LIBPQ wrapper c++

Generated by Doxygen 1.8.11

Contents

1 Namespace Index **1**

1.1 Namespace List 1

2 Class Index **3**

2.1 Class List 3

3 File Index **5**

3.1 File List 5

4 Namespace Documentation **7**

4.1 libpqsql Namespace Reference 7

4.1.1 Function Documentation 8

4.1.1.1 begin(PGconn ∗conn) 8

4.1.1.2 cfield(PGresult ∗res, int nfield) 8

4.1.1.3 closeDB(PGconn ∗conn) 8

4.1.1.4 commit(PGconn ∗conn) 9

4.1.1.5 connectDB(std::string info) 9

4.1.1.6 countfield(PGresult ∗res) 9

4.1.1.7 countrow(PGresult ∗res) 9

4.1.1.8 DoubleToChar(double _X_, unsigned _precision_=0) 9

4.1.1.9 end(PGconn ∗conn) 9

4.1.1.10 fechsql(PGconn ∗conn, std::string cursor=""mycursor"") 9

4.1.1.11 fetch(PGresult ∗res, int row, int column) 10

4.1.1.12 fetchAll(PGconn ∗conn, std::string sql, std::string cursor=""mycursor"") 10

4.1.1.13 fetchDbl(PGresult ∗res, int row, int column) 10

4.1.1.14 fetchInt(PGresult *res, int row, int column) . . . 10
4.1.1.15 HashStringToInt(const char *str, unsigned long long int hash=0) . . . 10
4.1.1.16 is_Table(PGconn *conn, const char *table) . . . 10
4.1.1.17 nfield(PGresult *res, std::string field) . . . 11
4.1.1.18 opensql(PGconn *conn, std::string sql, std::string cursor=""mycursor"") . . . 11
4.1.1.19 operator>>(std::istream &is, char *t) . . . 11
4.1.1.20 qexec(PGconn *conn, std::string sql) . . . 11
4.1.1.21 query(PGconn *conn, std::string sql, bool binary=true) . . . 11
4.1.1.22 rollback(PGconn *conn) . . . 11
4.1.1.23 savpoint(PGconn *conn) . . . 11
4.1.1.24 savpointRelease(PGconn *conn) . . . 12
4.1.1.25 savpointRollback(PGconn *conn) . . . 12
4.1.2 Variable Documentation . . . 12
4.1.2.1 delete0 . . . 12
4.1.2.2 fetch0 . . . 12
4.1.2.3 fetchEOF . . . 12
4.1.2.4 insert0 . . . 12
4.1.2.5 records . . . 12
4.1.2.6 select0 . . . 12
4.1.2.7 update0 . . . 12

5 Class Documentation **13**

5.1 libpqsql::inforowcol Struct Reference . . . 13
5.1.1 Detailed Description . . . 13
5.1.2 Member Data Documentation . . . 13
5.1.2.1 cols . . . 13
5.1.2.2 rows . . . 13
5.1.2.3 Xcol . . . 14
5.1.2.4 Xrow . . . 14
5.2 libpqsql::QueryPrint Class Reference . . . 14
5.2.1 Detailed Description . . . 15
5.2.2 Member Function Documentation . . . 15
5.2.2.1 count_format_specifiers(std::string const &format) . . . 15
5.2.2.2 prepare(std::string const &format) . . . 15
5.2.2.3 prepare(std::string const &format, Head &&head, Args &&...args) . . . 15
5.2.2.4 result(PGresult *res) . . . 15
5.2.2.5 resultctrl(PGresult *res) . . . 15
5.2.3 Member Data Documentation . . . 15
5.2.3.1 fin_stage . . . 15
5.2.3.2 first . . . 16
5.2.3.3 FORMAT_SPECIFIER . . . 16
5.2.3.4 sql . . . 16

6 File Documentation **17**

6.1 libpqwrp.hpp File Reference 17

6.1.1 Macro Definition Documentation 19

6.1.1.1 DeLiMiTaTioN 19

6.1.1.2 NAMEOF 19

Index **21**

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

libpqsql 7

Chapter 2

Class Index

2.1 Class List

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

libpqsql::inforowcol 13
libpqsql::QueryPrint
Gestion parametre sql 14

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

libpqwrp.hpp 17

Chapter 4

Namespace Documentation

4.1 libpqsql Namespace Reference

Classes

- struct inforowcol
- class QueryPrint

    gestion parametre sql

Functions

- std::istream & operator>> (std::istream &is, char ∗t)
- char ∗ DoubleToChar (double _X_, unsigned _precision_=0)

