

Museum Project

Generated by Doxygen 1.8.18

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	5
2.1 Class List	5
3 Class Documentation	9
3.1 Hash::_ht Struct Reference	9
3.2 Account Class Reference	9
3.2.1 Constructor & Destructor Documentation	10
3.2.1.1 Account()	10
3.2.2 Member Function Documentation	11
3.2.2.1 add_row()	11
3.2.2.2 addNewUser()	11
3.2.2.3 getAllUsernames()	12
3.2.2.4 getNameFromId()	12
3.2.2.5 login()	12
3.2.2.6 newPIN()	13
3.2.2.7 remove_entries()	13
3.2.2.8 removeUser()	13
3.2.2.9 select_all()	14
3.2.2.10 select_cell()	14
3.2.2.11 select_elements_sql()	14
3.2.2.12 setAllUsernames()	15
3.2.2.13 update_value()	15
3.2.2.14 userData()	15
3.3 Admin Class Reference	16
3.3.1 Constructor & Destructor Documentation	16
3.3.1.1 Admin()	16
3.4 AgglInfo Struct Reference	17
3.5 AgglInfo::AgglInfo_col Struct Reference	17
3.6 AgglInfo::AgglInfo_func Struct Reference	18
3.7 analysisInfo Struct Reference	18
3.8 AuthContext Struct Reference	18
3.9 AutoincInfo Struct Reference	18
3.10 AuxData Struct Reference	19
3.11 Bitvec Struct Reference	19
3.12 BtCursor Struct Reference	19
3.13 BtLock Struct Reference	20
3.14 Btree Struct Reference	20
3.15 BtreePayload Struct Reference	21
3.16 BtShared Struct Reference	21
3.17 BusyHandler Struct Reference	22

3.18 CallCount Struct Reference	22
3.19 CellArray Struct Reference	22
3.20 CellInfo Struct Reference	23
3.21 CollSeq Struct Reference	23
3.22 Column Struct Reference	23
3.23 Comment Class Reference	24
3.23.1 Constructor & Destructor Documentation	24
3.23.1.1 Comment()	25
3.23.2 Member Function Documentation	25
3.23.2.1 add_row()	25
3.23.2.2 newComment()	25
3.23.2.3 remove_entries()	26
3.23.2.4 removeComment()	26
3.23.2.5 select_cells()	26
3.23.2.6 select_elements_sql()	27
3.23.2.7 setComment()	27
3.23.2.8 update_value()	27
3.24 compareInfo Struct Reference	28
3.25 CountCtx Struct Reference	28
3.26 With::Cte Struct Reference	29
3.27 Curator Class Reference	29
3.27.1 Constructor & Destructor Documentation	29
3.27.1.1 Curator() [1/2]	29
3.27.1.2 Curator() [2/2]	30
3.28 DateTime Struct Reference	30
3.29 Db Struct Reference	31
3.30 DbFixer Struct Reference	31
3.31 DbQuoteStr Struct Reference	31
3.32 DBTable Class Reference	32
3.32.1 Detailed Description	33
3.33 DBTool Class Reference	33
3.33.1 Detailed Description	33
3.33.2 Member Function Documentation	33
3.33.2.1 open_db()	34
3.34 DistinctCtx Struct Reference	34
3.35 et_info Struct Reference	34
3.36 Expr Struct Reference	35
3.37 ExprList Struct Reference	35
3.38 ExprList::ExprList_item Struct Reference	36
3.39 FileChunk Struct Reference	36
3.40 FilePoint Struct Reference	37
3.41 FKey Struct Reference	37

3.42 FrameBound Struct Reference	37
3.43 fts5_api Struct Reference	38
3.44 fts5_tokenizer Struct Reference	38
3.45 Fts5ExtensionApi Struct Reference	38
3.46 Fts5Phraseliter Struct Reference	39
3.47 FuncDef Struct Reference	39
3.48 FuncDefHash Struct Reference	40
3.49 FuncDestructor Struct Reference	40
3.50 Hash Struct Reference	40
3.51 HashElem Struct Reference	41
3.52 HiddenIndexInfo Struct Reference	41
3.53 IdList Struct Reference	41
3.54 IdList::IdList_item Struct Reference	42
3.55 IdxCover Struct Reference	42
3.56 IdxExprTrans Struct Reference	42
3.57 Incrblob Struct Reference	43
3.58 IncrMerger Struct Reference	43
3.59 Index Struct Reference	43
3.60 IndexSample Struct Reference	44
3.61 InitData Struct Reference	44
3.62 IntegrityCk Struct Reference	45
3.63 KeyInfo Struct Reference	45
3.64 LastValueCtx Struct Reference	45
3.65 Lookaside Struct Reference	46
3.66 LookasideSlot Struct Reference	46
3.67 MemJournal Struct Reference	46
3.68 MemPage Struct Reference	47
3.69 sqlite3_value::MemValue Union Reference	47
3.70 MergeEngine Struct Reference	48
3.71 Module Struct Reference	48
3.72 MuseumPage Class Reference	48
3.72.1 Constructor & Destructor Documentation	50
3.72.1.1 MuseumPage() [1/3]	50
3.72.1.2 MuseumPage() [2/3]	50
3.72.1.3 MuseumPage() [3/3]	50
3.72.2 Member Function Documentation	51
3.72.2.1 add_row()	51
3.72.2.2 addMuseum()	51
3.72.2.3 getMuseumByCuratorId()	52
3.72.2.4 getMuseumChildrenId()	52
3.72.2.5 getMuseumChildrenNames()	52
3.72.2.6 getMuseumId()	53

3.72.2.7 getMuseumPageFromId()	53
3.72.2.8 getMuseumSearchId()	54
3.72.2.9 remove_entries()	54
3.72.2.10 select_cell_search()	54
3.73 NameContext Struct Reference	55
3.74 NthValueCtx Struct Reference	55
3.75 NtileCtx Struct Reference	56
3.76 VdbeOp::p4union Union Reference	56
3.77 Pager Struct Reference	56
3.78 PagerSavepoint Struct Reference	58
3.79 Parse Struct Reference	58
3.80 PCache Struct Reference	59
3.81 PCache1 Struct Reference	60
3.82 PgFreeslot Struct Reference	60
3.83 PgHdr Struct Reference	61
3.84 PgHdr1 Struct Reference	61
3.85 PGroup Struct Reference	61
3.86 PmaReader Struct Reference	62
3.87 PmaWriter Struct Reference	62
3.88 PragmaName Struct Reference	62
3.89 PragmaVtab Struct Reference	63
3.90 PragmaVtabCursor Struct Reference	63
3.91 PreUpdate Struct Reference	63
3.92 PrintfArguments Struct Reference	64
3.93 RenameCtx Struct Reference	64
3.94 RenameToken Struct Reference	64
3.95 ReusableSpace Struct Reference	64
3.96 RowLoadInfo Struct Reference	65
3.97 RowSet Struct Reference	65
3.98 RowSetChunk Struct Reference	65
3.99 RowSetEntry Struct Reference	66
3.100 Savepoint Struct Reference	66
3.101 ScanStatus Struct Reference	66
3.102 Schema Struct Reference	67
3.103 FKey::sColMap Struct Reference	67
3.104 Select Struct Reference	67
3.105 SelectDest Struct Reference	68
3.106 SortCtx Struct Reference	68
3.107 SorterFile Struct Reference	68
3.108 SorterList Struct Reference	69
3.109 SorterRecord Struct Reference	69
3.110 SortSubtask Struct Reference	69

