**О Логах программы**

*Логи можно включать и выключать! Для этого зайдите в программу в режиме администратора и включи панель навигации (см.* [*здесь*](Получение%20Полного%20Доступа.docx#РежимОдмэна) *как это сделать) . Найди таблицу TableTheSystem2 и убери/поставь галочку в столбцах LogState(вкл/откл записи ошибок) и Debug\_State(вкл/откл записи действий пользователя)*

В программе есть возможность прописывать все ошибки и все действия (90% действий) совершаемые пользователями в процессе работы. Все ошибки и действия пользователя записываются в соответствующие текстовые файлы. Они находятся в V:\TEEEEST

- LogFile – файл с записями ошибок

-DebugFile – файл со всеми действиями ошибок

*В лог файл одна запись выглядит следующим образом:*

**10.11.2020\*\*\*11:14:58\*\*\*PRO001\*\*\*Текущая запись отсутствует.\*\*\*CoprusFloor(MainDatabase)**

Здесь указано слева-направо: дата, затем – время, сетевое имя компьютера, Описание ошибки, Имя Процедуры (Имя формы)

*В DebugFile Запись выглядит следующим образом:*

**10.11.2020\*\*\*11:33:29\*\*\*PRO001\*\*\*Форма:Interface\_Zayavok\_Otdeleniya\*\*\*Процедура:Filter\_Byro\_AfterUpdate\*\*\*Знач: Ч. необраб.**

Здесь указано слева-направо: дата, затем – время, сетевое имя компьютера, Имя Формы, Имя Процедуры, Значение Полученное/Обрабатываемое в процедуре.

Лог-файл содержит все записи, DebugFile же, содержит записи за текущий день. Каждый день записи Debug архивируются. Это происходит следующим образом: как только наступает новый день, файл DebugFile перемещается в папку Debugs (V:\TEEEEST\ Debugs), там ему к имени DebugFile Добавляется еще и дата прошедшего дня. А в пути V:\TEEEEST создается новый файл с именем DebugFile.

Теперь рассмотрим код (logModule):

1. **Public Sub ReadWriteDebug(ByRef Descrp As String)**
   1. On Error GoTo ErB:
   2. If CheckState("Debug\_State") = True Then
      1. Dim DebugDate As Date
      2. Dim fso As Object
      3. Dim ts As Object
      4. Set fso = CreateObject("scripting.filesystemobject")
      5. Set ts = fso.GetFile("V:\TEEEEST\DebugFile.txt")
      6. DebugDate = CDate(Format$(ts.DateLastModified, "dd\/mm\/yyyy"))
      7. If DebugDate <> Date Then
      8. Call ArchiveDebug(fso, ts)
      9. End If
      10. Set ts = fso.OpenTextFile("V:\TEEEEST\DebugFile.txt", 8, True)
      11. Dim TextDesrp As String
      12. TextDesrp = Date & "\*\*\*" & Time() & "\*\*\*" & Environ$("computername") & "\*\*\*" & Descrp
      13. ts.WriteLine TextDesrp
      14. Set ts = Nothing: Set fso = Nothing
      15. AddIntoTXTfile = Err = 0
   3. End If
2. ErB:
   1. 'MsgBox Err.Description
3. **End Sub**

В начале идет проверка на то, разрешена ли запись в DebugFile (1.2). Для этого вызывается соответствующая функция – CheckState, которой передается строка "Debug\_State".

Код функции:

1. **Private Function CheckState(ByRef What As String) As Boolean**
   1. Dim db As Database
   2. Dim rs As DAO.Recordset
   3. Set db = DBEngine.Workspaces(0).Databases(0)
   4. SQLstr = "SELECT [" & What & "] From TableTheSystem2"
   5. Set rs = db.OpenRecordset(SQLstr)
   6. CheckState = CBool(rs.Fields(0))
2. **End Function**

В функции идет обращение к таблице TableTheSystem2, к ее столбцу с названием, которое мы передали в функцию (т.е. Debug\_State) и соответственно мы получаем какое там стоит значение. Это значение и возвращается функцией. Если оно будет True – то после 1.2 код процедуры (не функции) будет продолжен. Далее в 1.2.1-1.2.5 мы загружаем файл DebugFile, выясняем когда он был последний раз изменен. Если последнее изменение было вчера (1.2.6), т.е. последнее изменение не соответствует сегодняшнему числу, то реализуется процесс указанный выше желтым. Описанная процедура реализуется в отдельной процедуре описанной ниже:

1. **Private Sub ArchiveDebug(ByRef fso As Object, ByRef ts As Object)**
   1. Set fs2 = CreateObject("Scripting.FileSystemObject")
   2. Dim PathDebug As String
   3. PathDebug = Replace("V:\TEEEEST\Debugs\Debug" & Format$(Date - 1, "yyyy\/mm\/dd") & ".txt", "/", "")
   4. 'Set fs3 = fs2.CreateTextFile(PathDebug, True)
   5. fso.CopyFile Source:="V:\TEEEEST\DebugFile.txt", Destination:=PathDebug
   6. 'Open "V:\TEEEEST\DebugFile.txt" For Output As fs3
   7. 'Kill ts
   8. ts.Write ""
   9. 'Set ts = fso.CreateTextFile("V:\TEEEEST\DebugFile.txt", True)
   10. 'Set ts = Nothing
2. **End Sub**

Остальная часть кода реализует запись в DebugFile.

Код процедуры делающий запись в LogFile по смыслу такой же, только там нет механизма архивации данных:

1. **Public Sub ReadWriteLog(ByRef Descrp As String, ByRef Location As String)**
   1. On Error GoTo ErB:
   2. If CheckState("LogState") = True Then
      1. Set fso = CreateObject("scripting.filesystemobject")
      2. Set ts = fso.OpenTextFile("V:\TEEEEST\LogFile.txt", 8, True)
      3. Dim TextDesrp As String
      4. TextDesrp = Date & "\*\*\*" & Time() & "\*\*\*" & Environ$("computername") & "\*\*\*" & Descrp & "\*\*\*" & Location
      5. ts.WriteLine TextDesrp
      6. Set ts = Nothing: Set fso = Nothing
      7. AddIntoTXTfile = Err = 0
   3. End If
2. ErB:
   1. 'MsgBox Err.Description
3. **End Sub**