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

    a function to make your alphanumeric switch
- PGconn ∗ connectDB (std::string info)

    $$
- bool qexec (PGconn ∗conn, std::string sql)

    PQexec.
- PGresult ∗ query (PGconn ∗conn, std::string sql, bool binary=true)

    PQquery.
- char ∗ fetch (PGresult ∗res, int row, int column)

    get the value of the row and the column
- double fetchDbl (PGresult ∗res, int row, int column)

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

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

    column of the field
- const char ∗ cfield (PGresult ∗res, int nfield)

    name of the field
- int countfield (PGresult ∗res)

    number of columns
- int countrow (PGresult ∗res)

number of rows

- bool is_Table (PGconn ∗conn, const char ∗table)

  if exist table of the database ->fasle = does not exist
- void closeDB (PGconn ∗conn)

  close to the database
- void begin (PGconn ∗conn)
- void commit (PGconn ∗conn)
- void end (PGconn ∗conn)
- void rollback (PGconn ∗conn)
- void savpoint (PGconn ∗conn)
- void savpointRollback (PGconn ∗conn)
- void savpointRelease (PGconn ∗conn)
- PGresult ∗ fetchAll (PGconn ∗conn, std::string sql, std::string cursor="mycursor")

  fetch ALL include requete
- PGresult ∗ opensql (PGconn ∗conn, std::string sql, std::string cursor="mycursor")

  query for fetchsql record / record
- PGresult ∗ fechsql (PGconn ∗conn, std::string cursor="mycursor")

  fetch record use openSQL

Variables

- static bool fetch0
- static bool update0
- static bool select0
- static bool delete0
- static bool insert0
- static bool fetchEOF
- struct libpqsql::inforowcol records

4.1.1 Function Documentation

4.1.1.1 void libpqsql::begin (PGconn ∗ *conn*)

full

Definition at line 433 of file libpqwrp.hpp.

4.1.1.2 const char∗ libpqsql::cfield (PGresult ∗ *res,* int *nfield*)

name of the field

Definition at line 379 of file libpqwrp.hpp.

4.1.1.3 void libpqsql::closeDB (PGconn ∗ *conn*)

close to the database

Definition at line 423 of file libpqwrp.hpp.

4.1.1.4 void libpqsql::commit (PGconn ∗ *conn*)

full

Definition at line 441 of file libpqwrp.hpp.

4.1.1.5 PGconn∗ libpqsql::connectDB (std::string *info*)

$$$

connect to the database

Definition at line 264 of file libpqwrp.hpp.

4.1.1.6 int libpqsql::countfield (PGresult ∗ *res*)

number of columns

Definition at line 392 of file libpqwrp.hpp.

4.1.1.7 int libpqsql::countrow (PGresult ∗ *res*)

number of rows

Definition at line 399 of file libpqwrp.hpp.

4.1.1.8 char∗ libpqsql::DoubleToChar (double *_X_*, unsigned *_precision_* = 0)

Double to char ----------------------------—.

Definition at line 101 of file libpqwrp.hpp.

4.1.1.9 void libpqsql::end (PGconn ∗ *conn*)

full

Definition at line 448 of file libpqwrp.hpp.

4.1.1.10 PGresult∗ libpqsql::fechsql (PGconn ∗ *conn*, std::string *cursor* = "mycursor")

fetch record use openSQL

read a line record only

end of row or without row

close cursor

no end of row

Definition at line 579 of file libpqwrp.hpp.

4.1.1.11 char* libpqsql::fetch (PGresult * *res,* int *row,* int *column*)

get the value of the row and the column

Definition at line 324 of file libpqwrp.hpp.

4.1.1.12 PGresult* libpqsql::fetchAll (PGconn * *conn,* std::string *sql,* std::string *cursor* = "mycursor")

fetch ALL include requete

——— process complex

lecture full records ??? memory

end of row or without row

close cursor

no end of row

Definition at line 490 of file libpqwrp.hpp.