3.111 sqlite3 Struct Reference	70
3.112 sqlite3_api_routines Struct Reference	72
3.113 sqlite3_backup Struct Reference	77
3.114 sqlite3_context Struct Reference	77
3.115 sqlite3_file Struct Reference	77
3.116 sqlite3_index_info::sqlite3_index_constraint Struct Reference	78
3.117 sqlite3_index_info::sqlite3_index_constraint_usage Struct Reference	78
3.118 sqlite3_index_info Struct Reference	78
3.119 sqlite3_index_info::sqlite3_index_orderby Struct Reference	79
3.120 sqlite3_io_methods Struct Reference	79
3.121 sqlite3_mem_methods Struct Reference	80
3.122 sqlite3_module Struct Reference	80
3.123 sqlite3_mutex Struct Reference	81
3.124 sqlite3_mutex_methods Struct Reference	81
3.125 sqlite3_pcache_methods Struct Reference	81
3.126 sqlite3_pcache_methods2 Struct Reference	82
3.127 sqlite3_pcache_page Struct Reference	82
3.128 sqlite3_rtree_geometry Struct Reference	82
3.129 sqlite3_rtree_query_info Struct Reference	83
3.130 sqlite3_snapshot Struct Reference	83
3.131 sqlite3_str Struct Reference	83
3.132 sqlite3_value Struct Reference	84
3.133 sqlite3_vfs Struct Reference	84
3.134 sqlite3_vtab Struct Reference	85
3.135 sqlite3_vtab_cursor Struct Reference	85
3.136 Sqlite3Config Struct Reference	85
3.137 sqlite3::sqlite3InitInfo Struct Reference	86
3.138 SQLiteThread Struct Reference	86
3.139 SrcCount Struct Reference	87
3.140 SrcList Struct Reference	87
3.141 SrcList::SrcList_item Struct Reference	87
3.142 Stat4Accum Struct Reference	88
3.143 Stat4Sample Struct Reference	89
3.144 SubProgram Struct Reference	89
3.145 SubstContext Struct Reference	89
3.146 SumCtx Struct Reference	90
3.147 Table Struct Reference	90
3.148 TableLock Struct Reference	91
3.149 TabResult Struct Reference	91
3.150 Token Struct Reference	91
3.151 TrigEvent Struct Reference	92
3.152 Trigger Struct Reference	92

3.153 TriggerPrg Struct Reference	92
3.154 TriggerStep Struct Reference	93
3.155 unixFile Struct Reference	93
3.156 unixFileId Struct Reference	94
3.157 unixInodeInfo Struct Reference	94
3.158 unixShm Struct Reference	94
3.159 unixShmNode Struct Reference	95
3.160 UnixUnusedFd Struct Reference	95
3.161 UnpackedRecord Struct Reference	95
3.162 Upsert Struct Reference	96
3.163 ValueNewStat4Ctx Struct Reference	96
3.164 Vdbe Struct Reference	96
3.165 VdbeCursor Struct Reference	98
3.166 VdbeFrame Struct Reference	98
3.167 VdbeOp Struct Reference	99
3.168 VdbeOpList Struct Reference	99
3.169 VdbeSorter Struct Reference	100
3.170 Viewer Class Reference	100
3.170.1 Constructor & Destructor Documentation	101
3.170.1.1 Viewer()	101
3.171 VtabCtx Struct Reference	102
3.172 VTable Struct Reference	102
3.173 vxworksFileId Struct Reference	102
3.174 Wal Struct Reference	103
3.175 WalCkptInfo Struct Reference	103
3.176 WalHashLoc Struct Reference	104
3.177 WalIndexHdr Struct Reference	104
3.178 WalIterator Struct Reference	104
3.179 Walker Struct Reference	105
3.180 WalIterator::WalSegment Struct Reference	105
3.181 WalWriter Struct Reference	106
3.182 WhereAndInfo Struct Reference	106
3.183 WhereClause Struct Reference	106
3.184 WhereConst Struct Reference	107
3.185 WhereExprMod Struct Reference	107
3.186 WhereInfo Struct Reference	107
3.187 WhereLevel Struct Reference	108
3.188 WhereLoop Struct Reference	109
3.189 WhereLoopBuilder Struct Reference	109
3.190 WhereMaskSet Struct Reference	110
3.191 WhereOrCost Struct Reference	110
3.192 WhereOrInfo Struct Reference	110

3.193 WhereOrSet Struct Reference	110
3.194 WherePath Struct Reference	111
3.195 WhereScan Struct Reference	111
3.196 WhereTerm Struct Reference	111
3.197 Window Struct Reference	112
3.198 WindowCodeArg Struct Reference	113
3.199 WindowCsrAndReg Struct Reference	113
3.200 WindowRewrite Struct Reference	113
3.201 With Struct Reference	114
3.202 YYMINORTYPE Union Reference	114
3.203 yyParser Struct Reference	115
3.204 yyStackEntry Struct Reference	115
Index	117

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Hash::_ht	9
AggInfo	17
AggInfo::AggInfo_col	17
AggInfo::AggInfo_func	18
analysisInfo	18
AuthContext	18
AutoinclInfo	18
AuxData	19
Bitvec	19
BtCursor	19
BtLock	20
Btree	20
BtreePayload	21
BtShared	21
BusyHandler	22
CallCount	22
CellArray	22
CellInfo	23
CollSeq	23
Column	23
compareInfo	28
CountCtx	28
With::Cte	29
DateTime	30
Db	31
DbFixer	31
DblquoteStr	31
DBTable	32
Account	9
Comment	24
MuseumPage	48
DBTool	33
DistinctCtx	34
et_info	34
Expr	35

ExprList	35
ExprList::ExprList_item	36
FileChunk	36
FilePoint	37
FKey	37
FrameBound	37
fts5_api	38
fts5_tokenizer	38
Fts5ExtensionApi	38
Fts5Phraseltr	39
FuncDef	39
FuncDefHash	40
FuncDestructor	40
Hash	40
HashElem	41
HiddenIndexInfo	41
IdList	41
IdList::IdList_item	42
IdxCover	42
IdxExprTrans	42
Incrblob	43
IncrMerger	43
Index	43
IndexSample	44
InitData	44
IntegrityCk	45
KeyInfo	45
LastValueCtx	45
Lookaside	46
LookasideSlot	46
MemJournal	46
MemPage	47
sqlite3_value::MemValue	47
MergeEngine	48
Module	48
NameContext	55
NthValueCtx	55
NtileCtx	56
VdbeOp::p4union	56
Pager	56
PagerSavepoint	58
Parse	58
PCache	59
PCache1	60
PgFreeslot	60
PgHdr	61
PgHdr1	61
PGroup	61
PmaReader	62
PmaWriter	62
PragmaName	62
PragmaVtab	63
PragmaVtabCursor	63
PreUpdate	63
PrintfArguments	64
RenameCtx	64
RenameToken	64
ReusableSpace	64

