HTMakeL Database Merge Help Contents

The Contents lists Help topics available for HTMakeL.   Use the scroll bar to see entries not currently visible in the Help window.

Table of Contents

Introduction
Usage
Limitations
Tips

Introduction

With the advent of the World Wide Web, the ability to disseminate data to a wide audience is tremendous.   HTML pages can be created with a wide variety of formats and looks.   One of the major obstacles is getting data out of a local database and into an HTML document. HTMakeL is designed to perform that task.   One creates a standardized document, HTMakeL can then be used to place the data in the document or will create multiple pages representing different records or catagories.   The base document can use any of the HTML standards by any party and be created with any authoring tool, HTMakeL simply copies the original document to a new document while adding data to the new document.   All original formatting is preserved.

Usage

HTMakeL Database Merge takes any ascii or ansi text file and merges data from an ODBC data source with the text file to create a new text file.   HTML documents are typically text files capable of such merges.   Although the coding and browsers might give HTML documents a unique look, the underlying file is plain text.

Requirements

Installing HTMakeL

Using HTMakeL

Advanced Commands

Requirements

In order to run HTMakeL your system should have the following minimum components:

Hardware

Hardware required for the ODBC source, the database engine(if any) and the operating system.

Software

Windows 95 or Windows NT for the 32 bit version, Windows 3.0 or above for the 16 bit version.

ODBC Source

You must have your own ODBC drivers for the data and your system must be set up to run ODBC. **HTMakeL does not provide any ODBC files.**

Installing HTMakeL

HTMakeL comes with this help file, the executable, and some example files.   The user can determine where to put HTMakeL.   As the user must enter file names, placing HTMakeL in the same directory as the source file can facilitate using HTMakeL.   No drivers other than the users ODBC drivers are required for HTMakeL.

Using HTMakeL

HTMakeL takes an ascii or ansi text file (the source file) and merges data from an ODBC data source into a new file (the target file). The requirements are:

1) The first part of the source file must contain [ODBC String]. The ODBC String can contain the minimum [ODBC;] in which case ODBC will prompt for other information. Additional information in the string can be DSN=Data Source Name, UID=User ID, PWD=Password. Thus, the connect string can read [ODBC;DSN=Cars;UID=admin;PWD=;].

2) The second part of the source file must contain {TABLE=Table Name}. This should be the name of the recordset containing the data. This portion is case sensitive. Please note that a SQL statement will not work, while a query will work depending on the ODBC data source. As an example {TABLE=Cars} is used. Additionally an ORDER BY and/or WHERE Clause can be included in this portion. Each portion should be seperated by a ;. As an example {TABLE=Cars;ORDER BY [Number];WHERE [Model] = P0000001;} can be used. Again, please note the case sensitivity.

3) Fields can be inserted anywhere within the document by placing them within brackets such as [Field_Name]. For this reason, [and] can not be used in the source document unless it is for HTMakeL. HTMakeL will interpret these marks as fields and attempt to update them. Field names should be considered case sensitive.

4) Enter the name of the Source file, the name can be full path or relative path and the name of the Target file which can not be the same as the Source file. Users of the 16 bit version will be required to use the 8.3 naming structure for the target file.

5) Click process.

Advanced Commands

HTMakeL Database Merge gives you flexibility to create multiple pages, use Headers and footers to create category pages as well as the ability to go through the recordset and create lists.   Additionally, command line usage helps to automate the use of HTMakeL.   The following enhancements can be made to the source file and target file name to achieve the desired look and pages.

Repeating Information

Headers and Footers

Multiple Files

Command Line Usage

Repeating Information

HTMakeL Database Merge gives you flexibility to create lists through repeating information. Please see the samples pages carss8.htm and mods8.htm for use of the [START] and [STOP] commands. The requirements for creating repeating records are:

1. Start the repeating portion of the page with the [START] command.
2. End the repeating portion of the page with the [STOP] command.
3. The area between the [START] and [STOP] is the information that is repeated. This will be repeated for all records in the table.
4. The area preceeding the [START] command acts as a page header.
5. The area following the [STOP] command acts as a page footer.

Headers and Footers

HTMakeL Database Merge gives you flexibility to category pages and more advanced lists using the {HEADER=[field_name]} and {FOOTER} commands. The requirements for using Headers can be shown in the carss8.htm and mods8.htm sample pages. The user may use multiple headers and footers.