4.1.1.13 double libpqsql::fetchDbl (PGresult * *res,* int *row,* int *column*)

get the value of the row and the column

Definition at line 337 of file libpqwrp.hpp.

4.1.1.14 int libpqsql::fetchInt (PGresult * *res,* int *row,* int *column*)

get the value of the row and the column

Definition at line 351 of file libpqwrp.hpp.

4.1.1.15 constexpr unsigned long long int libpqsql::HashStringToInt (const char * *str,* unsigned long long int *hash* = 0)

a function to make your alphanumeric switch

Definition at line 123 of file libpqwrp.hpp.

4.1.1.16 bool libpqsql::is_Table (PGconn * *conn,* const char * *table*)

if exist table of the database ->fasle = does not exist

false for return value count

Definition at line 406 of file libpqwrp.hpp.

4.1.1.17 int libpqsql::nfield (PGresult * *res,* std::string *field*)

column of the field

Definition at line 365 of file libpqwrp.hpp.

4.1.1.18 PGresult* libpqsql::opensql (PGconn * *conn,* std::string *sql,* std::string *cursor* = "mycursor")

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 538 of file libpqwrp.hpp.

4.1.1.19 std::istream& libpqsql::operator>> (std::istream & *is,* char * *t*)

dumy

Definition at line 84 of file libpqwrp.hpp.

4.1.1.20 bool libpqsql::qexec (PGconn * *conn,* std::string *sql*)

PQexec.

Definition at line 279 of file libpqwrp.hpp.

4.1.1.21 PGresult* libpqsql::query (PGconn * *conn,* std::string *sql,* bool *binary* = true)

PQquery.

Definition at line 296 of file libpqwrp.hpp.

4.1.1.22 void libpqsql::rollback (PGconn * *conn*)

full

Definition at line 455 of file libpqwrp.hpp.

4.1.1.23 void libpqsql::savpoint (PGconn * *conn*)

point de sauvegarde

Definition at line 462 of file libpqwrp.hpp.

4.1.1.24 void libpqsql::savpointRelease (PGconn * *conn*)

delete savepoint

Definition at line 476 of file libpqwrp.hpp.

4.1.1.25 void libpqsql::savpointRollback (PGconn * *conn*)

roolback sur savepoint

Definition at line 469 of file libpqwrp.hpp.

4.1.2 Variable Documentation

4.1.2.1 bool libpqsql::delete0 [static]

Definition at line 63 of file libpqwrp.hpp.

4.1.2.2 bool libpqsql::fetch0 [static]

$$$ fonction sql de libpq $$$

Definition at line 60 of file libpqwrp.hpp.

4.1.2.3 bool libpqsql::fetchEOF [static]

Definition at line 65 of file libpqwrp.hpp.

4.1.2.4 bool libpqsql::insert0 [static]

Definition at line 64 of file libpqwrp.hpp.

4.1.2.5 struct libpqsql::inforowcol libpqsql::records

4.1.2.6 bool libpqsql::select0 [static]

Definition at line 62 of file libpqwrp.hpp.

4.1.2.7 bool libpqsql::update0 [static]

Definition at line 61 of file libpqwrp.hpp.

Chapter 5

Class Documentation

5.1 libpqsql::inforowcol Struct Reference

`#include <libpqwrp.hpp>`

Public Attributes

- int rows
- int cols

  initialise par les requetes
- int Xrow

  initialise par les requetes
- int Xcol

  variable de traitement

5.1.1 Detailed Description

Definition at line 67 of file libpqwrp.hpp.

5.1.2 Member Data Documentation

5.1.2.1 int libpqsql::inforowcol::cols

initialise par les requetes

Definition at line 70 of file libpqwrp.hpp.

5.1.2.2 int libpqsql::inforowcol::rows

Definition at line 69 of file libpqwrp.hpp.

5.1.2.3 int libpqsql::inforowcol::Xcol

variable de traitement

Definition at line 73 of file libpqwrp.hpp.

5.1.2.4 int libpqsql::inforowcol::Xrow

initialise par les requetes

Definition at line 72 of file libpqwrp.hpp.