RowLoadInfo	65
RowSet	65
RowSetChunk	65
RowSetEntry	66
Savepoint	66
ScanStatus	66
Schema	67
FKey::sColMap	67
Select	67
SelectDest	68
SortCtx	68
SorterFile	68
SorterList	69
SorterRecord	69
SortSubtask	69
sqlite3	70
sqlite3_api_routines	72
sqlite3_backup	77
sqlite3_context	77
sqlite3_file	77
sqlite3_index_info::sqlite3_index_constraint	78
sqlite3_index_info::sqlite3_index_constraint_usage	78
sqlite3_index_info	78
sqlite3_index_info::sqlite3_index_orderby	79
sqlite3_io_methods	79
sqlite3_mem_methods	80
sqlite3_module	80
sqlite3_mutex	81
sqlite3_mutex_methods	81
sqlite3_pcache_methods	81
sqlite3_pcache_methods2	82
sqlite3_pcache_page	82
sqlite3_rtree_geometry	82
sqlite3_rtree_query_info	83
sqlite3_snapshot	83
sqlite3_str	83
sqlite3_value	84
sqlite3_vfs	84
sqlite3_vtab	85
sqlite3_vtab_cursor	85
Sqlite3Config	85
sqlite3::sqlite3InitInfo	86
SQLiteThread	86
SrcCount	87
SrcList	87
SrcList::SrcList_item	87
Stat4Accum	88
Stat4Sample	89
SubProgram	89
SubstContext	89
SumCtx	90
Table	90
TableLock	91
TabResult	91
Token	91
TrigEvent	92
Trigger	92
TriggerPrg	92

TriggerStep	93
unixFile	93
unixFileId	94
unixInodeInfo	94
unixShm	94
unixShmNode	95
UnixUnusedFd	95
UnpackedRecord	95
Upsert	96
ValueNewStat4Ctx	96
Vdbe	96
VdbeCursor	98
VdbeFrame	98
VdbeOp	99
VdbeOpList	99
VdbeSorter	100
Viewer	100
Curator	29
Admin	16
VtabCtx	102
VTable	102
vxworksFileId	102
Wal	103
WalCkptInfo	103
WalHashLoc	104
WalIndexHdr	104
WalIterator	104
Walker	105
WalIterator::WalSegment	105
WalWriter	106
WhereAndInfo	106
WhereClause	106
WhereConst	107
WhereExprMod	107
WhereInfo	107
WhereLevel	108
WhereLoop	109
WhereLoopBuilder	109
WhereMaskSet	110
WhereOrCost	110
WhereOrInfo	110
WhereOrSet	110
WherePath	111
WhereScan	111
WhereTerm	111
Window	112
WindowCodeArg	113
WindowCsrAndReg	113
WindowRewrite	113
With	114
YYMINORTYPE	114
yyParser	115
yyStackEntry	115

Chapter 2

Class Index

2.1 Class List

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

Hash::_ht	9
Account	9
Admin	16
AggInfo	17
AggInfo::AggInfo_col	17
AggInfo::AggInfo_func	18
analysisInfo	18
AuthContext	18
AutoincInfo	18
AuxData	19
Bitvec	19
BtCursor	19
BtLock	20
Btree	20
BtreePayload	21
BtShared	21
BusyHandler	22
CallCount	22
CellArray	22
CellInfo	23
CollSeq	23
Column	23
Comment	24
compareInfo	28
CountCtx	28
With::Cte	29
Curator	29
DateTime	30
Db	31
DbFixer	31
DblquoteStr	31
DBTable	
The DBTable (p. 32) class	32
DBTool	
The Database management tool	33

DistinctCtx	34
et_info	34
Expr	35
ExprList	35
ExprList::ExprList_item	36
FileChunk	36
FilePoint	37
FKey	37
FrameBound	37
fts5_api	38
fts5_tokenizer	38
Fts5ExtensionApi	38
Fts5Phraselter	39
FuncDef	39
FuncDefHash	40
FuncDestructor	40
Hash	40
HashElem	41
HiddenIndexInfo	41
IdList	41
IdList::IdList_item	42
IdxCover	42
IdxExprTrans	42
Incrblob	43
IncrMerger	43
Index	43
IndexSample	44
InitData	44
IntegrityCk	45
KeyInfo	45
LastValueCtx	45
Lookaside	46
LookasideSlot	46
MemJournal	46
MemPage	47
sqlite3_value::MemValue	47
MergeEngine	48
Module	48
MuseumPage	48
NameContext	55
NthValueCtx	55
NtileCtx	56
VdbeOp::p4union	56
Pager	56
PagerSavepoint	58
Parse	58
PCache	59
PCache1	60
PgFreeslot	60
PgHdr	61
PgHdr1	61
PGroup	61
PmaReader	62
PmaWriter	62
PragmaName	62
PragmaVtab	63
PragmaVtabCursor	63
PreUpdate	63

PrintfArguments	64
RenameCtx	64
RenameToken	64
ReusableSpace	64
RowLoadInfo	65
RowSet	65
RowSetChunk	65
RowSetEntry	66
Savepoint	66
ScanStatus	66
Schema	67
FKey::sColMap	67
Select	67
SelectDest	68
SortCtx	68
SorterFile	68
SorterList	69
SorterRecord	69
SortSubtask	69
sqlite3	70
sqlite3_api_routines	72
sqlite3_backup	77
sqlite3_context	77
sqlite3_file	77
sqlite3_index_info::sqlite3_index_constraint	78
sqlite3_index_info::sqlite3_index_constraint_usage	78
sqlite3_index_info	78
sqlite3_index_info::sqlite3_index_orderby	79
sqlite3_io_methods	79
sqlite3_mem_methods	80
sqlite3_module	80
sqlite3_mutex	81
sqlite3_mutex_methods	81
sqlite3_pcache_methods	81
sqlite3_pcache_methods2	82
sqlite3_pcache_page	82
sqlite3_rtree_geometry	82
sqlite3_rtree_query_info	83
sqlite3_snapshot	83
sqlite3_str	83
sqlite3_value	84
sqlite3_vfs	84
sqlite3_vtab	85
sqlite3_vtab_cursor	85
Sqlite3Config	85
sqlite3::sqlite3InitInfo	86
SQLiteThread	86
SrcCount	87
SrcList	87
SrcList::SrcList_item	87
Stat4Accum	88
Stat4Sample	89
SubProgram	89
SubstContext	89
SumCtx	90
Table	90
TableLock	91
TabResult	91

Token	91
TrigEvent	92
Trigger	92
TriggerPrg	92
TriggerStep	93
unixFile	93
unixFileId	94
unixInodeInfo	94
unixShm	94
unixShmNode	95
UnixUnusedFd	95
UnpackedRecord	95
Upsert	96
ValueNewStat4Ctx	96
Vdbe	96
VdbeCursor	98
VdbeFrame	98
VdbeOp	99
VdbeOpList	99
VdbeSorter	100
Viewer	100
VtabCtx	102
VTable	102
vxworksFileId	102
Wal	103
WalCkptInfo	103
WalHashLoc	104
WalIndexHdr	104
WalIterator	104
Walker	105
WalIterator::WalSegment	105
WalWriter	106
WhereAndInfo	106
WhereClause	106
WhereConst	107
WhereExprMod	107
WhereInfo	107
WhereLevel	108
WhereLoop	109
WhereLoopBuilder	109
WhereMaskSet	110
WhereOrCost	110
WhereOrInfo	110
WhereOrSet	110
WherePath	111
WhereScan	111
WhereTerm	111
Window	112
WindowCodeArg	113
WindowCsrAndReg	113
WindowRewrite	113
With	114
YYMINORTYPE	114
yyParser	115
yyStackEntry	115

