

Firebird Embedded C# Code Example

5-Dec-2019

```
//-----
static public bool boModifyRecordUsingKeyIDValue(string stDbPathFilename,
                                                string stUserID,
                                                string stPassword,
                                                string stTableName,
                                                string stFieldKeyIDName,
                                                string stKeyIDValue,
                                                string stFieldNamesWithoutKeyIDNameCommaDelimited,
                                                List<string> liststFieldModifiedValuesWithoutKeyIDValue)
{
    bool boErrorFlag = false;
    string stConn = new FbConnectionStringBuilder {
        Database = stDbPathFilename,
        UserID = stUserID,
        Password = stPassword,
        ServerType = FbServerType.Embedded,
        ClientLibrary = @"(path)\fbclient.dll"
    }.ToString();
    try {
        using(FbConnection fbConn = new FbConnection(stConn)) {
            fbConn.Open();
            List<string> liststFieldNamesWithoutKeyIDName = new List<string>();
            ParseStringToListst(stFieldNamesWithoutKeyIDNameCommaDelimited, ',', liststFieldNamesWithoutKeyIDName);
            string stCmd = "UPDATE " + stTableName +
                " SET " + liststFieldNamesWithoutKeyIDName[0] + " = @p0";
            for (int iii = 1; iii < liststFieldNamesWithoutKeyIDName.Count(); iii++)
                stCmd += ", " + liststFieldNamesWithoutKeyIDName[iii] + " = @p" + iii.ToString();
            stCmd += " WHERE " + stFieldKeyIDName + " = @p" + liststFieldNamesWithoutKeyIDName.Count().ToString();
            var transactionOptions = new FbTransactionOptions() {
                TransactionBehavior = FbTransactionBehavior.Concurrency | FbTransactionBehavior.Wait
            };
            FbTransaction fbTransaction = fbConn.BeginTransaction(transactionOptions);
            using (FbCommand fbCmd = new FbCommand(stCmd, fbConn, fbTransaction)) {
                for (int iii = 0; iii < liststFieldNamesWithoutKeyIDName.Count(); iii++) {
                    string stPlaceHolder = "@p" + (iii).ToString();
                    string stValue = liststFieldModifiedValuesWithoutKeyIDValue[iii];
                    fbCmd.Parameters.AddWithValue(stPlaceHolder, stValue);
                }
                int iKeyIDValue = Convert.ToInt32(stKeyIDValue);
                fbCmd.Parameters.AddWithValue("@p" + liststFieldNamesWithoutKeyIDName.Count().ToString(), iKeyIDValue);
                fbCmd.ExecuteNonQuery();
                fbTransaction.Commit();
            }
        }
    }
    catch (Exception ex) {
        boErrorFlag = true;
        MessageBox.Show(ex.Message);
    }
}
```

```
    return boErrorFlag;
} //boModifyRecordContentUsingKeyIDValue
//-----
//-----
static private void ParseStringToListst(string stValue,
                                         char cParse,
                                         List<string> liststStringParsed)
{
    string[] astValue = stValue.Split(cParse);
    for (Int16 iii = 0; iii < astValue.Count(); iii++)
        liststStringParsed.Add(astValue[iii]);
} //ParseStringToListst
//-----
```