LIBPQ wrapper c++

Generated by Doxygen 1.8.11

Contents

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

Namespace Documentation

4.1 libpqwrp Namespace Reference

Classes

- class libPQwrp

    gestion parametre sql

Functions

- constexpr unsigned long long int HashStringToInt (const char ∗str, unsigned long long int hash=0)

    a function to make your alphanumeric switch

Variables

- static bool fetch0
- static bool update0
- static bool select0
- static bool delete0
- static bool insert0
- static bool fetchEOF

4.1.1 Function Documentation

4.1.1.1 constexpr unsigned long long int libpqwrp::HashStringToInt (const char ∗ *str,* unsigned long long int *hash =* 0)

a function to make your alphanumeric switch

Definition at line 70 of file libpqwrp.hpp.

4.1.2 Variable Documentation

4.1.2.1 bool libpqwrp::delete0 [static]

Definition at line 62 of file libpqwrp.hpp.

4.1.2.2 bool libpqwrp::fetch0 [static]

$$ fonction sql de libpq $$

Definition at line 59 of file libpqwrp.hpp.

4.1.2.3 bool libpqwrp::fetchEOF [static]

Definition at line 64 of file libpqwrp.hpp.

4.1.2.4 bool libpqwrp::insert0 [static]

Definition at line 63 of file libpqwrp.hpp.

4.1.2.5 bool libpqwrp::select0 [static]

Definition at line 61 of file libpqwrp.hpp.

4.1.2.6 bool libpqwrp::update0 [static]

Definition at line 60 of file libpqwrp.hpp.

Chapter 5

Class Documentation

5.1 libpqwrp::libPQwrp Class Reference

gestion parametre sql

```
#include <libpqwrp.hpp>
```

Public Member Functions

- libPQwrp ()
- ∼libPQwrp ()
- void connectDB (std::string info)

  connect to the database
- void qexec (std::string sql)

  connect to the database
- void query (std::string sql, bool binary=true)

  PQexec.
- char ∗ fetch (int row, int column)

  PQquery.
- double fetchDbl (int row, int column)

  get the value of the row and the column
- int fetchInt (int row, int column)

  get the value of the row and the column
- int nfield (std::string field)

  get the value of the row and the column
- const char ∗ cfield (int nfield)

  number column of the field
- int countfield ()

  name of the field
- int countrow ()

  number of columns
- bool is_Table (const char ∗table)

  number of rows
- void closeDB ()

  if exist table of the database

- void begin ()

  close to the database
- void commit ()

  start transction for commit
- void rollback ()

  commit for transaction init
- void end ()

  rollbck for transaction init
- void clearRes ()

  end transaction for init
- void savpoint ()

  clear PGresult
- void savpointRollback ()

  point de sauvegarde
- void savpointRelease ()

  roolback base savepoint
- void fetchAll (std::string, std::string cursor="mycursor")

  delete savepoint
- void opensql (std::string sql, std::string cursor="mycursor")

  fetch ALL include requete
- void fechsql (std::string cursor="mycursor")

  query for fetchsql record / record
- template<typename Head , typename... Args>
  std::string prepare (std::string const &format, Head &&head, Args &&...args)

  fetch record use openSQL
- std::stringstream result ()

  formatting the query with parameters
- char ∗ DoubleToChar (double _X_, unsigned _precision_=0)

  out buffer;

Public Attributes

- int rows

  format without parameters causes an error
- int cols

  initialise par les requetes
- int rown

  initialise par les requetes
- int coln

  variable de traitement
- PGresult ∗ res

  variable de traitement

Protected Member Functions

- unsigned count_format_specifiers (std::string const &format)

  contrôle PGresult
- void prepare (std::string const &format)

  count the number of parameters in the format

Private Attributes

- PGconn ∗ conn
- std::stringstream sqlstream
- bool first = true
- std::string fin_stage
- bool clean

Static Private Attributes

- static constexpr char FORMAT_SPECIFIER = '?'

Friends

- std::istream & operator>> (std::istream &is, char ∗t)

  Double to char.

5.1.1 Detailed Description

gestion parametre sql

$$ prepare sql formatage avec template variadique etc........ $$

Definition at line 90 of file libpqwrp.hpp.

5.1.2 Constructor & Destructor Documentation

5.1.2.1 libpqwrp::libPQwrp ( ) [inline]

Definition at line 129 of file libpqwrp.hpp.