1. The [START] command must preceed the first Header.
2. The [STOP] command must follow the last Footer.
3. Enter {HEADER=[field_name]} in the source file following the intended header section.
4. Enter {FOOTER} preceeding the corresponding footer section.
5. The use of {HEADER=[field_name]} is required to use {FOOTER}
6. The use of {FOOTER} is not required to use {HEADER=[field_name]}, but it is recommended. In the case of mulitple Headers, the first footer matches to the last header and so on from inside out. The use of different numbers of headers and footers may result in unintended looks.
7. The area between {HEADER=[field_name]} and {FOOTER} will be repeating.
8. The fields included by Headers will preceed the ORDER BY fields in the final table. Fields included by Headers should not be mentionned in the ORDER BY clause.

Multiple Files

HTMakeL Database Merge gives you flexibility to create multiple pages, even on category pages.   The sample pages of mods8.htm and inds8.htm demonstrate examples of multiple pages.   The mods8.htm provides an example of a category page in which multiple files are created.   Creating mulitple files is done by:

Use [field_name] in the target file name, i.e in mods8.htm, the target name is [Model8].htm. Thus a new file will be created for each distinct record in the [Model8] column.

The [field_name] in the target file name will preceed the Header field names and the [ORDER BY clause in the final grouping of the table.   Headers should not be created for the same field as included in the target file name as the area preceeding the [START} command will serve the same purpose.   The ORDER BY clause should not include the same field as is included in the target file name as this would be redundant.

16 bit users please note that the 8.3 file requirement may pose a problem.   Users of this version should ensure that no records in the column should be longer than 8 charactors. The [Model8] column in the sample database was created for this purpose.

Command Line Usage

HTMakeL Database Merge can be run from the command line.   This is usefull in automating the use of HTMakeL.   The included page bat8.htm shows the file used to create all samle pages.   This usage is subject to the following limitations:

Command Line is HTMakeL.exe source_file target_file [-o]
Where source_file is required and should be the name of the source file.
Where target_file is required and can include [field_name} for multiple pages.
Where -o is optional and its use will prevent prompts to the user about overwriting files.
Files will be overwritten automatically.   Use this option carefully.

The ODBC connect parameters must be complete in either ODBC admin or the source file to use the command lines.   The error Invalid ODBC string will occur if this is not the case.
Please update ODBC prior to running the experimental bat file.

Limitations

HTMakeL Database Merge has the following limitations:

1) Memo fields can be a maximum of 32,500 bytes. HTMakeL will not run if the any of the data is longer.
2) The Source file can be a maximum of 32,500 bytes. This should not be a problem except for potentially recurring uses. See Headers on Tips and Tricks.
3) Binary fields will not insert data.
4) SQL strings can not be used as the table name.
5) [START] can not be the name of a field if the [START] command is not used.
6) [STOP] can not be the name of a field if the [START] command is used.
7) The source document can not contain [or] except in relation to HTMakeL.

HTMakeL has been tested with the following data sources:

1) Microsoft Access 2.0 and Microsoft Access 7.0
2) Watcom SQL 4.0
3) dBase 4.0

HTMakeL has been tested with recordsets of over 950 records and created pages as long as 255,000 bytes.

Tips and Tricks

1) Create the page originally with any HTML editor, using sample data where the fields will go. Remember to format the page to look right will multiple records, i.e. place <UL> before the [START] and the </UL> after the [STOP]. One will most likely want a <BR> or <P> prior to the [STOP] additionally. Add in links and graphics using the data in the records where possible - See the Cars page for an example. Fields can be placed anywhere including within the Header marks for the HTML document, so be creative.
2) Plan your files and their names prior to using HTMakeL, the more things that are done with the data such as file names, links and graphics, the more use one can make of HTMakeL.
3) Certain fields may not produce the desired format as expected from the database. HTMakeL uses the raw format of ODBC. Therefor, with some databases, consider creating a query based on the table that formats all data into char or string data formats. This should provide the desired look in the final document. A query has been included with the cars.mdb database to demonstrate this. In order to use this query with the sample pages, the field names will need to be updated by replacing [Price] with [Cost] and [DateUpdated] with [Date].
4) If using in conjunction with a server to respond to user requests consider keeping the database engine running to speed up response time.
5) If the table shows No Records Returned try removing any WHERE, or ORDER BY Clause first, the remove HEADERS and do not create multiple pages. If there is still a No Records Returned, check your ODBC connection with some other source. Otherwise insert first the multiple pages command, if it works then attempt the Headers, and finally add back the WHERE or ORDER BY requirements. The error can usually be pinpointed by this process.
6) Ensure that for each <UL> tag there is a </UL> tag.
7) Finally, be creative and have fun.