Chapter 3

Class Documentation

3.1 Hash::_ht Struct Reference

Public Attributes

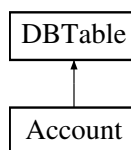
- unsigned int **count**
- **HashElem** * **chain**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.2 Account Class Reference

Inheritance diagram for Account:



Public Member Functions

- **Account** (**DBTool** *db, string name, bool _ifTesting)
Account::Account Constructor that initializes table.
- **Viewer** * **login** (string _un, string _pw)
Account::login (p. 12) Logs the user in if the username and password match.
- **Viewer** * **addNewUser** (string UN, string PW, string PIN, string DESC, string PIC)
Account::addNewUser (p. 11) Adds new user to the database table if all formatting is correct and PIN matches.
- string **removeUser** (string)
Account::removeUser (p. 13) Removes user by changing the boolean deleted field.
- bool **newPIN** (string)

- **Account::newPIN** (p. 13) *Changes the admin's authentication PIN in the table.*
- `vector< string > userData` (string)
- **Account::userData** (p. 15) *Returns all data about a given user.*
- `string getNameFromId` (int)
- **Account::getNameFromId** (p. 12) *Returns the name of the user given the user's id.*
- `void store_create_sql` ()
- **Account::store_create_sql** (p. 10) *String that creates SQL query to create a new table.*
- `virtual void store_add_row_sql` ()
- `bool add_row` (int id, string username, string password, string privilege, string description, string picture, string uniquePW, string deleted)
- **Account::add_row** (p. 11) *Adds a new row to the table.*
- `bool select_all` ()
- **Account::select_all** (p. 14) *Prints out all of the entries in the table.*
- `void select_elements_sql` (string)
- **Account::select_elements_sql** (p. 14) *Returns the values of an entire column.*
- `void select_cell` (string, string, string)
- **Account::select_cell** (p. 14) *Finds the requested value in a particular cell.*
- `void count_rows` ()
- **Account::count_rows** (p. 10) *Counts how many rows are in the table.*
- `void remove_entries` ()
- *Account::remove_row Removes a row given a column name and key value.*
- `void update_value` (string, string, string, string)
- **Account::update_value** (p. 15) *Updates the value of a given cell.*
- `vector< string > getAllUsernames` ()
- **Account::getAllUsernames** (p. 12) *Getter method for allUsernames.*
- `void setAllUsernames` (vector< string >)
- **Account::setAllUsernames** (p. 14) *Setter method for allUsernames.*

Protected Attributes

- `string sql_select_all`
- `string sql_select`

3.2.1 Constructor & Destructor Documentation

3.2.1.1 Account()

```
Account::Account (
    DBTool * _db,
    string name,
    bool _ifTesting )
```

Account::Account Constructor that initializes table.

Parameters

<code>_db</code>	Reference to DBTool (p. 33)
<code>name</code>	Name of the database table
<code>_ifTesting</code>	This is true if we are unit testing

3.2.2 Member Function Documentation

3.2.2.1 add_row()

```
bool Account::add_row (
    int id,
    string username,
    string password,
    string privelege,
    string description,
    string picture,
    string uniquePW,
    string deleted )
```

Account::add_row (p. 11) Adds a new row to the table.

Parameters

<i>id</i>	Unique ID of the user
<i>username</i>	The user's username
<i>password</i>	The user's password
<i>privelege</i>	The privelege level of the user
<i>description</i>	The user's profile description
<i>picture</i>	The user's profile picture file name
<i>uniquePW</i>	The user's PIN, only the admin's cell will be filled
<i>deleted</i>	Whether the admin deleted said account or not

Returns

Whether the addition was successful or not

3.2.2.2 addNewUser()

```
Viewer * Account::addNewUser (
    string UN,
    string PW,
    string PIN,
    string DESC,
    string PIC )
```

Account::addNewUser (p. 11) Adds new user to the database table if all formatting is correct and PIN matches.

Parameters

<i>UN</i>	New username
<i>PW</i>	New password
<i>PIN</i>	The PIN that is supposed to match the Admin (p. 16)'s PIN
<i>DESC</i>	New description
<i>PIC</i>	New picture file name

Returns

Reference to the new user

3.2.2.3 getAllUsernames()

```
vector< string > Account::getAllUsernames ( )
```

Account::getAllUsernames (p. 12) Getter method for allUsernames.

Returns

Vector of all usernames in the system

3.2.2.4 getNameFromId()

```
string Account::getNameFromId (
    int ID )
```

Account::getNameFromId (p. 12) Returns the name of the user given the user's id.

Parameters

<i>ID</i>	The user's ID
-----------	---------------

Returns

The username

3.2.2.5 login()

```
Viewer * Account::login (
    string _un,
    string _pw )
```

Account::login (p. 12) Logs the user in if the username and password match.

Parameters

<i>_un</i>	Inputted username
<i>_pw</i>	Inputted password

Returns

A reference to the viewer class that contains all user information

3.2.2.6 newPIN()

```
bool Account::newPIN (
    string p )
```

Account::newPIN (p. 13) Changes the admin's authentication PIN in the table.

Parameters

<i>p</i>	The new PIN
----------	-------------

Returns

Whether the PIN change was successful or not

3.2.2.7 remove_entries()

```
void Account::remove_entries ( )
```

Account::remove_row Removes a row given a column name and key value.

Parameters

<i>n</i>	The key value of the row
<i>cn</i>	The column name to match the key

3.2.2.8 removeUser()

```
string Account::removeUser (
    string name )
```

Account::removeUser (p. 13) Removes user by changing the boolean deleted field.

Parameters

<i>name</i>	The username that should be deleted
-------------	-------------------------------------

Returns

The picture file name of the user so that the file can be deleted in another method

3.2.2.9 select_all()

```
bool Account::select_all ( )
```

Account::select_all (p. 14) Prints out all of the entries in the table.

Returns

Whether the query was successful or not

3.2.2.10 select_cell()

```
void Account::select_cell (
    string whichRow,
    string row,
    string col )
```

Account::select_cell (p. 14) Finds the requested value in a particular cell.

Parameters

<i>whichRow</i>	Key to match so that the row is identified
<i>row</i>	Column (p. 23) name to match whichRow with to identify the row of the desired cell
<i>col</i>	Column (p. 23) name of the desired cell

3.2.2.11 select_elements_sql()

```
void Account::select_elements_sql (
    string col )
```

Account::select_elements_sql (p. 14) Returns the values of an entire column.

Parameters

<i>col</i>	Column (p. 23) name
------------	----------------------------

3.2.2.12 setAllUsernames()

```
void Account::setAllUsernames (
    vector< string > x )
```

Account::setAllUsernames (p. 14) Setter method for allUsernames.

Parameters

x	New vector that allUsernames will be set to
---	---

3.2.2.13 update_value()

```
void Account::update_value (
    string colName,
    string newValue,
    string rowName,
    string rowValue )
```

Account::update_value (p. 15) Updates the value of a given cell.

Parameters

<i>colName</i>	The column name of the cell that needs to be changed
<i>newValue</i>	New value of the cell
<i>rowName</i>	The column name where the rowValue will be matched with
<i>rowValue</i>	Decides which row to update

3.2.2.14 userData()

```
vector< string > Account::userData (
    string x )
```

Account::userData (p. 15) Returns all data about a given user.

