

視窗程式設計

身分證號碼驗證

王昱景 Brian Wang
brian.wang@hkmci.com

功能需求

- 製作身分證號碼驗證程式，由鍵盤輸入10個字元身分證號碼，按“檢查”鈕後會按照公式檢查輸入的身分證號碼是否有誤
- 當身分證號碼輸入正確，則出現“身分證號碼合法!!”文字
- 若身分證號碼輸入錯誤，則出現“身分證號碼不合法!!”文字

- 身分證號碼總共有10個字元，如：
：“B124161817”為一組身分證號碼
- B代表A1，1代表A2，2代表A3，...其他以此類推
- 身分證號碼每個字元對應如下表：

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
B	1	2	4	1	6	7	8	1	7

- 將身分證號碼第一個英文字母轉換成下列對應的數值：

英文字母	A	B	C	D	E	F	G	H	I	J	K	L	M
對應數字	10	11	12	13	14	15	16	17	34	18	19	20	21
英文字母	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
對應數字	22	35	23	24	25	26	27	28	29	30	41	42	33

- 接著再使用下列公式計算身分證號碼是否合法？

$$(A1\text{的十位數} + A1\text{的個位數} * 9 + A2 * 8 + A3 * 7 + A4 * 6 + A5 * 5 + A6 * 4 + A7 * 3 + A8 * 2 + A9 * 1 + A10) / 10$$

- 假若計算結果的餘數為0，代表身分證號碼合法
- 假設”B124167817”為正確身分證號碼，其計算結果如下：

$$(\underline{1+1} * 9 + 1 * 8 + 2 * 7 + 4 * 6 + 1 * 5 + 6 * 4 + 7 * 3 + 8 * 2 + 1 * 1 + 7) / 10 \rightarrow \text{餘數為0}$$

↑表示身分證號碼第一個英文字B為11

身份證驗證程式

身份證號碼：

身份證驗證程式

身份證號碼：

身份證號碼不合法!!

身份證驗證程式

身份證號碼：

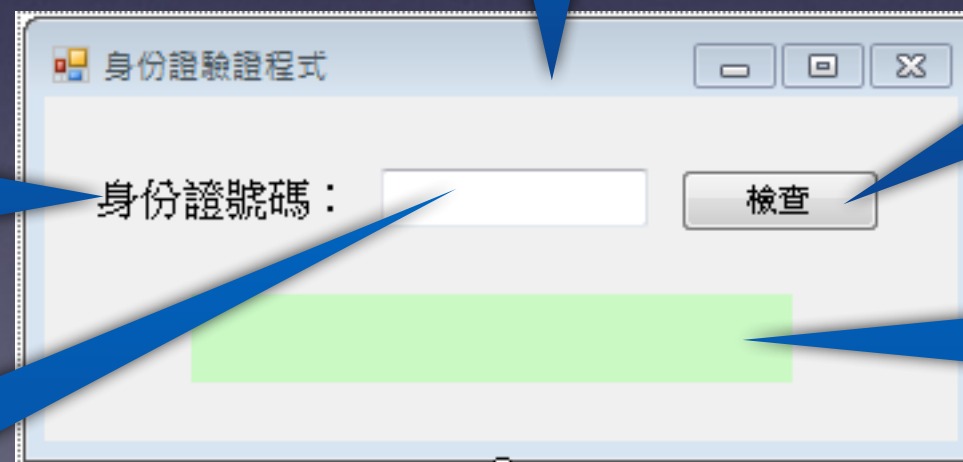
身份證號碼合法!!

