

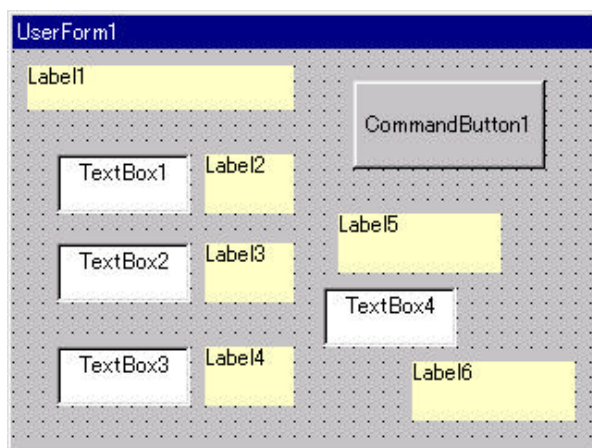
テーマ	Windowsプログラミングを体験しよう()				
概要	VBA (Visual Basic for Applications) を用いた簡単なプログラミングを体験する。				
ねらい	グレゴリオ歴の規則うるう年を考慮し、自分の生まれた日の曜日を求めるアルゴリズム(手順)を、簡単なプログラミングにより理解させる。				
関連する主な科目・項目	情報B (2)コンピュータの仕組みと働き イ コンピュータにおける情報の処理	難易度	易・ 普 ・難	想定時間数	約1時間
準備するもの	ExcelなどVBAが動作するもの				

1 学習の展開

(1) ユーザーフォームにコントロール配置

次のようにラベル (Label1 ~ Label6)、テキストボックス (TextBox1 ~ TextBox4)、コマンドボタン (CommandButton1) の各コントロールをフォームに配置する。

(注：一部説明のためにプロパティが変更された画面です)



(2) プロパティの設定

ラベル (Label1 ~ Label6) とコマンドボタン (CommandButton1) をそれぞれクリックし、プロパティウィンドウのCaption プロパティの値を次のように変更する。

コントロール	プロパティ	設定内容
Label1	Caption	曜日を求める
Label2	Caption	年
Label3	Caption	月
Label4	Caption	日
Label5	Caption	この日は
Label6	Caption	曜日です
CommandButton1	Caption	計算する

ユーザーフォーム画面の変化を確認する。


プログラムコード (命令) の記述例

```
Private Sub CommandButton1_Click()
    Dim mt(12) As Integer
    mt(1)=31: mt(2)=28: mt(3)=31: mt(4)=30
    mt(5)=31: mt(6)=30: mt(7)=31: mt(8)=31
    mt(9)=30: mt(10)=31: mt(11)=30: mt(12)=31
    y = Val(TextBox1.Text)
    m = Val(TextBox2.Text)
    d = Val(TextBox3.Text)
    If (y Mod 4)=0 And (y Mod 100) <> 0 Or
        (y Mod 400)=0 Then
        mt(2) = 29
    Else
        mt(2) = 28
    End If
    zure = y+(y-1) \ 4 - (y-1) \ 100 + (y-1) \ 400
    days = 0
    For k = 1 To m - 1
        days = days + mt(k)
    Next k
    week = (zure + days + d - 1) Mod 7
    If week = 0 Then
        TextBox4.Text = "Sunday"
    ElseIf week = 1 Then
        TextBox4.Text = "Monday"
    ElseIf week = 2 Then
        TextBox4.Text = "Tuesday"
    ElseIf week = 3 Then
        TextBox4.Text = "Wednesday"
    ElseIf week = 4 Then
        TextBox4.Text = "Thursday"
    ElseIf week = 5 Then
        TextBox4.Text = "Friday"
    ElseIf week = 6 Then
        TextBox4.Text = "Saturday"
    Else
        TextBox4.Text = "Error"
    End If
End Sub
```

(3) プログラムコード(命令)の記述

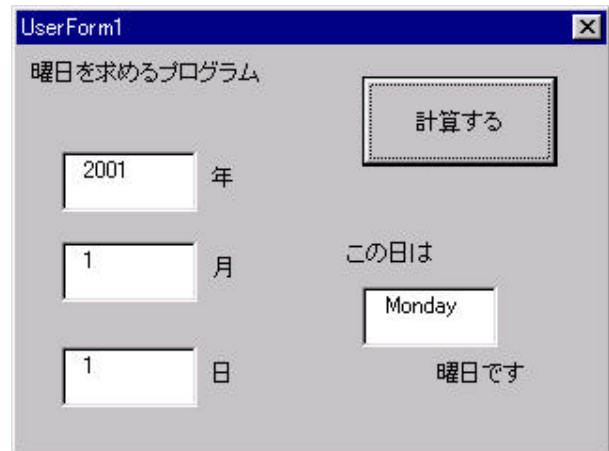
コマンドボタンをダブルクリックし、コード記述画面を出して、前ページのようにコードを記述する。

(4) プログラムの実行

メニューバー[実行]-[1-ザ-フォームの実行]
またはツールバーの  をクリックする。

実行画面の年,月,日のテキストボックス
に自分の誕生日などの数字を入力する。

「計算する」と書かれたコマンドボタンを
クリックすると、その日の曜日が表示さ
れる。



UserForm1
曜日を求めるプログラム

計算する

2001 年

1 月

1 日

この日は
Monday
曜日です

(5) プログラムの解説

年月日の値から、どのように曜日が求められるか、そのアルゴリズム(手順)を考えてみる。地球の公転周期は 365.2422日である。1年を365日にするはずれが生じるため、4年に一度閏年が設けられる。このグレゴリオ暦の規則から

西暦の年数が4で割れ、かつ100で割り切れない年を閏年とする。

西暦が400で割切れる年も閏年とする。

都合上西暦1年1月1日を月曜日とする。

1年365日を1週間の7日で割ると1日余ることから、Y年たつとY日分曜日がずれることになる。また、閏年の年数分だけさらにずれる。このずれる日数は次の式で求まる。

$$\text{zure} = y + (y-1) \cdot 4 - (y-1) \cdot 100 + (y-1) \cdot 400$$

曜日を求めたい年の1月1日からの日数は、とすることで、days日と求められる。

$$\text{days} = 0$$

For k = 1 To m - 1

$$\text{days} = \text{days} + \text{mt}(k)$$

Next k

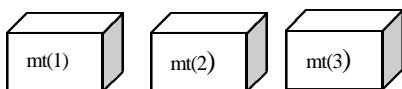
西暦1年1月1日(月曜日)からのずれた日数を7で割り、余りの数字 week から曜日を求める。

$$\text{week} = (\text{zure} + \text{days} + d - 1) \text{ Mod } 7$$

week=0を日曜日, week=1を月曜日...と考え、次の条件判断文で答えをテキストボックスに表示する。

If week = 0 Then TextBox4.Text = "Sunday"

2 進んだ学習



配列変数(一次元配列)

このプログラムで使われている mt(1)~mt(12) という文字は配列変数と呼ばれる。配列名(添字)という形で記述される。

配列は Dim m 文であらかじめ宣言しておく必要がある。

```
Dim mt(12) As Integer
```