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

- libpqwrp.hpp

5.2 libpqsql::QueryPrint Class Reference

gestion parametre sql

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

Public Member Functions

- template<typename Head , typename... Args>
  std::string prepare (std::string const &format, Head &&head, Args &&...args)
  error not parameter query result
- std::stringstream result (PGresult *res)
  formatting the query with parameters

Protected Member Functions

- unsigned count_format_specifiers (std::string const &format)
- void prepare (std::string const &format)
  count the number of parameters in the format
- void resultctrl (PGresult *res)
  format without parameters causes an error

Private Attributes

- std::stringstream sql
- bool first = true
- std::string fin_stage

Static Private Attributes

- static constexpr char FORMAT_SPECIFIER = '?'

5.2.1 Detailed Description

gestion parametre sql

$$ prepare sql formatage avec template variadique $$

Definition at line 140 of file libpqwrp.hpp.

5.2.2 Member Function Documentation

5.2.2.1 unsigned libpqsql::QueryPrint::count_format_specifiers (std::string const & *format*) [protected]

Definition at line 167 of file libpqwrp.hpp.

5.2.2.2 void libpqsql::QueryPrint::prepare (std::string const & *format*) [protected]

count the number of parameters in the format

Definition at line 186 of file libpqwrp.hpp.

5.2.2.3 template<typename Head , typename... Args> std::string libpqsql::QueryPrint::prepare (std::string const & *format,* Head && *head,* Args &&... *args*)

error not parameter query result

formate la requete

Definition at line 195 of file libpqwrp.hpp.

5.2.2.4 std::stringstream libpqsql::QueryPrint::result (PGresult * *res*)

formatting the query with parameters

retrieve values on the fields

Definition at line 227 of file libpqwrp.hpp.

5.2.2.5 void libpqsql::QueryPrint::resultctrl (PGresult * *res*) [protected]

format without parameters causes an error

5.2.3 Member Data Documentation

5.2.3.1 std::string libpqsql::QueryPrint::fin_stage [private]

Definition at line 147 of file libpqwrp.hpp.

5.2.3.2 bool libpqsql::QueryPrint::first = true [private]

Definition at line 146 of file libpqwrp.hpp.

5.2.3.3 constexpr char libpqsql::QueryPrint::FORMAT_SPECIFIER = '?' [static], [private]

Definition at line 143 of file libpqwrp.hpp.

5.2.3.4 std::stringstream libpqsql::QueryPrint::sql [private]

