

## ЛЕКЦИЯ 5

### ТЕМА: ОРГАНИЗАЦИЯ УСЛОВИЕ В ПРОГРАММЕ. ОПЕРАТОР IF

1. Условные выражения
2. Условный оператор IF
3. Безусловный оператор перехода Goto

#### *Условные выражения*

Основанием для принятия решений в управляющих конструкциях являются условные выражения, поэтому предварительно необходимо сказать несколько слов об этих выражениях и работе с ними.

*Условные выражения* — это такие выражения, которые возвращают одно из двух значений **True** (Истина) или **False** (Ложь). В условных выражениях используются операторы сравнения, приведенные в табл. 6.1.

*Таблица 4.1. Операторы сравнения для условных выражений*

Оператор	Назначение
=	Равно
>	Больше
<	Меньше
<>	Не равно
>=	Больше или равно
<=	Меньше или равно

Над условными выражениями можно выполнять действия логической математики (логические операции), а именно:

- **AND (И)** — возвращает значение **True** (Истина), если все участвующие в операции выражения имеют значение **True**. В остальных случаях возвращается значение **False** (Ложь);

- **OR (ИЛИ)** — возвращает значение **True**, если хотя бы одно из участвующих в операции выражений имеет значение **True**. В случае, когда все выражения имеют значение **False**, возвращается значение **False**;

- XOR (Исключающее ИЛИ) — возвращает значение **True** (Истина), если только одно из участвующих в операции выражений имеет значение **True**. В остальных случаях возвращается значение **False**;

- NOT (НЕ) — операция отрицания. Возвращает обратное для значения выражения значение, то есть если выражение равно **True**, то возвращается **False** и наоборот, если значение выражения равно **False**, то возвращается значение **True**.

### **Условный оператор *If... Then***

Условный оператор *If... Then* применяется в том случае, когда необходимо выполнить один или группу операторов при соблюдении определенного условия, то есть когда значение заданного условия равно **True**. Существует две разновидности данного оператора: однострочный и многострочный. Однострочный оператор имеет следующий синтаксис:

#### ***IF условие Then Оператор***

В этом операторе условие и выполняемые при соблюдении условий действия располагаются в одной строке.

В том случае, если при выполнении условия требуется выполнение блока операторов, используется многострочный оператор, имеющий следующий синтаксис:

#### ***IF условие Then***

*Группа операторов*

#### ***End If***

Исходя из синтаксиса, приведенные ниже программные коды выполняют одни и те же действия:

' Однострочный оператор

*IF y>20 Then y=2*

' Многострочный оператор

*IF y>20 Then*

*y=2*

*End If*

После имени конструкции **if** должно следовать логическое выражение, содержащее условие. В качестве условия могут выступать следующие логические выражения:

- сравнение переменной с другой переменной, константой или функцией;
- любая переменная, выражение, поле базы данных или функция, принимающие значения **True** или **False**.

Ключевое слово **End if** обозначает конец многострочной конструкции и его наличие в команде в этом случае обязательно. Если указанное условие выполняется, то есть результат проверки равен **True**, то Visual Basic выполнит конструкции, следующие за ключевым словом **Then**. Если условие не выполняется, то Visual Basic переходит к выполнению операторов, следующих за указанным оператором.

Условный оператор также обладает следующими синтаксисами:

***IF условие Then .***

*1-ая группа операторов*

*Else*

*2-ая группа операторов*

*End If*

2.

***IF 1-ое условие Then***

*1-ая группа операторов*

***ElseIf 2-ое условие Then***

*2-ая группа операторов*

*.....*

*.....*

*Else*

*n-ая группа операторов*

*End If*

**GoTo Метка**

где Метка – метка с которой продолжится выполнение программы.

**Примечание:** Частое употребление оператора **GoTo** приводит к запутыванию программ. Программисты считают употребление оператора **GoTo** плохим тоном. Старайтесь реже его употреблять.

Пример

metka:

Print "Привет!!!" ' -бесконечный цикл

GoTo metka: