

# Экономическая информатика I

## Visual Basic for Applications

Алексей Николаевич Попов

# Константы

- арифметические (числовые) – 5; -12; 2,4E15
- текстовые (символьные) – "пример процедуры"
- Логические (булевы) – true, false
- Даты - #08/30/20012#

\* ? 31.08.2012

# Переменные

- Переменные – это область памяти, где программы сохраняют значения.
- Переменные не должны содержать пробелов, но могут содержать знак подчеркивания (\_).
- Примеры: x, y1, top, Проц\_ставка.

# Встроенные функции

Имя для VBA	Описание	Пример	Имя для Excel
Abs	Абсолютное значение	$\text{Abs}(-4) = 4$	ABS
Log	Натуральный логарифм	$\text{Log}(1) = 0$	LN
Exp	Показательная функция с основанием $e$	$\text{Exp}(1) = 2.71$	EXP
Sqr	Извлечение квадратного корня	$\text{Sqr}(9) = 3$	КОРЕНЬ
Int	Округление отбрасыванием дробной части	$\text{Int}(2,6) = 2$ $\text{Int}(-1,2) = -2$	ЦЕЛОЕ
Rnd	Генерация случайных чисел, равномерно распределенных на отрезке $[0, 1]$	$\text{Rnd}()$ (аргумент не требуется)	СЛЧИС

# Арифметические выражения

- 3.14                      константа
- X1                        переменная
- Cos(1.57)                функция
- $3 * X^2 + z1$             эквивалентно обычной записи  $3X^2 + z1$
- $2 + a/b/c$                 эквивалентно  $2 + \frac{a}{bc}$
- $(2+a)/b * c$             эквивалентно  $\frac{2+a}{b} \cdot c$
- $(4 + \text{Cos}(x+y))/(1+z)$    эквивалентно  $\frac{4 + \cos(x+y)}{1+z}$

# Символы операций сравнения

=	равно
◇	неравно
<	меньше
>	больше
<=	меньше или равно
>=	больше или равно
Like	сравнение строк

# Примеры использования операторов сравнения

- $5 \geq 4$  равно True
- "Миша" = "Маша" равно False (текстовые константы не равны).
- "Маша" > "Миша" равно False («а» по алфавиту идет раньше «и»)
- $-1 = \text{True}$  равно True (числовое значение True -1)
- $0 \triangleq \text{False}$  равно False (числовое значение False 0)
- "Маша" Like "M\*a" равно True (\* заменяет любое количество любых символов)
- "Маша" Like "M?[a-я]a" равно True (? заменяет один любой символ, [a-я] заменяет любой символ в диапазоне от «а» до «я»).



# Символы логических операций

Not	логическое отрицание
And	логическое умножение (логическое «И»)
Or	логическое сложение (логическое «ИЛИ»)

L1	L2	Not L1	L1 And L2	L1 Or L2
True	True	False	True	True
True	False	False	False	True
False	True	True	False	True
False	False	True	False	False



# Логические выражения

- False                      константа;
- True And  $7 < 1$            равно False;
- True Or  $8 = 9$             равно True;
- $x1 + x2 < 0$                 значение зависит от значений переменных  $x1$  и  $x2$ ;
- $X < 100$  And  $X \geq 50$     значение зависит от значения  $X$ ;

# Приоритеты операций

## Арифметические

выше

Возведение в степень (^)

Унарный минус (—)

Умножение и деление (\*, /)

Целочисленное деление (\)

Остаток от деления (Mod)

Сложение и вычитание (+, —)

ниже

Конкатенация (&)

# Приоритеты операций

## **Сравнение**

Равно (=)

Неравно ( $\neq$ )

Меньше ( $<$ )

Больше ( $>$ )

Меньше или равно ( $\leq$ )

Больше или равно ( $\geq$ )