Parameters

x	The username
---	--------------

Returns

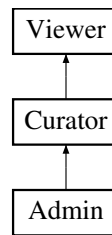
Vector of all of the data

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/account.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/account.cpp

3.3 Admin Class Reference

Inheritance diagram for Admin:



Public Member Functions

- **Admin** ()
Admin::Admin (p. 16) *Default Constructor.*
- **Admin** (int _userID, string _un, string _pw, string _priv, string _desc, string _selfPic, string _uniquePW)
Admin::Admin (p. 16) *Constructor that initializes all of the class instance variables.*

Additional Inherited Members

3.3.1 Constructor & Destructor Documentation

3.3.1.1 Admin()

```

Admin::Admin (
    int _userID,
    string _un,
    string _pw,
    string _priv,
    string _desc,
    string _selfPic,
    string _uniquePW )
  
```

Admin::Admin (p. 16) Constructor that initializes all of the class instance variables.

Parameters

<code>_userID</code>	Admin (p. 16)'s ID
<code>_un</code>	Admin (p. 16)'s username
<code>_pw</code>	Admin (p. 16)'s password
<code>_priv</code>	Admin (p. 16)'s privilege level
<code>_desc</code>	Admin (p. 16)'s self description
<code>_selfPic</code>	Admin (p. 16)'s profile picture
<code>_uniquePW</code>	Admin (p. 16)'s determined unique authentication PIN password

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/admin.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/admin.cpp

3.4 AggInfo Struct Reference

Classes

- struct **AggInfo_col**
- struct **AggInfo_func**

Public Attributes

- u8 **directMode**
- u8 **useSortingIdx**
- int **sortingIdx**
- int **sortingIdxPTab**
- int **nSortingColumn**
- int **mnReg**
- int **mxReg**
- **ExprList** * **pGroupBy**
- struct **AggInfo::AggInfo_col** * **aCol**
- int **nColumn**
- int **nAccumulator**
- struct **AggInfo::AggInfo_func** * **aFunc**
- int **nFunc**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.5 AggInfo::AggInfo_col Struct Reference

Public Attributes

- **Table** * **pTab**
- int **iTable**
- int **iColumn**
- int **iSorterColumn**
- int **iMem**
- **Expr** * **pExpr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.6 AggInfo::AggInfo_func Struct Reference

Public Attributes

- **Expr** * **pExpr**
- **FuncDef** * **pFunc**
- int **iMem**
- int **iDistinct**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.7 analysisInfo Struct Reference

Public Attributes

- **sqlite3** * **db**
- const char * **zDatabase**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.8 AuthContext Struct Reference

Public Attributes

- const char * **zAuthContext**
- **Parse** * **pParse**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.9 AutoincInfo Struct Reference

Public Attributes

- **AutoincInfo** * **pNext**
- **Table** * **pTab**
- int **iDb**
- int **regCtr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.10 AuxData Struct Reference

Public Attributes

- int **iAuxOp**
- int **iAuxArg**
- void * **pAux**
- void(* **xDeleteAux**)(void *)
- **AuxData** * **pNextAux**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.11 Bitvec Struct Reference

Public Attributes

- u32 **iSize**
- u32 **nSet**
- u32 **iDivisor**
- union {
 BITVEC_TELEM **aBitmap** [BITVEC_NELEM]
 u32 **aHash** [BITVEC_NINT]
 Bitvec * **apSub** [BITVEC_NPTR]
} **u**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.12 BtCursor Struct Reference

Public Attributes

- u8 **eState**
- u8 **curFlags**
- u8 **curPagerFlags**
- u8 **hints**
- int **skipNext**
- **Btree** * **pBtree**
- Pgno * **aOverflow**
- void * **pKey**
- **BtShared** * **pBt**
- **BtCursor** * **pNext**
- **CellInfo** **info**

- i64 **nKey**
- Pgno **pgnoRoot**
- i8 **iPage**
- u8 **curlIntKey**
- u16 **ix**
- u16 **aildx** [BTCursor_MAX_DEPTH-1]
- struct **KeyInfo** * **pKeyInfo**
- **MemPage** * **pPage**
- **MemPage** * **apPage** [BTCursor_MAX_DEPTH-1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.13 BtLock Struct Reference

Public Attributes

- **Btree** * **pBtree**
- Pgno **iTable**
- u8 **eLock**
- **BtLock** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.14 Btree Struct Reference

Public Attributes

- **sqlite3** * **db**
- **BtShared** * **pBt**
- u8 **inTrans**
- u8 **sharable**
- u8 **locked**
- u8 **hasIncrblobCur**
- int **wantToLock**
- int **nBackup**
- u32 **iDataVersion**
- **Btree** * **pNext**
- **Btree** * **pPrev**
- **BtLock** **lock**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.15 BtreePayload Struct Reference

Public Attributes

- const void * **pKey**
- sqlite3_int64 **nKey**
- const void * **pData**
- sqlite3_value * **aMem**
- u16 **nMem**
- int **nData**
- int **nZero**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.16 BtShared Struct Reference

Public Attributes

- Pager * **pPager**
- sqlite3 * **db**
- BtCursor * **pCursor**
- MemPage * **pPage1**
- u8 **openFlags**
- u8 **autoVacuum**
- u8 **incrVacuum**
- u8 **bDoTruncate**
- u8 **inTransaction**
- u8 **max1bytePayload**
- u16 **btsFlags**
- u16 **maxLocal**
- u16 **minLocal**
- u16 **maxLeaf**
- u16 **minLeaf**
- u32 **pageSize**
- u32 **usableSize**
- int **nTransaction**
- u32 **nPage**
- void * **pSchema**
- void(* **xFreeSchema**)(void *)
- sqlite3_mutex * **mutex**
- Bitvec * **pHasContent**
- int **nRef**
- BtShared * **pNext**
- BtLock * **pLock**
- Btree * **pWriter**
- u8 * **pTmpSpace**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.17 BusyHandler Struct Reference

Public Attributes

- `int(* xBusyHandler)(void *, int)`
- `void * pBusyArg`
- `int nBusy`
- `u8 bExtraFileArg`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.18 CallCount Struct Reference

Public Attributes

- `i64 nValue`
- `i64 nStep`
- `i64 nTotal`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.19 CellArray Struct Reference

Public Attributes

- `int nCell`
- `MemPage * pRef`
- `u8 ** apCell`
- `u16 * szCell`
- `u8 * apEnd [NB *2]`
- `int ixNx [NB *2]`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.20 CellInfo Struct Reference

Public Attributes

- i64 **nKey**
- u8 * **pPayload**
- u32 **nPayload**
- u16 **nLocal**
- u16 **nSize**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.21 CollSeq Struct Reference

Public Attributes

- char * **zName**
- u8 **enc**
- void * **pUser**
- int(* **xCmp**)(void *, int, const void *, int, const void *)
- void(* **xDel**)(void *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.22 Column Struct Reference

Public Attributes

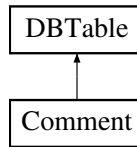
- char * **zName**
- Expr * **pDflt**
- char * **zColl**
- u8 **notNull**
- char **affinity**
- u8 **szEst**
- u16 **colFlags**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.23 Comment Class Reference

