

LIBPQ wrapper c++

Generated by Doxygen 1.8.11

Contents

1	Class Index	1
1.1	Class List	1
2	Class Documentation	3
2.1	libpqwrp::libPQwrp Class Reference	3
2.1.1	Detailed Description	5
2.1.2	Member Function Documentation	5
2.1.2.1	autocommitOFF()	5
2.1.2.2	autocommitON()	5
2.1.2.3	begin()	6
2.1.2.4	cfield(int nfield)	6
2.1.2.5	clearRes()	6
2.1.2.6	closeDB()	6
2.1.2.7	commit()	6
2.1.2.8	connectDB(std::string info)	6
2.1.2.9	count_format_specifiers(std::string const &format)	7
2.1.2.10	countfield()	7
2.1.2.11	countrow()	7
2.1.2.12	DoubleToChar(double _X_, unsigned _precision_=0)	7
2.1.2.13	end()	7
2.1.2.14	fetch(int row, int column)	7
2.1.2.15	fetchAll(std::string, std::string cursor=""mycursor"")	8
2.1.2.16	fetchsql(std::string cursor=""mycursor"")	8
2.1.2.17	fetchupd(std::string sql)	8

2.1.2.18	<code>is_Table(const char *table)</code>	8
2.1.2.19	<code>nfield(std::string field)</code>	9
2.1.2.20	<code>openssl(std::string sql, std::string cursor=""mycursor"")</code>	9
2.1.2.21	<code>prepare(std::string const &format, Head &&head, Args &&...args)</code>	9
2.1.2.22	<code>qexec(std::string sql)</code>	9
2.1.2.23	<code>query(std::string sql, bool binary=true)</code>	9
2.1.2.24	<code>rollback()</code>	10
2.1.2.25	<code>savpoint()</code>	10
2.1.2.26	<code>savpointRelease()</code>	10
2.1.2.27	<code>savpointRollback()</code>	10
2.1.3	Friends And Related Function Documentation	10
2.1.3.1	<code>operator>></code>	10
Index		11

Chapter 1

Class Index

1.1 Class List

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

libpqwrp::libPQwrp	
Gestion parametre sql	3

Chapter 2

Class Documentation

2.1 libpqwrp::libPQwrp Class Reference

gestion parametre sql

```
#include <libpqwrp.h>
```

Public Member Functions

- 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 [autocommitON](#) ()
close to the database

- void `autocommitOFF` ()
set autocommit ON
- void `begin` ()
set autocommit OFF
- void `commit` ()
start transaction for commit
- void `rollback` ()
commit for transaction init
- void `end` ()
rollback for transaction init
- void `clearRes` ()
end transaction for init
- void `savpoint` ()
clear PGresult
- void `savpointRollback` ()
point de sauvegarde
- void `savpointRelease` ()
rollback 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 `fetchSql` (std::string cursor="mycursor")
query for fetchsql record / record
- void `fetchUpd` (std::string sql)
fetch record use openSQL
- void `prepare` (std::string const &format)
fetch select for update
- template<typename Head , typename... Args>
std::string `prepare` (std::string const &format, Head &&head, Args &&...args)
format without parameters causes an error
- std::stringstream `result` ()
retrieve values on the fields
- char * `DoubleToChar` (double _X_, unsigned _precision_=0)
out buffer;

Public Attributes

- bool `errorSQL`
count the number of parameters in the format
- bool `fetchEOF`
fatal_error
- int `rows`
fin de traitement get records
- 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

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.

2.1.1 Detailed Description

gestion parametre sql

prepare sql for-
matage avec template variadique etc..... #####

Definition at line 95 of file libpqwrp.h.

2.1.2 Member Function Documentation

2.1.2.1 void libpqwrp::libPQwrp::autocommitOFF ()

```
set autocommit ON
```

```
set autocommit OFF
```

Definition at line 258 of file libpqwrp.cpp.

2.1.2.2 void libpqwrp::libPQwrp::autocommitON ()

close to the database

```
set autocommit ON
```

Definition at line 252 of file libpqwrp.cpp.

2.1.2.3 void libpqwrp::libPQwrp::begin ()

set autocommit OFF

start transction for commit FOR UPDATE LOCK ENRG.

Definition at line 265 of file libpqwrp.cpp.

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

number column of the field

name of the field

Definition at line 198 of file libpqwrp.cpp.

2.1.2.5 void libpqwrp::libPQwrp::clearRes ()

end transaction for init

clear PGresult

Definition at line 293 of file libpqwrp.cpp.

2.1.2.6 void libpqwrp::libPQwrp::closeDB ()

if exist table of the database

close to the database

Definition at line 246 of file libpqwrp.cpp.

2.1.2.7 void libpqwrp::libPQwrp::commit ()

start transction for commit

commit for transaction init

Definition at line 272 of file libpqwrp.cpp.

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

connect to the database

init PGconn

init PGresult

Definition at line 74 of file libpqwrp.cpp.

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

contrôle PGresult

process

Definition at line 14 of file libpqwrp.cpp.

2.1.2.10 int libpqwrp::libPQwrp::countfield ()

name of the field

number of columns

Definition at line 210 of file libpqwrp.cpp.

