

Introduzione

Questa TN descrive come gestire un database in SQLite con Wonderware System Platform, ovvero costruisce un esempio su come leggere da un file con estensione db.

Versioni

Quanto descritto è stato verificato con la versione 2020 R2 SP1 (20.1.100) installata su Windows Server 2019 e 2023 P01 (23.0.001) installata su Windows Server 2022.

Cosa sono i file .db

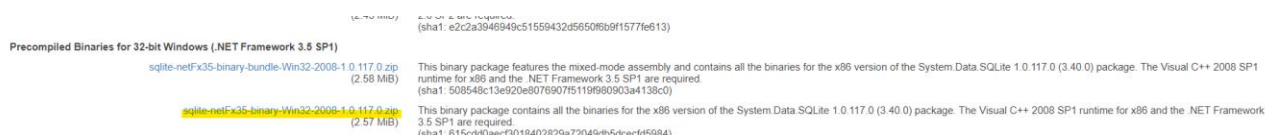
I dispositivi mobili, come i telefoni cellulari Android e iOS, memorizzano una serie di informazioni in file di database DB. Ad esempio, un file DB può contenere i contatti, le informazioni sugli SMS o altri dati del dispositivo o dell'applicazione. I file DB sono generalmente salvati in un formato di database SQLite, ma sono anche generalmente bloccati o crittografati, in modo da non poter accedere o modificare i dati.

Un esempio di file db dell'ambiente Wonderware è il file LicenseManagerData.db che contiene informazioni sulle licenze XML importate sul License Manager.

Procedura

Di seguito la procedura per costruire il nostro esempio:

1. Per prima cosa scarichiamo il pacchetto evidenziato in foto dal link <https://system.data.sqlite.org/index.html/doc/trunk/www/downloads.wiki>



2. Sblocchiamo il file zip subito dopo il download (tasto destro -> proprietà -> annulla blocco).
3. A questo punto importiamo il file System.Data.SQLite.dll nella galaxy entrando nell'IDE e poi cliccando su Galaxy > Import > Script Function Library.
4. Copiamo i file System.Data.SQLite.dll e System.Data.SQLite.dll.config nei seguenti percorsi:

- a. C:\Program Files (x86)\Common Files\Archestra

- b. C:\Program Files (x86)\Common Files\Archestra\Services
- c. C:\Program Files (x86)\Archestra\Framework\Bin
- d. C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Data.SQLite\v4.0_1.0.117.0__db937bc2d44ff139 (da System.Data.SQLite dobbiamo creare le cartelle).

A questo punto possiamo leggere il nostro file db con uno script simile a questo di seguito

```
dim connection as System.Data.SQLite.SQLiteConnection;
dim connString as string;
dim query as string;
dim command as System.Data.SQLite.SQLiteCommand;
dim reader as System.Data.SQLite.SQLiteDataReader;

try

    connString = "Data Source=c:\myDbFile.db;Version=3";
    connection = new System.Data.SQLite.SQLiteConnection(connString);
    connection.Open();
    LogWarning("connessione aperta");

    'SELECT
    try

        query = "Select * from myDbTable";
        command = new System.Data.SQLite.SQLiteCommand(query,connection);
        reader = command.ExecuteReader();
        LogWarning("Inizio Lettura");
        while reader.Read()

            LogMessage(reader("myDbField"));

        endwhile;
        reader.Close();
        LogWarning("Fine Lettura");
    catch

        LogError(error);
    endtry;

    connection.Close();
    LogWarning("connessione chiusa");
```

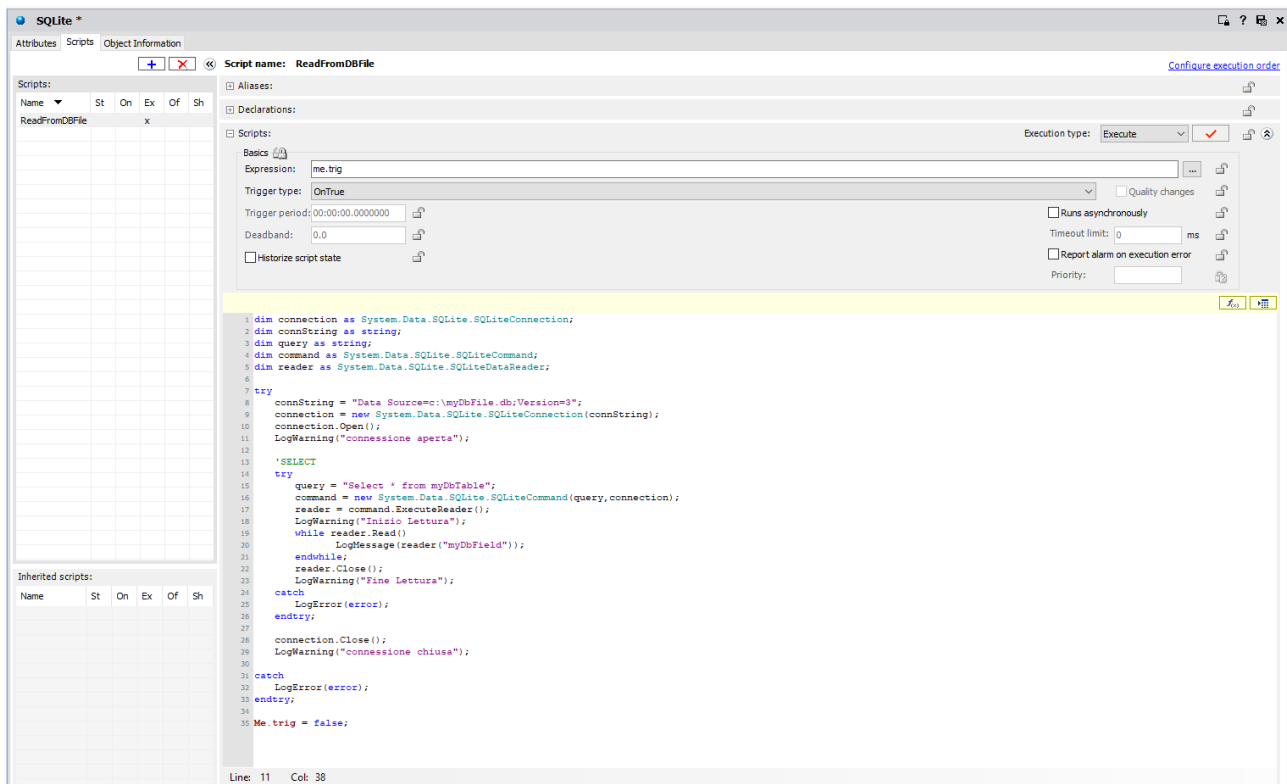
catch

LogError(error);

endtry;

Me.trig = false;

Lo script è inserito in un oggetto UserDefined con un attributo trig che viene valorizzato a false a fine script. Lo script viene eseguito sull'OnTrue dell'espressione me.trig



Nel nostro caso il risultato della query viene scritto nel log.

Autore: F. Pastore

Disclaimer

Il presente documento è fornito a scopo di esempio e non sostituisce la documentazione AVEVA. L'applicazione di quanto contenuto, in un preciso ambito applicativo, deve essere sempre validata da un tecnico Wonderware. La documentazione rilasciata da AVEVA resta il riferimento tecnico ufficiale da seguire: softwaresupport.aveva.com. Wonderware Italia non si assume la responsabilità di un'applicazione scorretta di questo documento.