5.1.2.2 libpqwrp::libPQwrp::∼libPQwrp ( ) [inline]

Definition at line 139 of file libpqwrp.hpp.

5.1.3 Member Function Documentation

5.1.3.1 void libpqwrp::libPQwrp::begin ( )

close to the database

start transction for commit

Definition at line 478 of file libpqwrp.hpp.

5.1.3.2 const char * libpqwrp::libPQwrp::cfield (int *nfield*)

number column of the field

name of the field

Definition at line 422 of file libpqwrp.hpp.

5.1.3.3 void libpqwrp::libPQwrp::clearRes ( )

end transaction for init

clear PGresult

Definition at line 505 of file libpqwrp.hpp.

5.1.3.4 void libpqwrp::libPQwrp::closeDB ( )

if exist table of the database

close to the database

Definition at line 470 of file libpqwrp.hpp.

5.1.3.5 void libpqwrp::libPQwrp::commit ( )

start transction for commit

commit for transaction init

Definition at line 484 of file libpqwrp.hpp.

5.1.3.6 void libpqwrp::libPQwrp::connectDB (std::string *info*)

connect to the database

init PGconn

init PGresult

Definition at line 312 of file libpqwrp.hpp.

5.1.3.7 unsigned libpqwrp::libPQwrp::count_format_specifiers (std::string const & *format*) [protected]

contrôle PGresult

$$ process $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

Definition at line 220 of file libpqwrp.hpp.

5.1.3.8 int libpqwrp::libPQwrp::countfield ( )

name of the field

number of columns

Definition at line 434 of file libpqwrp.hpp.

5.1.3.9 int libpqwrp::libPQwrp::countrow ( )

number of columns

number of rows

Definition at line 441 of file libpqwrp.hpp.

5.1.3.10 char ∗ libpqwrp::libPQwrp::DoubleToChar (double *_X_*, unsigned *_precision_* = `0`)

out buffer;

Double to char.

Definition at line 660 of file libpqwrp.hpp.

5.1.3.11 void libpqwrp::libPQwrp::end ( )

rollbck for transaction init

end transaction for init

Definition at line 498 of file libpqwrp.hpp.

5.1.3.12 void libpqwrp::libPQwrp::fechsql (std::string *cursor* = `"mycursor"`)

query for fetchsql record / record

fetch record use openSQL read a line record only

end of row or without row

close cursor

no end of row

Definition at line 631 of file libpqwrp.hpp.

5.1.3.13 char ∗ libpqwrp::libPQwrp::fetch (int *row*, int *column*)

PQquery.

get the value of the row and the column

Definition at line 370 of file libpqwrp.hpp.

5.1.3.14 void libpqwrp::libPQwrp::fetchAll (std::string *sql*, std::string *cursor* = `"mycursor"`)

delete savepoint

fetch ALL include requete

———— process complex

lecture full rc ??? memory

end of row or without row

close cursor

no end of row

Definition at line 538 of file libpqwrp.hpp.

5.1.3.15 double libpqwrp::libPQwrp::fetchDbl (int *row*, int *column*)

get the value of the row and the column

Definition at line 383 of file libpqwrp.hpp.

5.1.3.16 int libpqwrp::libPQwrp::fetchInt (int *row*, int *column*)

get the value of the row and the column

Definition at line 396 of file libpqwrp.hpp.

5.1.3.17 bool libpqwrp::libPQwrp::is_Table (const char ∗ *table*)

number of rows

if exist table of the database false for return value count

Definition at line 447 of file libpqwrp.hpp.

5.1.3.18 int libpqwrp::libPQwrp::nfield (std::string *field*)

get the value of the row and the column

number column of the field

Definition at line 409 of file libpqwrp.hpp.

5.1.3.19 void libpqwrp::libPQwrp::opensql (std::string *sql,* std::string *cursor =* "mycursor")

fetch ALL include requete

query for fetchsql record / record end of row or without row

close cursor

no end of row

read a line record only

Definition at line 588 of file libpqwrp.hpp.

5.1.3.20 void libpqwrp::libPQwrp::prepare (std::string const & *format*) [protected]

count the number of parameters in the format

Definition at line 239 of file libpqwrp.hpp.

5.1.3.21 template<typename Head , typename... Args> std::string libpqwrp::libPQwrp::prepare (std::string const & *format,* Head && *head,* Args &&... *args*)

fetch record use openSQL

formate la requete

Definition at line 248 of file libpqwrp.hpp.

5.1.3.22 void libpqwrp::libPQwrp::qexec (std::string *sql*)

connect to the database

PQexec.