Inheritance diagram for Comment:



Public Member Functions

- **Comment** (**DBTool** *db, string name, bool _isTesting)
Comment (p. 24) constructor.
- void **store_create_sql** ()
store_create_sql sql statement for creates on in table
- virtual void **store_add_row_sql** ()
store_add_row_sql sql statement for row add on ining to table
- bool **add_row** (int museumId, string author, string content, string deleted)
add_row adds entry to Comments database
- vector< string > **select_cells** (string whichRow, string row, string col)
select_cells selects a value for a single field for a single element in the table
- vector< string > **select_elements_sql** (string col)
select_elements_sql gets the entire column of a certain field in the database
- void **newComment** (int mID, string a, string c)
newComment basically just adds all functionality for a new comment; calls *add_row*
- void **update_value** (string colName, string newValue, string rowName, string rowValue)
Comment::update_value (p. 27) Updates the value of a given cell.
- bool **removeComment** (string content)
removeComment Changes the boolean status to "Deleted" according to a given comment string
- int **count_rows** ()
count_rows return number of rows
- vector< string > **getMuseumComments** (int)
- void **setComment** (string s)
setComment sets the comment for the current string
- void **remove_entries** ()
Comment::remove_row Removes a row given a column name and key value.
- void **setDB** (**sqlite3** *)

Protected Attributes

- string **sql_select_all**
- string **sql_select**

3.23.1 Constructor & Destructor Documentation

3.23.1.1 Comment()

```
Comment::Comment (
    DBTool * db,
    string name,
    bool _isTesting )
```

Comment (p. 24) constructor.

Parameters

<i>db</i>	database tool for all comments
<i>name</i>	name for database

3.23.2 Member Function Documentation

3.23.2.1 add_row()

```
bool Comment::add_row (
    int museumId,
    string author,
    string content,
    string deleted )
```

add_row adds entry to Comments database

Parameters

<i>museumId</i>	the museum id field
<i>author</i>	the author field
<i>content</i>	the content of the comment

Returns

bool

3.23.2.2 newComment()

```
void Comment::newComment (
    int mID,
    string a,
    string c )
```

newComment basically just adds all functionality for a new comment; calls **add_row**

Parameters

<i>mID</i>	the id of the museum that is being commented on
<i>a</i>	the author of the comment
<i>c</i>	the content in the comment

3.23.2.3 remove_entries()

```
void Comment::remove_entries ( )
```

Comment::remove_row Removes a row given a column name and key value.

Parameters

<i>n</i>	The key value of the row
<i>cn</i>	The column name to match the key

3.23.2.4 removeComment()

```
bool Comment::removeComment (
    string content )
```

removeComment Changes the boolean status to "Deleted" according to a given comment string

Parameters

<i>content</i>	The comment to be deleted
----------------	---------------------------

Returns

whether the deletion was successful or not

3.23.2.5 select_cells()

```
vector< string > Comment::select_cells (
    string whichRow,
    string row,
    string col )
```

select_cells selects a value for a single field for a single element in the table

Parameters

<i>whichRow</i>	the index of the row we want in the database
<i>row</i>	the name of the primary key
<i>col</i>	the field that we are looking for

Returns

vector of selected cells

3.23.2.6 select_elements_sql()

```
vector< string > Comment::select_elements_sql (
    string col )
```

`select_elements_sql` gets the entire column of a certain field in the database

Parameters

<i>col</i>	the field we are searching for
------------	--------------------------------

Returns

vector of selected elements

3.23.2.7 setComment()

```
void Comment::setComment (
    string s )
```

`setComment` sets the comment for the current string

Parameters

<i>s</i>	the comment we are writing
----------	----------------------------

3.23.2.8 update_value()

```
void Comment::update_value (
    string colName,
    string newValue,
```

```
string rowName,  
string rowValue )
```

Comment::update_value (p. 27) Updates the value of a given cell.

Parameters

<i>colName</i>	The column name of the cell that needs to be changed
<i>newValue</i>	New value of the cell
<i>rowName</i>	The column name where the rowValue will be matched with
<i>rowValue</i>	Decides which row to update

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/comment.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/comment.cpp

3.24 compareInfo Struct Reference

Public Attributes

- u8 **matchAll**
- u8 **matchOne**
- u8 **matchSet**
- u8 **noCase**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.25 CountCtx Struct Reference

Public Attributes

- i64 **n**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.26 With::Cte Struct Reference

Public Attributes

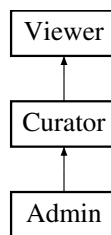
- char * **zName**
- ExprList * **pCols**
- Select * **pSelect**
- const char * **zCteErr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.27 Curator Class Reference

Inheritance diagram for Curator:



Public Member Functions

- **Curator** ()
Curator (p. 29) constructor for curator.
- **Curator** (int _userID, string _un, string _pw, string _priv, string _desc, string _selfPic)
Curator::Curator (p. 29) Constructor that initializes all class instance variables.

Additional Inherited Members

3.27.1 Constructor & Destructor Documentation

3.27.1.1 Curator() [1/2]

```
Curator::Curator ( )
```

Curator (p. 29) constructor for curator.

Curator::Curator (p. 29) Default constructor.

3.27.1.2 Curator() [2/2]

```
Curator::Curator (
    int _userID,
    string _un,
    string _pw,
    string _priv,
    string _desc,
    string _selfPic )
```

Curator::Curator (p. 29) Constructor that initializes all class instance variables.

Parameters

<code>_userID</code>	Curator (p. 29)'s ID
<code>_un</code>	Curator (p. 29)'s username
<code>_pw</code>	Curator (p. 29)'s password
<code>_priv</code>	Curator (p. 29)'s privelege level
<code>_desc</code>	Curator (p. 29)'s self description
<code>_selfPic</code>	Curator (p. 29)'s Profile Picture

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/curator.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/curator.cpp

3.28 DateTime Struct Reference

Public Attributes

- sqlite3_int64 **iJD**
- int **Y**
- int **M**
- int **D**
- int **h**
- int **m**
- int **tz**
- double **s**
- char **validJD**
- char **rawS**
- char **validYMD**
- char **validHMS**
- char **validTZ**
- char **tzSet**
- char **isError**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.29 Db Struct Reference

Public Attributes

- char * **zDbSName**
- Btree * **pBt**
- u8 **safety_level**
- u8 **bSyncSet**
- Schema * **pSchema**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.30 DbFixer Struct Reference

Public Attributes

- Parse * **pParse**
- Schema * **pSchema**
- u8 **bTemp**
- const char * **zDb**
- const char * **zType**
- const Token * **pName**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.31 DbQuoteStr Struct Reference

Public Attributes

- DbQuoteStr * **pNextStr**
- char **z** [8]

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

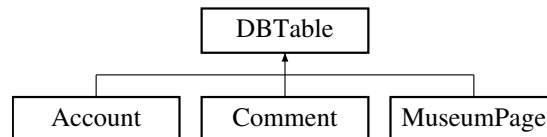
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.32 DBTable Class Reference

The **DBTable** (p. 32) class.

```
#include <dbtable.h>
```

Inheritance diagram for DBTable:



Public Member Functions

- **DBTable** (**DBTool** *db, std::string name)
- void **build_table** ()
- void **store_template_sql** ()
- void **store_exist_sql** ()
- virtual void **store_create_sql** ()
- void **store_drop_sql** ()
- void **store_size_sql** ()
- int **dbtemplate** ()
- bool **exist** ()
- bool **create** ()
- bool **drop** ()
- int **size** ()
- std::string **get_name** ()
- void **set_exists** ()
- void **unset_exists** ()
- void **set_size** (int cnt)

Protected Attributes

- std::string **table_name**
- **DBTool** * **curr_db**
- std::string **sql_template**
- std::string **sql_exist**
- std::string **sql_create**
- std::string **sql_drop**
- std::string **sql_size**
- std::string **sql_add_row**
- bool **table_exists**
- int **row_cnt**

3.32.1 Detailed Description

The **DBTable** (p. 32) class.

