片段的順序來完成函式？回答時，請將適合的程式碼片段從程式碼片段清單移動到回答區，並按正確的順序排列。

A

```
    with open(file, 'r') as file:
```

B

```
        return "檔案不存在"
```

C

```
        return file.readline()
```

D

```
    if os.path.isfile(file):
```

E

```
    else:
```

D

A

C

E

B

你設計一個 Python 應用程式，需要將資料讀寫到文字檔中。如果檔案不存在，則必須新增它。如果檔案已有內容，將文字加到最後。

你應該使用哪個程式碼？

- A. `open("file_data", "a+")`
- B. `open("file_data", "w+")`
- C. `open("file_data", "r+")`
- D. `open("file_data", "r")`

你正在設計一個Python 程式來讀取學生資料的檔案，文件中包含了學生的班級、座號和姓名，下面顯示的是檔案中的資料範例：

```
'1A', 1, 'David'
```

```
'1A', 2, 'Mary'
```

程式碼必須符合以下的需求：

- 檔案的每一行都必須讀取和列印。
- 如果遇到空行，則必須忽略。
- 在完成所有行的讀取後，必須關閉檔案。

你創建了以下的程式碼。其中包含的行號只是做為參考。

```
01 students = open("students.txt", 'r')
02 eof = False
03 while eof == False:
04     line = students.readline()
05
06
07     print(line.strip())
08 else:
09     print("檔案結束")
10     eof = True
11 students.close()
```

在 05 及 06 行你應該編寫哪些程式碼？

A.

```
05 if line != '':
06     if line != "\n":
```

B.

```
05 if line != '\n':
06     if line != "":
```

C.

```
05 if line != '\n':
06     if line != None:
```

D.

```
05 if line != '':
06     if line != "":
```

有一個旅行社需要一個簡單的程式用來輸入合作飯店及民宿的調查資料。程式必須接受輸入並返回基於五顆星規模的平均評等。輸出必須四捨五入到小數第二位。

你必須完成這個程式碼以符合需求。

你要如何完成這個程式碼？回答時，在回答區選擇適當的程式碼片段。

注意：每個正確的選擇都可獲得一分。

```
sum = count = done = 0
average = 0.0
while (done != -1):
    rating = ___(1)___
    if rating == -1:
        break
    sum += rating
    count += 1
average = float(sum / count)
___(2)___ + ___(3)___
```

(1)

- A. print("請輸入評等 (1-5), 完成請輸入-1")
- B. float(input("請輸入評等 (1-5), 完成請輸入-1"))
- C. input("請輸入評等 (1-5), 完成請輸入-1")
- D. input "請輸入評等 (1-5), 完成請輸入-1")

(2)

- A. output("飯店的平均星級評等是:")
- B. console.input("飯店的平均星級評等是: ")
- C. println("飯店的平均星級評等是: ")
- D. print("飯店的平均星級評等是: ")

(3)

- A. format(average, '.2f'))
- B. format(average, '.2d'))
- C. {average, '.2f'}
- D. format.average.{2d}

在以下的程式碼：

```
import datetime
d = datetime.datetime(2018, 5, 16)
print('{:%m-%d-%y}'.format(d))
num = 1234567.89
print('{:,.2f}'.format(num))
```

執行程式後輸出的結果會是什麼？

- A. 05-16-18
1234567.89
- B. 05-16-2018
1,234,567.8900
- C. 05-16-18
1,234,567.89
- D. May-16-18
1,234,567.89

老師正在設計一個Python程式，學生可以使用它來記錄他們考試的平均分數。

該程式必須允許使用者輸入他們的名字和當前分數。該程式將輸出使用者名和使用者的平均分數。輸出必須符合以下要求：

- 使用者姓名必須是靠左對齊的。
- 如果使用者姓名少於20個字元，則必須在右側添加額外的空間。
- 平均分數在小數點左方是三位數，小數點右方是二位數。(XXX.XX)。

你要如何完成程式碼？回答時，請在回答區中選擇適當的程式碼片段。

注意：每個正確的選擇都可獲得一分。

回答區

```
name = input("請輸入姓名")
score = count = 0
while (score != -1):
    score = int(input("輸入你的分數：(輸入-1結束程式)"))
    if score == -1:
        break
    sum += score
    count += 1
    average = sum / count
print("__(1)___，你的平均分數是：__(2)___"%(name, average))
```

