

ЛЕКЦИЯ 14

ТЕМА: РАБОТА С ВНЕШНИМИ ФАЙЛАМИ В VISUAL BASIC.

1. Работа с файлами в VB
2. Открытие внешних файлов
3. Закрытие внешних файлов

1. Работа с файлами

Традиционный подход при работе с файлами остается неизменным практически с самых первых версий Visual Basic и заключается в использовании функций и операторов, обеспечивающих прямой доступ к информации в файлах. Функции и операторы, используемые при работе с файлами, приведены в табл. 9.1. В столбце **Тип файла** этой таблицы приняты следующие сокращения типов файлов:

- П — файл последовательного доступа;
- Пр — файл произвольного доступа;
- Б — бинарный файл.

В данной лекции мы рассмотрим только основные функции и операторы, необходимые для получения навыков работы с файлами.

Таблица 14.1. Функции и операторы для работы с файлами

| Функция,оператор | Описание | Тип файла |
|------------------|--|-----------|
| Open | Открывает файл | П, Пр, Б |
| Close | Закрывает все файлы | П, Пр, Б |
| Close # | Закрывает файл по идентификатору (дескриптору) | П, Пр, Б |
| Reset | Закрывает все открытые файлы, записывает содержимое буферов | П, Пр, Б |
| Print tt | Записывает данные в файл | П |
| FileCopy | Копирует файл | П, Пр, Б |
| EOF | Определяет метку конца файла | П, Пр, Б |
| FileAttr | Возвращает режим доступа открытого файла | П, Пр, Б |
| FileDateTime | Возвращает дату и время создания файла | П, Пр, Б |
| FileLen | Возвращает размер файла в байтах | П, Пр, Б |
| FreeFile | Возвращает номер свободного идентификатора файла (дескриптора) | П, Пр, Б |
| GetAttr | Получает атрибуты файла | П, Пр, Б |
| SetAttr | Устанавливает атрибуты файла | П, Пр, Б |

| | | |
|--------------|--|----------|
| Loc | Возвращает номер текущей позиции в файле | Пр, Б |
| LOF | Возвращает размер открытого файла в байтах | П, Пр, Б |
| Seek | Устанавливает на заданную номером позицию или запись в файле | П, Пр, Б |
| Dir | Возвращает содержимое текущей папки | П, Пр, Б |
| Kill | Удаляет файл | П, Пр, Б |
| Lock | Блокирует файл при работе в многопользовательской среде | П, Пр, Б |
| Unlock | Снимает блокировку файла в многопользовательской среде | П, Пр, Б |
| Name | Задаёт (переименовывает) имя файла | П, Пр, Б |
| Get # | Читает данные из файла | Пр.Б |
| Input | Читает данные из файла | П, Б |
| Input # | Читает данные из файла | П |
| Line Input # | Читает строку из файла | П |
| Put # | Записывает данные в файл | Пр, Б |
| Write # | Записывает данные в файл | П |

Для удобства сгруппируем функции и операторы по выполняемому действию, как это принято в Visual Basic. Такое объединение удобно при выборе функции или оператора для выполнения необходимых действий с файлами (табл. 14.2).

Таблица 14.2. Функции и операторы для работы с файлами по группам

| Выполняемое действие | Функции, операторы |
|--------------------------------------|---|
| Открыть или создать файл | Open |
| Закрывать файл | Close, Reset |
| Определение параметров вывода данных | Format, Spc, Tab, Width # |
| Скопировать файл | FileCopy |
| Получить информацию о файле | EOF, FileAttr, FileDateTime, FileLen, FreeFile, GetAttr, Loc, LOF |
| Организовать управление файлами | Dir, Kill, Lock, Unlock, Name |
| Прочитать данные из файла | Get #, Input, Input ft, Line Input # |

| | |
|-------------------------------------|---------------------------|
| Получить информацию о размере файла | FileLen |
| Установить атрибуты файла | SetAttr |
| Найти позиции в файле | Seek |
| Записать данные в файл | Print tt, Put #, Write ft |

2. Открытие файлов

Как было указано выше, работа с каждым из типов файлов имеет свои особенности. Однако есть два действия, общие для всех типов файлов — их открытие и закрытие.

Понятно, что перед тем как записать данные в файл или прочитать данные из файла, необходимо сначала открыть этот файл. Открытие файла выполняется оператором `open`:

`Open pathName For mode [access] [lock] As [#]fileNumber [Len=recLength]`

где:

- `pathName` — полное имя файла;
- `mode` — режим доступа к файлу. Может принимать следующие значения: **Append**, **Binary**, **Input**, **Output** или **Random**;
- `access` — тип доступа к файлу. Определяет характер действий с файлом — чтение или запись данных. Может принимать значения: **Read** (Чтение), **Write** (Запись) или **Read/Write** (Чтение/Запись);
- `lock` — тип разрешения доступа к файлу другим процессам. Определяет возможность одновременной работы с файлом нескольких приложений или нескольких пользователей. Может принимать значения: **Shared** (Общий), **Lock Read** (Блокировка чтения), **Lock Write** (Блокировка записи) или **Lock Read Write** (Блокировка чтения и записи);
- `fileNumber` — целочисленное выражение, задающее идентификатор файла (дескриптор). Может иметь значение от 1 до 511 включительно;
- `recLength` — число, определяющее размер буфера данных для записи/чтения в файлах прямого доступа. Для файлов произвольного доступа это число задает длину одной записи файла. Может иметь значение до 32,767 (байтов).

При работе оператора `Open` создается специальный счетчик номеров (идентификаторов) открытых файлов (в операторе это параметр `fileNumber`) для однозначного определения файла, с которым программа работает в текущий момент.

Если указанный в операторе `open` файл не найден по заданному пути или не существует, он будет создан этим оператором для режимов доступа `Append`, `Binary`, `output` или `Random`. Для режима доступа `input` новый файл не создается.

Важной особенностью режимов доступа является возможность многократного открытия файла для режимов `Binary`, `input` и `Random`, то есть файл можно открыть несколько раз с разными номерами. Но для режимов

доступа Output и Append, используемых для записи данных, это недопустимо.

Если номер открываемого файла специально не контролируется и не задается программой, его можно узнать с помощью функции FreeFile, возвращающей последний свободный номер открываемого файла.

Замечание

При использовании оператора Open необходимо, чтобы режим доступа соответствовал типу открываемого файла.

3.Заккрытие файлов

Заккрытие файлов выполняется очень просто. Для этого необходимо использовать оператор close, имеющий следующий синтаксис:

Close [fileNumberLlist]

где fileNumberLlist — список закрываемых файлов, представленных номерами файлов и перечисляемых через запятую: [#fileNumber] [,#fileNumber] . . . [,# fileNumber]. При этом номер файла аналогичен номеру файла в функции open.

Предупреждение

Необходимо иметь в виду, что если список файлов не указан, оператор close закрывает все открытые файлы.