This class provides the parent functionality for a class that will be implementing the basic table functionality. The virtual methods also provide examples of the type of functionality that needs to be implemented in the inherited class.

There are also several functions that are not part of the class, but are required by the sqlite software function. They are generally named cb or "call back", indicating that they will be called by the sqlite software to provide functionality back to your code.

A called to `sqlite_exec` is detailed in the `exec` method, which is how these classes interfaces to the SQLite tool.

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/dbtable.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/dbtable.cpp

3.33 DBTool Class Reference

The Database management tool.

```
#include <dbtool.h>
```

Public Member Functions

- **DBTool** (std::string name)
- **DBTool** (const char *name)
- **DBTool** (std::string location, std::string name)
- **DBTool** (const char *location, const char *name)
- int **open_db** ()
- void **print** (std::ostream &sout)
- bool **db_open** ()
- **sqlite3** * **db_ref** ()

3.33.1 Detailed Description

The Database management tool.

This object represents the running **sqlite3** (p. 70) database instance, which is run as part of a library. This is the class that loads the sqlite file and will allow for the basic tools for manipulating that object. The Database object is essentially a specialized engine that implements the primary features of a SQL relational database as a local instance.

3.33.2 Member Function Documentation

3.33.2.1 open_db()

```
int DBTool::open_db ( )
```

Method that will open the provided database.

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/dbtool.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/dbtool.cpp

3.34 DistinctCtx Struct Reference

Public Attributes

- u8 **isTnct**
- u8 **eTnctType**
- int **tabTnct**
- int **addrTnct**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.35 et_info Struct Reference

Public Attributes

- char **fmttype**
- etByte **base**
- etByte **flags**
- etByte **type**
- etByte **charset**
- etByte **prefix**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.36 Expr Struct Reference

Public Attributes

- u8 **op**
- char **affExpr**
- u8 **op2**
- u32 **flags**
-
- union {
 char * **zToken**
 int **iValue**
 } **u**
- **Expr * pLeft**
- **Expr * pRight**
-
- union {
 ExprList * pList
Select * pSelect
 } **x**
- int **nHeight**
- int **iTable**
- ynVar **iColumn**
- i16 **iAgg**
- i16 **iRightJoinTable**
- **AgglInfo * pAgglInfo**
-
- union {
 Table * pTab
Window * pWin
 struct {
 int **iAddr**
 int **regReturn**
 } **sub**
 } **y**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.37 ExprList Struct Reference

Classes

- struct **ExprList_item**

Public Attributes

- int **nExpr**
- struct **ExprList::ExprList_item** **a** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.38 ExprList::ExprList_item Struct Reference

Public Attributes

- **Expr** * **pExpr**
- char * **zName**
- u8 **sortFlags**
- unsigned **eName**:2
- unsigned **done**:1
- unsigned **reusable**:1
- unsigned **bSorterRef**:1
- unsigned **bNulls**: 1
- union {
 struct {
 u16 **iOrderByCol**
 u16 **iAlias**
 } **x**
 int **iConstExprReg**
} **u**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.39 FileChunk Struct Reference

Public Attributes

- **FileChunk** * **pNext**
- u8 **zChunk** [8]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.40 FilePoint Struct Reference

Public Attributes

- sqlite3_int64 **iOffset**
- **FileChunk** * **pChunk**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.41 FKey Struct Reference

Classes

- struct **sColMap**

Public Attributes

- **Table** * **pFrom**
- **FKey** * **pNextFrom**
- char * **zTo**
- **FKey** * **pNextTo**
- **FKey** * **pPrevTo**
- int **nCol**
- u8 **isDeferred**
- u8 **aAction** [2]
- **Trigger** * **apTrigger** [2]
- struct **FKey::sColMap** **aCol** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.42 FrameBound Struct Reference

Public Attributes

- int **eType**
- **Expr** * **pExpr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.43 fts5_api Struct Reference

Public Attributes

- int **iVersion**
- int(* **xCreateTokenizer**)(fts5_api *pApi, const char *zName, void *pContext, fts5_tokenizer *pTokenizer, void(*xDestroy)(void *))
- int(* **xFindTokenizer**)(fts5_api *pApi, const char *zName, void **ppContext, fts5_tokenizer *pTokenizer)
- int(* **xCreateFunction**)(fts5_api *pApi, const char *zName, void *pContext, fts5_extension_function x↔Function, void(*xDestroy)(void *))

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.44 fts5_tokenizer Struct Reference

Public Attributes

- int(* **xCreate**)(void *, const char **azArg, int nArg, Fts5Tokenizer **ppOut)
- void(* **xDelete**)(Fts5Tokenizer *)
- int(* **xTokenize**)(Fts5Tokenizer *, void *pCtx, int flags, const char *pText, int nText, int(*xToken)(void *pCtx, int tflags, const char *pToken, int nToken, int iStart, int iEnd))

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.45 Fts5ExtensionApi Struct Reference

Public Attributes

- int **iVersion**
- void(* **xUserData**)(Fts5Context *)
- int(* **xColumnCount**)(Fts5Context *)
- int(* **xRowCount**)(Fts5Context *, sqlite3_int64 *pnRow)
- int(* **xColumnTotalSize**)(Fts5Context *, int iCol, sqlite3_int64 *pnToken)
- int(* **xTokenize**)(Fts5Context *, const char *pText, int nText, void *pCtx, int(*xToken)(void *, int, const char *, int, int, int))
- int(* **xPhraseCount**)(Fts5Context *)
- int(* **xPhraseSize**)(Fts5Context *, int iPhrase)
- int(* **xInstCount**)(Fts5Context *, int *pnInst)
- int(* **xInst**)(Fts5Context *, int ildx, int *piPhrase, int *piCol, int *piOff)
- sqlite3_int64(* **xRowid**)(Fts5Context *)
- int(* **xColumnText**)(Fts5Context *, int iCol, const char **pz, int *pn)
- int(* **xColumnSize**)(Fts5Context *, int iCol, int *pnToken)

- `int(* xQueryPhrase)(Fts5Context *, int iPhrase, void *pUserData, int*)(const Fts5ExtensionApi *, Fts5Context *, void *)`
- `int(* xSetAuxdata)(Fts5Context *, void *pAux, void(*xDelete)(void *))`
- `void *(* xGetAuxdata)(Fts5Context *, int bClear)`
- `int(* xPhraseFirst)(Fts5Context *, int iPhrase, Fts5Phraselter *, int *, int *)`
- `void(* xPhraseNext)(Fts5Context *, Fts5Phraselter *, int *piCol, int *piOff)`
- `int(* xPhraseFirstColumn)(Fts5Context *, int iPhrase, Fts5Phraselter *, int *)`
- `void(* xPhraseNextColumn)(Fts5Context *, Fts5Phraselter *, int *piCol)`

