

Visual Basic 第3回

Visual Basicの繰り返し1

Do…Loop文

テキスト P.175~183

考え方はJavaのWhile文と同じ。

UntilキーワードはWhileの逆で、条件式がFalseの場合に繰り返す。

①後判断(Until型)	②後判断(While型)	③前判断(Until型)	④前判断(While型)
Do 処理 Loop Until 条件	Do 処理 Loop While 条件	Do Until 条件 処理 Loop	Do While 条件 処理 Loop

前判断型は一回も処理が行われない可能性がある。

Visual Basicの繰り返し2

テキスト P.185 ~193

For...Next文

考え方はJavaのfor文と同じ。

基本構文

```
For Dim i As Integer = 0 To 4 Step 1
```

処理

Next

カウント変数iの宣言

0で初期化

4になるまで繰り返し

1ずつ増加

※

- Step 1ならStep句を省略可能
- カウント変数はFor文の宣言より前に宣言することも可

Visual Basicの繰り返し3

For Each...Next文

テキスト P.190~193

考え方はJavaの拡張for文やforEachメソッドと同じ。

Pythonのfor文とも似ている。

基本構文

For Each 変数名 As 型 In コレクション

処理

Next

※コレクションは配列のように複数の項目を含むようなデータのこと。

EX) なぜFor Each文があるのか、使われているか、考えてみよう。

繰り返しを途中で抜ける テキスト P.195~199

Do … Loop文の場合

Exit Doステートメント

For … Next/For Each … Nextの場合

Exit Forステートメント

ListBoxコントロールと乱数生成

ListBoxコントロール

これ

List状でデータを表示することができるコントロール。

要素を追加するには以下のメソッドを使用する。

ListBox名.Items.Add(**Data**)

For Each文で繰り返し処理をしたいときは、コレクションの部分に ListBox名.Items をセットする。

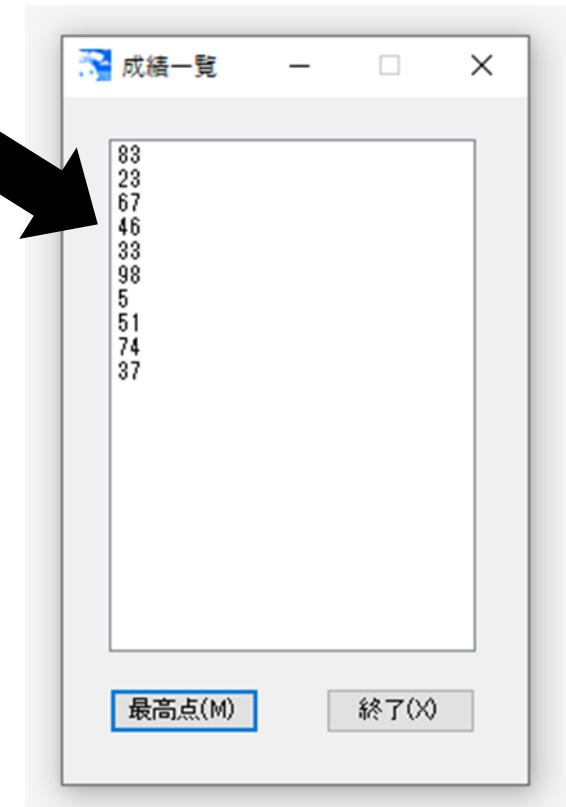
乱数生成

Randomオブジェクトを使用する。宣言は以下の通り。

```
Dim r As Random = new Random()
```

指定した範囲(0~100) の乱数を生成するには、以下のメソッドを使う。

```
Dim int As Integer = r.Next(0,101)
```



演習

実習

テキストChapter6 06(P. 202)のプログラミングにチャレンジを読みながら進めてみなさい。終わった人は、P.210の練習問題A,Bにチャレンジしてください。(できれば、ボタンを変更するのではなく、ボタンを追加するような感じで。)

それも終わった人は声をかけてください。追加の課題を渡します。

HINT

- ・画面が読み込みされたとき（表示されたとき）というイベントは、

「 MyBase.Load 」

- ・ポップアップの表示は以下のメソッドを使用する。

MessageBox.Show(ポップアップ本文, ポップアップのタイトル, MessageBoxButtons.OK)