(1) A. %-20i B. %-20d C. %-20f D. %-20s

(2) A. %1.6s B. %6.2f C. %6.2d D. %2.6f

你正在設計一個函式以讀取資料檔案並將結果列印為格式化表格。

資料檔案中包含有關蔬菜的資訊。每個記錄都包含蔬菜的名稱、重量和價格。

你需要列印資料，使其看起來像下面的範例：

Potatos	5.4	2.33
Carrots	2.5	1.50
Corns	5.2	5.96

具體地說，列印輸出必須符合以下要求：

- 蔬菜名稱必須印在10個空格範圍內並靠左對齊。
- 重量必須印在5個空格範圍內並靠右對齊，小數點最多一個位數。
- 價格必須印在7個空格範圍內並右對齊，小數點後最多兩位數。

`□□□□".format(fields[0], eval(fields[1]), eval(fields[2]))`

程式碼片段

A. `print(" B. {10:0} C. {5:1f} D. {7:2f}
E. {2:7.2f} F. {1:5.1f} G. {0:10}`

A G F E

你創建了以下的程式碼。其中包含的行號只是做為參考。

```
01 def print_table(file):
02     data = open(file, 'r')
03     for record in data:
04         fields = record.split(",")
05
```

你應該如何完成05行的程式碼？回答時，請將適合的程式碼片段拖動到正確的位置。每個程式碼片段都可以使用一次，多次，或者不使用。你可能需要在窗格之間拖動分欄的寬度或捲動位置來查看內容。

你正在設計一個電子商務程式，它可以接受自使用者輸入，並以逗號分隔的格式輸出資料。你可以編寫以下程式碼來接受輸入：

```
product = input("輸入產品名稱： ")
```

```
qty = input("輸入數量： ")
```

輸出必須符合以下要求：

- 字串必須括在雙引號內。
- 數字不得用引號或其他字元括起來。
- 每個產品必須用逗號隔開。

你需要完成程式碼以符合要求。你應該使用哪三個程式碼片段？

A. `print('"{}"',{1}'.format(product, qty))`

B. `print('"' + product + '", ' + qty)`

C. `print('"%s",%s' % (product, qty))`

D. `print("{0},{1}".format(product, qty))`

E. `print(product + ', ' + qty)`

你為公司設計 Python 應用程式需要接受來自使用者的輸入並將該資訊列印到螢幕上。你的程式碼如下：

```
01 print("請輸入產品名稱:")
```

```
02
```

```
03 print(product_name)
```

在02行應該編寫哪個程式碼？

- A. `product_name = input()`
- B. `input(product_name)`
- C. `input("product_name")`
- D. `name = product_input`

四、文件和結構代碼

使用註解和文件字串的文件程式碼區段

- 使用縮排、空格、註解和文件字串；使用 Pydoc 產生文件

建構和分析包含功能定義的程式碼區段

- 呼叫簽章；預設值；返回；DEF；通過

公司正在將過去的進銷存程式碼轉移到Python，哪個文件語法是正確的？

- A.

```
// 返回公司帳戶的目前營業額  
def get_saletotal():  
    return saletotal
```
- B.

```
def get_saletotal():  
    /* 返回公司帳戶的目前營業額 */  
    return saletotal
```
- C.

```
' 返回公司帳戶的目前營業額  
def get_saletotal():  
    return saletotal
```
- D.

```
def get_saletotal():  
    # 返回公司帳戶的目前營業額  
    return saletotal
```

開發 Python 應用程式時，如何在程式碼中添加注釋？

- A. 將注釋放在最後一行程式碼之後，用空行隔開。
- B. 將注釋放在第一行程式碼之前，用空白行隔開。
- C. 將注釋放在任何一行的 ' 符號之後。
- D. 將注釋放在任何一行的 # 符號之後。

你設計一個函式，透過使用Python計算矩形的面積。

在函式中有加入注釋，程式碼如下：

```
01 # area_rectangle 函式計算矩形面積
02 # x 是長
03 # y 是寬
04 # 返回 x*y 的值
05 def area_retangle(x, y):
06     comment = "# 返回值"
07     return x*y          # x*y
```

針對下列每個敘述，如果該敘述是正確的就選擇Yes。否則請選擇 No。

- A. 01到04行在語法檢查時將被忽略。 Yes No
- B. 02和03行中的井字符號（#）非必填的。 Yes No
- C. 06行中的字串將被解釋為注釋 Yes No
- D. 07行包含內嵌注釋。 Yes No

運動器材公司正在設計一個程式用來記錄客戶跑步時的距離，你設計以下的 Python 程式碼：

```
01
02     name = input("請輸入你的姓名。")
03     return name
04
05     calories = kms * calories_per_km
07     return calories
08 distance = int(input("你這週跑了幾公里?"))
09 burn_rate = 80
10 runner = get_name()
11 calories_burned = calc_calories(distance, burn_rate)
12 print(runner, ", 你燃燒了大約 ", calories_burned, " 卡路里.")
```

