

Data-On-Demand ListBox Example Guide

The Data-On-Demand ListBox comes with several examples.

DOD Example.rb

The DOD Example.rb project demonstrates the differences between setting up and using the Data-On-Demand ListBox vs. the standard REALbasic ListBox with arrays as a backend. The program starts with 5,000 rows, but you can use the **Make New DB** button to test different sizes. A word of warning: If you create more than 10,000 rows and attempt to sort the standard ListBox, be prepared to wait.

This project also shows how to implement editing of cells, alternate row coloring, and double-click events.

RDB Example.rb

The RDB Example.rb project shows how the Data-On-Demand ListBox can be used with a standard REALbasic database. It comes with a custom class called “My_RecordSet” that you can use in your own projects. My_RecordSet implements random access to a RecordSet.

Start by importing records from the “Import Zip Code Data.txt” file. For more records, import the same file repeatedly. The database files will be created automatically in the same folder as the project.

This main window will show the data and the time it takes to search, sort and load the data. You can alternate between letting the Data-On-Demand ListBox sort the data and using a SQL ORDER BY. Note that, if the data is sorted, the load time will reflect the time to sort the data.

VDB Example.rb

The VDB Example.rb project shows how the Data-On-Demand ListBox can be used with a Valentina v.2.0 database. Valentina v.2.0 is a high-speed database made by Paradigma Software (<http://www.paradigmasoft.com>). The project requires Valentina v.2.0.3, sold separately.

The project comes with a custom class called “My_VIterator” that you can use in your own projects. My_VIterator adds some features to the standard Valentina VIterator class to simplify its use.

Start by importing records from the “Import Zip Code Data.txt” file. For more records, import the same file repeatedly. The database files will be created automatically in the same folder as the project.

This main window will show the data and the time it takes to search, sort and load the data. You can alternate between letting the Data-On-Demand ListBox sort the data and using a Valentina’s native sort. Note that, if the data is sorted, the load time will reflect the time to sort the data.

MacTechnologies Consulting and Kem Tekinay, the author of the Data-On-Demand ListBox, are in no way affiliated with Paradigma Software.

SQLite Example.rb

The SQLite Example.rb project shows how the Data-On-Demand ListBox can be used with a SQLite database as implemented through the SQLitePlugin provided by SQLabs (<http://www.sqlabs.net>). The project requires the SQLitePlugin or the SQLitePluginPro, sold separately and to be included with REALbasic 2005 and later.

The project comes with a custom class called "My_RecordSet" that you can use in your own projects. My_RecordSet implements random access to a RecordSet.

Start by importing records from the "Import Zip Code Data.txt" file. For more records, import the same file repeatedly. The database files will be created automatically in the same folder as the project.

This main window will show the data and the time it takes to search, sort and load the data. You can alternate between letting the Data-On-Demand ListBox sort the data and using a SQL ORDER BY. Note that, if the data is sorted, the load time will reflect the time to sort the data.

MacTechnologies Consulting and Kem Tekinay, the author of the Data-On-Demand ListBox, are in no way affiliated with SQLabs.

REALSQLDB Example.rb

The REALSQLDB Example.rb project shows how the Data-On-Demand ListBox can be used with a REALSQLDatabase, included with REALbasic 2005 and later.

The project comes with a custom class called "My_RecordSet" that you can use in your own projects. My_RecordSet implements random access to a RecordSet.

Start by importing records from the "Import Zip Code Data.txt" file. For more records, import the same file repeatedly. The database files will be created automatically in the same folder as the project.

This main window will show the data and the time it takes to search, sort and load the data. You can alternate between letting the Data-On-Demand ListBox sort the data and using a SQL ORDER BY. Note that, if the data is sorted, the load time will reflect the time to sort the data.