Definition at line 328 of file libpqwrp.hpp.

5.1.3.23 void libpqwrp::libPQwrp::query (std::string *sql,* bool *binary =* true)

PQexec.

PQquery.

Definition at line 343 of file libpqwrp.hpp.

5.1.3.24 std::stringstream libpqwrp::libPQwrp::result ( )

formatting the query with parameters

retrieve values on the fields

Definition at line 280 of file libpqwrp.hpp.

5.1.3.25 void libpqwrp::libPQwrp::rollback ( )

commit for transaction init

rollbck for transaction init

Definition at line 491 of file libpqwrp.hpp.

5.1.3.26 void libpqwrp::libPQwrp::savpoint ( )

clear PGresult

point de sauvegarde

Definition at line 511 of file libpqwrp.hpp.

5.1.3.27 void libpqwrp::libPQwrp::savpointRelease ( )

roolback base savepoint

delete savepoint

Definition at line 525 of file libpqwrp.hpp.

5.1.3.28 void libpqwrp::libPQwrp::savpointRollback ( )

point de sauvegarde

roolback base savepoint

Definition at line 518 of file libpqwrp.hpp.

5.1.4 Friends And Related Function Documentation

5.1.4.1 std::istream& operator>> (std::istream & *is,* char ∗ *t*) [friend]

Double to char.

retrieve char∗ for stringstream dumy

Definition at line 197 of file libpqwrp.hpp.

5.1.5 Member Data Documentation

5.1.5.1 bool libpqwrp::libPQwrp::clean [private]

Definition at line 105 of file libpqwrp.hpp.

5.1.5.2 int libpqwrp::libPQwrp::coln

variable de traitement

Definition at line 124 of file libpqwrp.hpp.

5.1.5.3 int libpqwrp::libPQwrp::cols

initialise par les requetes

Definition at line 121 of file libpqwrp.hpp.

5.1.5.4 PGconn* libpqwrp::libPQwrp::conn [private]

Definition at line 94 of file libpqwrp.hpp.

5.1.5.5 std::string libpqwrp::libPQwrp::fin_stage [private]

Definition at line 103 of file libpqwrp.hpp.

5.1.5.6 bool libpqwrp::libPQwrp::first = true [private]

Definition at line 101 of file libpqwrp.hpp.

5.1.5.7 constexpr char libpqwrp::libPQwrp::FORMAT_SPECIFIER = '?' [static], [private]

Definition at line 97 of file libpqwrp.hpp.

5.1.5.8 PGresult* libpqwrp::libPQwrp::res

variable de traitement

Definition at line 127 of file libpqwrp.hpp.

5.1.5.9 int libpqwrp::libPQwrp::rown

initialise par les requetes

Definition at line 123 of file libpqwrp.hpp.

5.1.5.10 int libpqwrp::libPQwrp::rows

format without parameters causes an error

Definition at line 120 of file libpqwrp.hpp.

5.1.5.11 std::stringstream libpqwrp::libPQwrp::sqlstream `[private]`

Definition at line 99 of file libpqwrp.hpp.

The documentation for this class was generated from the following file:

- libpqwrp.hpp

Chapter 6

File Documentation

6.1 libpqwrp.hpp File Reference

```
#include <string.h>
#include <cstdio>
#include <cstdlib>
#include <string>
#include <iostream>
#include <ostream>
#include <sstream>
#include <iomanip>
#include <stdexcept>
#include <libpq-fe.h>
#include <typeinfo>
```

Classes

- class libpqwrp::libPQwrp

  gestion parametre sql

Namespaces

- libpqwrp

Macros

- #define NAMEOF(variable) ((void)variable, #variable)
- #define DeLiMiTaTioN '∼'

Functions

- constexpr unsigned long long int libpqwrp::HashStringToInt (const char ∗str, unsigned long long int hash=0)

  a function to make your alphanumeric switch

Variables

- static bool libpqwrp::fetch0
- static bool libpqwrp::update0
- static bool libpqwrp::select0
- static bool libpqwrp::delete0
- static bool libpqwrp::insert0
- static bool libpqwrp::fetchEOF

6.1.1 Macro Definition Documentation

6.1.1.1 #define DeLiMiTaTioN '∼'

Definition at line 79 of file libpqwrp.hpp.

6.1.1.2 #define NAMEOF(*variable*) ((void)variable, #variable)

Definition at line 76 of file libpqwrp.hpp.

Index