Like

# Приоритеты операций

## **Логические**

Отрицание Not

Логическое умножение And

Логическое сложение Or

# Оператор присваивания

```
ПроцСтавка = 0,13  
x1 = 1500 + 200  
ПодНалог = ПроцСтавка * x1
```

```
FullName = "Иванов " & "И. И."
```

```
X = 1  
X = X+2  
X = 3*X
```

```
Alpha = 2  
Beta = 3  
Gamma = Alfa + Beta
```

# Ввод данных в диалоговом режиме

```
Sub Приветствие()  
    Dim user  
    user = InputBox(«Ваше имя? »)  
    MsgBox "Привет", user  
End Sub
```

# Ввод данных в диалоговом режиме

```
Sub Ввод()  
    Dim Message, Title, Default, MyValue ' Объявляем переменные  
    Message = "Введите число между 1 и 3" ' Сообщение  
    Title = "Окно ввода" ' Заголовок  
    Default = "1" ' Начальное значение переменной  
    MyValue = InputBox(Message, Title, Default)  
End Sub
```



# Однострочный оператор IF/Then

**If L Then P,**

где L - логическое выражение, P -  
оператор.

**Function Частн(х,у)**

**If  $y \neq 0$  Then Частн =  $x/y$**

**End Function**

# Многострочный If / Then / End

## If

```
If L Then  
    P1  
    P2  
    ...  
    Pn  
End If
```

Здесь L - логическое выражение

```
Function Корень4(x)  
    Dim Кор2  
    If x >= 0 Then  
        Кор2 = sqr(x) ' Кор2 - промежуточная переменная  
        Корень4 = sqr(Кор2)  
    End If  
End Function
```

# Оператор If / Then / Else / End If

**If L Then**

P1

...

Pn

**Else**

Q1

...

Qm

**End If**

Здесь L - логическое выражение, P1,..., Pn, Q1,..., Qm - операторы.

# Оператор Select Case / Case / Case Else / End Select

Select Case A

Case A1

P1

17

....

Case An

Pn

Case Else

Ps

End Sel

Здесь A – выражение, A1,A2,...,An – подмножества множества значений A.

**Function FunInt(X)**

**Select Case X**

**Case Is <= -1** ' Is заменяет текущее значение переменной X.

FunInt = 0

**Case -1 To 3**

FunInt = 0,2

**Case 3 To 5**

FunInt = 0,7

**Case Else** ' эквивалентная запись Case Is > 5

FunInt = 1

**End Select**

**End Function**

# Оператор цикла For / Next

**For** I = start **To** end

P1

...

Pn

**Next** I

Здесь I – переменная (счетчик), start – начальное значение счетчика, end – конечное

значение счетчика, P1, ..., Pn – операторы.

# Пример

```
Function Сумм1(N1,N2)
    Dim k
    Сумм1 = 0
    For k =N1 To N2
        Сумм1 = Сумм1 + k
    Next k
End Function
```



# Оператор цикла DO While / Loop

**Do While L**

P1

...

Pn

**Loop**

Здесь L - логическое выражение, P1, ..., Pn - операторы.

# Пример

```
Function Сумм100(Nstart)
```

```
    Dim n
```

```
    Сумм100 = 0 ' начальное значение суммы
```

```
    n = Nstart
```

```
    Do While n <= 100
```

```
        Сумм100 = Сумм100 + n
```

```
        n = n + 1
```

```
    Loop
```

```
End Function
```

# Оператор цикла Do Until / Loop

**Do Until L**

P1

...

Pn

**Loop**

Здесь L – логическое выражение, P1, ..., Pn – операторы.

# Пример

```
Sub PassWord()
```

```
    Dim PW
```

```
    PW = " " ' начальное значение пароля -  
пробел
```

```
    Do Until PW = "Stud123"
```

```
        PW = InputBox( "Пароль?" )
```

```
    Loop
```

```
End Sub
```

# Оператор выхода Exit

- Exit Sub
- Exit Function
- Exit For
- Exit Do