Definition at line 145 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 <iostream>
#include <ostream>
#include <iomanip>
#include <sstream>
#include <stdexcept>
#include <string>
#include <libpq-fe.h>
#include <cstdlib>
```

Classes

- struct libpqsql::inforowcol
- class libpqsql::QueryPrint

  gestion parametre sql

Namespaces

- libpqsql

Macros

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

Functions

- std::istream & libpqsql::operator>> (std::istream &is, char ∗t)
- char ∗ libpqsql::DoubleToChar (double _X_, unsigned _precision_=0)

  Double to char ---------------------------—.
- constexpr unsigned long long int libpqsql::HashStringToInt (const char ∗str, unsigned long long int hash=0)

  a function to make your alphanumeric switch
- PGconn ∗ libpqsql::connectDB (std::string info)

  $$
- bool libpqsql::qexec (PGconn ∗conn, std::string sql)

  PQexec.
- PGresult ∗ libpqsql::query (PGconn ∗conn, std::string sql, bool binary=true)

  PQquery.
- char ∗ libpqsql::fetch (PGresult ∗res, int row, int column)

  get the value of the row and the column
- double libpqsql::fetchDbl (PGresult ∗res, int row, int column)

  get the value of the row and the column
- int libpqsql::fetchInt (PGresult ∗res, int row, int column)

  get the value of the row and the column
- int libpqsql::nfield (PGresult ∗res, std::string field)

  column of the field
- const char ∗ libpqsql::cfield (PGresult ∗res, int nfield)

  name of the field
- int libpqsql::countfield (PGresult ∗res)

  number of columns
- int libpqsql::countrow (PGresult ∗res)

  number of rows
- bool libpqsql::is_Table (PGconn ∗conn, const char ∗table)

  if exist table of the database ->fasle = does not exist
- void libpqsql::closeDB (PGconn ∗conn)

  close to the database
- void libpqsql::begin (PGconn ∗conn)
- void libpqsql::commit (PGconn ∗conn)
- void libpqsql::end (PGconn ∗conn)
- void libpqsql::rollback (PGconn ∗conn)
- void libpqsql::savpoint (PGconn ∗conn)
- void libpqsql::savpointRollback (PGconn ∗conn)
- void libpqsql::savpointRelease (PGconn ∗conn)
- PGresult ∗ libpqsql::fetchAll (PGconn ∗conn, std::string sql, std::string cursor="mycursor")

  fetch ALL include requete
- PGresult ∗ libpqsql::opensql (PGconn ∗conn, std::string sql, std::string cursor="mycursor")

  query for fetchsql record / record
- PGresult ∗ libpqsql::fechsql (PGconn ∗conn, std::string cursor="mycursor")

  fetch record use openSQL

Variables

- static bool libpqsql::fetch0
- static bool libpqsql::update0
- static bool libpqsql::select0
- static bool libpqsql::delete0
- static bool libpqsql::insert0
- static bool libpqsql::fetchEOF
- struct libpqsql::inforowcol libpqsql::records

6.1.1 Macro Definition Documentation

6.1.1.1 #define DeLiMiTaTioN '∼'

Definition at line 54 of file libpqwrp.hpp.

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

Definition at line 129 of file libpqwrp.hpp.

Index

begin
    libpqsql, 8

cfield
    libpqsql, 8
closeDB
    libpqsql, 8
cols
    libpqsql::inforowcol, 13
commit
    libpqsql, 8
connectDB
    libpqsql, 9
count_format_specifiers
    libpqsql::QueryPrint, 15
countfield
    libpqsql, 9
countrow
    libpqsql, 9

DeLiMiTaTioN
    libpqwrp.hpp, 19
delete0
    libpqsql, 12
DoubleToChar
    libpqsql, 9

end
    libpqsql, 9

FORMAT_SPECIFIER
    libpqsql::QueryPrint, 16
fechsql
    libpqsql, 9
fetch
    libpqsql, 9
fetch0
    libpqsql, 12
fetchAll
    libpqsql, 10
fetchDbl
    libpqsql, 10
fetchEOF
    libpqsql, 12
fetchInt
    libpqsql, 10
fin_stage
    libpqsql::QueryPrint, 15
first
    libpqsql::QueryPrint, 15

HashStringToInt
    libpqsql, 10

insert0
    libpqsql, 12
is_Table
    libpqsql, 10

libpqsql, 7
    begin, 8
    cfield, 8
    closeDB, 8
    commit, 8
    connectDB, 9
    countfield, 9
    countrow, 9
    delete0, 12
    DoubleToChar, 9
    end, 9
    fechsql, 9
    fetch, 9
    fetch0, 12
    fetchAll, 10
    fetchDbl, 10
    fetchEOF, 12
    fetchInt, 10
    HashStringToInt, 10
    insert0, 12
    is_Table, 10
    nfield, 10
    opensql, 11
    operator>>, 11
    qexec, 11
    query, 11
    records, 12
    rollback, 11
    savpoint, 11
    savpointRelease, 11
    savpointRollback, 12
    select0, 12
    update0, 12
libpqsql::QueryPrint, 14
    count_format_specifiers, 15
    FORMAT_SPECIFIER, 16
    fin_stage, 15
    first, 15
    prepare, 15
    result, 15
    resultctrl, 15
    sql, 16

libpqsql::inforowcol, 13
    cols, 13
    rows, 13
    Xcol, 13
    Xrow, 14
libpqwrp.hpp, 17
    DeLiMiTaTioN, 19
    NAMEOF, 19

NAMEOF
    libpqwrp.hpp, 19
nfield
    libpqsql, 10

opensql
    libpqsql, 11
operator>>
    libpqsql, 11

prepare
    libpqsql::QueryPrint, 15

qexec
    libpqsql, 11
query
    libpqsql, 11

records
    libpqsql, 12
result
    libpqsql::QueryPrint, 15
resultctrl
    libpqsql::QueryPrint, 15
rollback
    libpqsql, 11
rows
    libpqsql::inforowcol, 13

savpoint
    libpqsql, 11
savpointRelease
    libpqsql, 11
savpointRollback
    libpqsql, 12
select0
    libpqsql, 12
sql
    libpqsql::QueryPrint, 16

update0
    libpqsql, 12

Xcol
    libpqsql::inforowcol, 13
Xrow
    libpqsql::inforowcol, 14