實作步驟

- 新增“CheckID”專案
- 建立輸出入畫面

Font: 12pt
X,Y,W,L: 14, 27, 106, 23
Text: 身分證號碼 :
TextAlign: MiddleRight

Name: txtId
MaxLength: 10
X,Y,W,L: 126, 27, 100, 22



Size: 358, 167
Text: 身分證驗證程式

Name: btnCheck
X,Y,W,L: 238, 37, 75, 24
Text: 檢查

Name: labMsg
AutoSize: false
BackColor: 192, 255, 192
Font: 15.75pt, Bold
X,Y,W,L: 55, 74, 225, 33
TextAlign: MiddleLeft

- 檢查身分證號碼字元長度必須為10個字元
- 身分證號碼第1個字的位置只能輸入英文字母，並將小寫英文字母轉成大寫英文字母

```
// 取得身份證號碼第1個字是英文字母，並將小寫字母轉成大寫字母
string str = Convert.ToString(e.KeyChar).ToUpper();

if (str.CompareTo("A") < 0 || str.CompareTo("Z") > 0)
{
    e.Handled = true;
}
```


- 身分證號碼第2~10個字的位置只能輸入0~9或是使用倒退鍵

```
// 身分證第2~10個字只能輸入0~9或退位鍵
if ((e.KeyChar.CompareTo('0') < 0 || e.KeyChar.CompareTo('9') > 0)
    && e.KeyChar != '\b')
{
    e.Handled = true;
}
```

```
// 在txtId文字方塊按下鍵盤再放開時執行
private void txtId_KeyPress(object sender, KeyPressEventArgs e)
{
    if (txtId.Text == "")
    {
        // 取得身份證號碼第1個字是英文字母，並將小寫字母轉成大寫字母
        string str = Convert.ToString(e.KeyChar).ToUpper();

        if (str.CompareTo("A") < 0 || str.CompareTo("Z") > 0)
        {
            e.Handled = true;
        }
    }
    else
    {
        // 身份證第2~10個字只能輸入0~9或退位鍵
        if ((e.KeyChar.CompareTo('0') < 0 || e.KeyChar.CompareTo('9') > 0) && e.KeyChar != '\b')
        {
            e.Handled = true;
        }
    }
}
```

- 使用公式驗證身分證號碼是否正確
- 若正確則在labMsg標籤控制項上顯示”身分證號碼合法!!”文字
- 若身分證號碼輸入錯誤，則在labMsg標籤控制項上顯示”身分證號碼不合法!!”文字

```
// 按 [檢查] 鈕時執行
private void btnCheck_Click(object sender, EventArgs e)
{
    // 身分證字數不是10位,不合法
    if (txtId.Text.Length != 10)
    {
        labMsg.Text = "身分證號碼不合法!!";
        labMsg.ForeColor = Color.Red;
        return; // 離開此事件處理函式
    }

    // 將身分證號碼第一個英文字母轉換成對應的數值
    int[] head = new int[] {
        10, 11, 12, 13, 14, 15, 16, 17, 34, 18,
        19, 20, 21, 22, 35, 23, 24, 25, 26, 27,
        28, 29, 30, 41, 42, 33 };
    int id_head;
    id_head = head[Convert.ToChar(txtId.Text.Substring(0, 1).ToUpper()) - ('A')];

    // 組合轉換過的身分證號碼id
    string id = id_head + txtId.Text.Substring(1, 9);
}
```



```
// 組合轉換過的身分證號碼id
string id = id_head + txtId.Text.Substring(1, 9);

//使用公式計算身分證號碼是否合法
int n =
    Convert.ToInt32(id.Substring(0, 1)) +
    Convert.ToInt32(id.Substring(1, 1)) * 9 +
    Convert.ToInt32(id.Substring(2, 1)) * 8 +
    Convert.ToInt32(id.Substring(3, 1)) * 7 +
    Convert.ToInt32(id.Substring(4, 1)) * 6 +
    Convert.ToInt32(id.Substring(5, 1)) * 5 +
    Convert.ToInt32(id.Substring(6, 1)) * 4 +
    Convert.ToInt32(id.Substring(7, 1)) * 3 +
    Convert.ToInt32(id.Substring(8, 1)) * 2 +
    Convert.ToInt32(id.Substring(9, 1));

// 判斷身分證是否合法
if ((n % 10) == 0)
{
    labMsg.Text = "身分證號碼合法!!";
    labMsg.ForeColor = Color.Blue;
}
else
{
    labMsg.Text = "身分證號碼不合法!!";
    labMsg.ForeColor = Color.Red;
}
}
```