2.1.2.11 int libpqwrp::libPQwrp::countrow ()

number of columns

number of rows

Definition at line 217 of file libpqwrp.cpp.

2.1.2.12 char * libpqwrp::libPQwrp::DoubleToChar (double *_X_*, unsigned *_precision_* = 0)

out buffer;

Double to char.

Definition at line 473 of file libpqwrp.cpp.

2.1.2.13 void libpqwrp::libPQwrp::end ()

rollback for transaction init

end transaction for init

Definition at line 286 of file libpqwrp.cpp.

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

PQQuery.

get the value of the row and the column

Definition at line 146 of file libpqwrp.cpp.

2.1.2.15 `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 326 of file libpqwrp.cpp.

2.1.2.16 `void libpqwrp::libPQwrp::fetchsql (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 416 of file libpqwrp.cpp.

2.1.2.17 `void libpqwrp::libPQwrp::fetchupd (std::string sql)`

fetch record use openSQL

query for fetchsql record / record read a line record only for update

end of row or without row

no end of row

Definition at line 442 of file libpqwrp.cpp.

2.1.2.18 `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 223 of file libpqwrp.cpp.

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

get the value of the row and the column

number column of the field

Definition at line 185 of file libpqwrp.cpp.

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

fetch ALL include requete

query for fetchsql record / record read a line record only

end of row or without row

close cursor

no end of row

Definition at line 374 of file libpqwrp.cpp.

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

format without parameters causes an error

formatting the query with parameters

Definition at line 201 of file libpqwrp.h.

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

connect to the database

PQexec.

Definition at line 90 of file libpqwrp.cpp.

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

PQexec.

PQquery. lock for update

Definition at line 105 of file libpqwrp.cpp.

2.1.2.24 void libpqwrp::libPQwrp::rollback ()

commit for transaction init

rollback for transaction init

Definition at line 279 of file libpqwrp.cpp.

2.1.2.25 void libpqwrp::libPQwrp::savpoint ()

clear PGresult

point de sauvegarde

Definition at line 299 of file libpqwrp.cpp.

2.1.2.26 void libpqwrp::libPQwrp::savpointRelease ()

rollback base savepoint

delete savepoint

Definition at line 313 of file libpqwrp.cpp.

2.1.2.27 void libpqwrp::libPQwrp::savpointRollback ()

point de sauvegarde

rollback base savepoint

Definition at line 306 of file libpqwrp.cpp.

2.1.3 Friends And Related Function Documentation

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

Double to char.

retrieve char* for stringstream dummy

Definition at line 241 of file libpqwrp.h.

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

- src/libpqwrp.h
- src/libpqwrp.cpp

Index

- autocommitOFF
 - libpqwrp::libPQwrp, 5
- autocommitON
 - libpqwrp::libPQwrp, 5
- begin
 - libpqwrp::libPQwrp, 5
- cfield
 - libpqwrp::libPQwrp, 6
- clearRes
 - libpqwrp::libPQwrp, 6
- closeDB
 - libpqwrp::libPQwrp, 6
- commit
 - libpqwrp::libPQwrp, 6
- connectDB
 - libpqwrp::libPQwrp, 6
- count_format_specifiers
 - libpqwrp::libPQwrp, 6
- countfield
 - libpqwrp::libPQwrp, 7
- countrow
 - libpqwrp::libPQwrp, 7
- DoubleToChar
 - libpqwrp::libPQwrp, 7
- end
 - libpqwrp::libPQwrp, 7
- fetch
 - libpqwrp::libPQwrp, 7
- fetchAll
 - libpqwrp::libPQwrp, 7
- fetchsql
 - libpqwrp::libPQwrp, 8
- fetchupd
 - libpqwrp::libPQwrp, 8
- is_Table
 - libpqwrp::libPQwrp, 8
- libpqwrp::libPQwrp, 3
 - autocommitOFF, 5
 - autocommitON, 5
 - begin, 5
 - cfield, 6
 - clearRes, 6
 - closeDB, 6
 - commit, 6
 - connectDB, 6
 - count_format_specifiers, 6
 - countfield, 7
 - countrow, 7
 - DoubleToChar, 7
 - end, 7
 - fetch, 7
 - fetchAll, 7
 - fetchsql, 8
 - fetchupd, 8
 - is_Table, 8
 - nfield, 8
 - opensql, 9
 - operator>>, 10
 - prepare, 9
 - qexec, 9
 - query, 9
 - rollback, 9
 - savpoint, 10
 - savpointRelease, 10
 - savpointRollback, 10
- nfield
 - libpqwrp::libPQwrp, 8
- opensql
 - libpqwrp::libPQwrp, 9
- operator>>
 - libpqwrp::libPQwrp, 10
- prepare
 - libpqwrp::libPQwrp, 9
- qexec
 - libpqwrp::libPQwrp, 9
- query
 - libpqwrp::libPQwrp, 9
- rollback
 - libpqwrp::libPQwrp, 9
- savpoint
 - libpqwrp::libPQwrp, 10
- savpointRelease
 - libpqwrp::libPQwrp, 10
- savpointRollback
 - libpqwrp::libPQwrp, 10