在程式中必須要定義二個必要的函式。哪些程式碼片段你將使用在01及04行中？

- A. 01 def get_name():
- B. 01 def get_name(runner):
- C. 01 def get_name(name):
- D. 04 def calc_calories():
- E. 04 def calc_calories(kms, burn_rate):
- F. 04 def calc_calories(kms, calories_per_km):

你正設計一個函式用來增加遊戲中的玩家得分。該函式具有以下的要求：

如果沒有為變數points指定值，則points等於 1。

如果變數plus是True，那麼points必須加倍。

程式碼如下：

```
01 def add_score(score, plus, points):
02     if plus == True:
03         points = points * 2
04     score = score + points
05     return score
06 points = 5
07 score = 10
08 new_score = add_score(score, True, points)
```

針對下列每個敘述，如果該敘述是正確的就選擇Yes。否則請選擇 No。

A. 為了符合要求，必須將01行更改為以下內容：

def add_score(score, plus, points = 1): Yes No

B. 一旦使用預設值定義了任何參數，其右側的任何參數也必須使用默認值進行定義。 Yes No

C. 如果只用兩個參數呼叫函式，則第三個參數的值將為None。 Yes No

D. 03行的結果會改變在06行中變數 points 的值。 Yes No

你正在設計一個線上遊戲記分的Python應用程式。

目前需要符合以下條件的函式：

- 函式名為 `calc_score`
- 函式接收二個參數：目前分數和一個值
- 函式將值增加到目前分數
- 函式返回新分數

你要如何完成程式碼？回答時，請在回答區中選擇適當的程式碼片段。

回答區

```
____(1)____  ____ (2)____  
    current += value  
____(3)____
```

- (1) A. `calc_score` B. `def calc_score`
 C. `return calc_score`
- (2) A. `(current, value):` B. `():`
 C. `(current, value)` D. `()`
- (3) A. `pass current` B. `return current`
 C. `return` D. `pass`

五、執行疑難排解和錯誤處理

分析、偵測及修復出現錯誤的程式碼區段

- 語法錯誤、邏輯錯誤、運行時間錯誤

分析與建構處理例外狀況的程式碼區段

- 嘗試；排除；其他；最終；改善

在這個問題中需要評估以下段落中有底線的文字所敘述的內容是否正確。

程式碼的內容為：

```
import sys
try:
    file1 = open("fileIn.txt", 'r')
    file2 = open("fileOut.txt", 'w+')
except IOError:
    print(file1.name, '檔案不存在')
else:
    i = 1
    for line in file1:
        print(line.rstrip())
        file2.write("line " + str(i) + ": " + line)
        i = i + 1
    file1.close()
    file2.close()
```

fileIn.txt 檔不存在。在執行時這個程式碼將執行而不出錯。

再檢查帶底線的文字敘述。如果語法正確，請選擇 "不需要改變"。如果語法不正確，請選擇語法正確的答案選項。

- A. 不需要改變。
- B. 程式碼執行，但產生邏輯錯誤。
- C. 程式碼將產生執行階段錯誤。
- D. 程式碼將產生語法錯誤。

一家玩具公司使用 Python 來控制它的新玩具：機器車。這個程式有錯誤導致玩具車沿著一個圓圈不停的跑。請你為以下的程式碼除錯：

```
01 import math
02 # default motion for robot car
03 fire = True
04 move = 0
05 while(fire):
06     if move == 0:
07         turnValue = math.pi / move
08         move += 5
09     else:
10         turnValue = 0
11         move = 0
```

在程式碼中存在哪個錯誤？

- A. 05行導致執行階段錯誤，因為運算式不完整。
- B. 05行有語法錯誤，因為它應該讀取(`fire == True`)。
- C. 07行因為運算時除以零引起的執行階段錯誤。
- D. 08行有語法錯誤，因為 `+=` 是無效語法。

學校要求你對一些導致成績單出現問題的程式碼除錯。

在程式中已經宣告了下列變數：

```
scores = [80, 90, 75, 85]
```

```
count = 0
```

```
sum = 0
```

在以下的程式碼有二個錯誤：

```
for index in range(0, len(scores) - 1):
```

```
    count += 1
```

```
    sum += scores[index]
```

```
average = sum // count
```

```
print("總計分數是:", sum)
```

```
print("平均分數是:", average)
```

你應使用哪些程式碼來修復錯誤？回答時，請在回答區中選擇適當的程式碼片段。

注意：每個正確的選擇都可獲得一分。

回答區

```
for index in range ___(1)___
```

```
    count += 1
```

```
    sum += scores[index]
```

```
average = ___(2)___
```

```
print("總計分數是:", sum)
```

```
print("平均分數是:", average)
```

(1)

- A. (size(scores)):
- B. (size(scores)-1):
- C. (len(scores)+1):
- D. (len(scores)):

(2)

- A. sum/count
- B. sum**count
- C. sum*count

你設計了一個函式用來讀取資料檔案並且列印檔案的每一行。

以下是程式碼內容：

```
01 import os
02 def read_file(filename):
03     line = None
04     if os.path.isfile(filename):
05         datafile = open(filename, 'r')
06         while line != '':
07             line = datafile.readline()
08             print(line)
```

你需要更正程式碼，哪些三行有縮排問題？

- A. Line 01
- B. Line 02
- C. Line 03
- D. Line 04
- E. Line 05
- F. Line 06
- G. Line 07
- H. Line 08

你在測試以下程式碼時發現錯誤。其中包含的行號只是做為參考。

```
01 numList = [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
02 i = 0
03 while (i < 10)
04     print(numList[i])
05
06     if numList(i) = 6
07         break
08     else:
09         i += 1
```

你需要更正03行和06行中的程式碼。

你要如何更正程式碼？

- (1) 在03行中應使用哪個程式碼片段？
- A. `while (i < 10) :`
 - B. `while [i < 10]`
 - C. `while (i < 5) :`
 - D. `while [i < 5]`
- (2) 在06行中應使用哪個程式碼片段？
- A. `if numList[i] == 6`
 - B. `if numList[i] == 6 :`
 - C. `if numList(i) = 6 :`
 - D. `if numList(i) != 6`

在下列的語法敘述中，如果是正確的就選擇Yes，否則請選擇No。

在try 語法中可以有一個或多個except子句。

Yes No

在try 語法中可以有一個沒有except子句的 finally 子句。

Yes No

在try 語法中可以一個finally子句與except子句。

Yes No

在try 語法中可以有一個或多個finally子句。

Yes No

你正在創建一個接受使用者輸入的程式。這個程式必須將輸入值轉換為整數。如果輸入值不能轉換為整數，程式碼必須正確處理該錯誤。

你要如何完成程式碼？回答時，請在回答區中選擇適當的程式碼片段。

回答區

```
while True:
```

```
    ____ (1) ____
```

```
        x = int(input("請輸入一個數字: "))
```

```
        break
```

```
    ____ (2) ____ ValueError:
```

```
        print("這不是一個有效的數字. 再試一次...")
```

(1)

A. try: B. else: C. except: D. raise: E. finally:

(2)

A. try B. else C. except D. raise E. finally

六、使用模組和 工具執行操作

使用內建模組執行基本操作

- 數學；日期時間；IO；SYS；OS 路徑；隨機

透過使用內建模組解決複雜運算問題

- 數學；日期時間；隨機

在程式中要使用datetime模組中datetime函式，再設定 dt 為替代名稱，在導入時

應該使用哪個程式碼片段？

- A. `from datetime as dt`
- B. `from datetime import datetime as dt`
- C. `import datetime from datetime as dt`
- D. `import datetime.datetime as dt`

你設計一個讀取檔案後將檔案中的每一行列印出來的函式。

程式碼如下：

```
01 def print_file(filename):  
02     line = None  
03     if os.path.isfile(filename):  
04         data = open(filename, 'r')  
05         for line in data:  
06             print(line)
```

當你執行該程式時，你會收到03行上的錯誤。

導致錯誤的原因是什麼？

- A. 你需要導入os模組。
- B. path方法並不存在os模組中。
- C. path物件中不存在isfile方法。
- D. isfile 方法不接受一個參數。

你設計程式碼用來生成的隨機整數，最小值為 **11**，最大值為**20**。

你應該使用哪兩種函式？

- A. `random.randrange(11, 21, 1)`
- B. `random.randrange(11, 20, 1)`
- C. `random.randint(11, 20)`
- D. `random.randint(11, 21)`

你設計程式碼用來生成的隨機浮點數，最小值是0.0，最大值是1.0。

你應該使用哪個語法？

- A. `random.random()`
- B. `random.randrange(0.0, 1.0)`
- C. `random.randrange()`
- D. `random.randint(0, 1)`

你設計程式碼來產生一個隨機數來符合以下要求：

- 數字是2的倍數。
- 最低的數字是2。
- 最高的數字是50。

哪兩個程式碼片段將符合要求？

- A.

```
from random import randint
print(randint(1, 25) * 2)
```
- B.

```
from random import randint
print(randint(1, 50))
```
- C.

```
from random import randrange
print(randrange(2, 51, 2))
```
- D.

```
from random import randrange
print(randrange(2, 51, 1))
```

你正在設計一個處理數字的函式。該函式具有以下要求：

- 將浮點數傳遞到函式中
- 函式必須取浮點數的絕對值
- 函式必須無條件進位到整數

你應該使用哪兩個數學函式？

- A. `math.fabs(x)`
- B. `math.floor(x)`
- C. `math.fmod(x)`
- D. `math.ceil(x)`
- E. `math.frexp(x)`