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.46 Fts5Phraselter Struct Reference

Public Attributes

- `const unsigned char * a`
- `const unsigned char * b`

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.47 FuncDef Struct Reference

Public Attributes

- `i8 nArg`
- `u32 funcFlags`
- `void * pUserData`
- `FuncDef * pNext`
- `void(* xSFunc)(sqlite3_context *, int, sqlite3_value **)`
- `void(* xFinalize)(sqlite3_context *)`
- `void(* xValue)(sqlite3_context *)`
- `void(* xInverse)(sqlite3_context *, int, sqlite3_value **)`
- `const char * zName`
- `union {
 FuncDef * pHash
 FuncDestructor * pDestructor
} u`

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.48 FuncDefHash Struct Reference

Public Attributes

- **FuncDef** * **a** [SQLITE_FUNC_HASH_SZ]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.49 FuncDestructor Struct Reference

Public Attributes

- int **nRef**
- void(* **xDestroy**)(void *)
- void * **pUserData**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.50 Hash Struct Reference

Classes

- struct **_ht**

Public Attributes

- unsigned int **htsize**
- unsigned int **count**
- **HashElem** * **first**
- struct **Hash::_ht** * **ht**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.51 HashElem Struct Reference

Public Attributes

- **HashElem** * **next**
- **HashElem** * **prev**
- void * **data**
- const char * **pKey**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.52 HiddenIndexInfo Struct Reference

Public Attributes

- **WhereClause** * **pWC**
- **Parse** * **pParse**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.53 IdList Struct Reference

Classes

- struct **IdList_item**

Public Attributes

- struct **IdList::IdList_item** * **a**
- int **nId**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.54 IdList::IdList_item Struct Reference

Public Attributes

- char * **zName**
- int **idx**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.55 IdxCover Struct Reference

Public Attributes

- Index * **pIdx**
- int **iCur**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.56 IdxExprTrans Struct Reference

Public Attributes

- Expr * **pIdxExpr**
- int **iTabCur**
- int **ildxCur**
- int **ildxCol**
- int **iTabCol**
- WhereInfo * **pWInfo**
- sqlite3 * **db**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.57 Incrblob Struct Reference

Public Attributes

- int **nByte**
- int **iOffset**
- u16 **iCol**
- **BtCursor** * **pCsr**
- sqlite3_stmt * **pStmt**
- **sqlite3** * **db**
- char * **zDb**
- **Table** * **pTab**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.58 IncrMerger Struct Reference

Public Attributes

- **SortSubtask** * **pTask**
- **MergeEngine** * **pMerger**
- i64 **iStartOff**
- int **mxSz**
- int **bEof**
- int **bUseThread**
- **SorterFile** **aFile** [2]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.59 Index Struct Reference

Public Attributes

- char * **zName**
- i16 * **aiColumn**
- LogEst * **aiRowLogEst**
- **Table** * **pTable**
- char * **zColAff**
- **Index** * **pNext**
- **Schema** * **pSchema**
- u8 * **aSortOrder**
- const char ** **azColl**
- **Expr** * **pPartIdxWhere**
- **ExprList** * **aColExpr**

- int **tnum**
- LogEst **szIdxRow**
- u16 **nKeyCol**
- u16 **nColumn**
- u8 **onError**
- unsigned **idxType**:2
- unsigned **bUnordered**:1
- unsigned **uniqNotNull**:1
- unsigned **isResized**:1
- unsigned **isCovering**:1
- unsigned **noSkipScan**:1
- unsigned **hasStat1**:1
- unsigned **bNoQuery**:1
- unsigned **bAscKeyBug**:1
- unsigned **bHasVCol**:1
- Bitmask **colNotIdxed**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.60 IndexSample Struct Reference

Public Attributes

- void * **p**
- int **n**
- tRowcnt * **anEq**
- tRowcnt * **anLt**
- tRowcnt * **anDLt**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.61 InitData Struct Reference

Public Attributes

- **sqlite3** * **db**
- char ** **pzErrMsg**
- int **iDb**
- int **rc**
- u32 **mInitFlags**
- u32 **nInitRow**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.62 IntegrityCk Struct Reference

Public Attributes

- **BtShared** * **pBt**
- **Pager** * **pPager**
- u8 * **aPgRef**
- Pgno **nPage**
- int **mxErr**
- int **nErr**
- int **mallocFailed**
- const char * **zPfx**
- int **v1**
- int **v2**
- **StrAccum** **errMsg**
- u32 * **heap**
- **sqlite3** * **db**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.63 KeyInfo Struct Reference

Public Attributes

- u32 **nRef**
- u8 **enc**
- u16 **nKeyField**
- u16 **nAllField**
- **sqlite3** * **db**
- u8 * **aSortFlags**
- **CollSeq** * **aColl** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.64 LastValueCtx Struct Reference

Public Attributes

- **sqlite3_value** * **pVal**
- int **nVal**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.65 Lookaside Struct Reference

Public Attributes

- u32 **bDisable**
- u16 **sz**
- u16 **szTrue**
- u8 **bMalloced**
- u32 **nSlot**
- u32 **anStat** [3]
- **LookasideSlot** * **pInit**
- **LookasideSlot** * **pFree**
- **LookasideSlot** * **pSmallInit**
- **LookasideSlot** * **pSmallFree**
- void * **pMiddle**
- void * **pStart**
- void * **pEnd**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.66 LookasideSlot Struct Reference

Public Attributes

- **LookasideSlot** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.67 MemJournal Struct Reference

Public Attributes

- const **sqlite3_io_methods** * **pMethod**
- int **nChunkSize**
- int **nSpill**
- int **nSize**
- **FileChunk** * **pFirst**
- **FilePoint** **endpoint**
- **FilePoint** **readpoint**
- int **flags**
- **sqlite3_vfs** * **pVfs**
- const char * **zJournal**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.68 MemPage Struct Reference

Public Attributes

- u8 **isInit**
- u8 **bBusy**
- u8 **intKey**
- u8 **intKeyLeaf**
- Pgno **pgno**
- u8 **leaf**
- u8 **hdrOffset**
- u8 **childPtrSize**
- u8 **max1bytePayload**
- u8 **nOverflow**
- u16 **maxLocal**
- u16 **minLocal**
- u16 **cellOffset**
- int **nFree**
- u16 **nCell**
- u16 **maskPage**
- u16 **aiOvfl** [4]
- u8 * **apOvfl** [4]
- **BtShared** * **pBt**
- u8 * **aData**
- u8 * **aDataEnd**
- u8 * **aCellIdx**
- u8 * **aDataOfst**
- **DbPage** * **pDbPage**
- u16(* **xCellSize**)(**MemPage** *, u8 *)
- void(* **xParseCell**)(**MemPage** *, u8 *, **CellInfo** *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.69 sqlite3_value::MemValue Union Reference

Public Attributes

- double **r**
- i64 **i**
- int **nZero**
- const char * **zPType**
- **FuncDef** * **pDef**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.70 MergeEngine Struct Reference

Public Attributes

- int **nTree**
- **SortSubtask** * **pTask**
- int * **aTree**
- **PmaReader** * **aReadr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.71 Module Struct Reference

Public Attributes

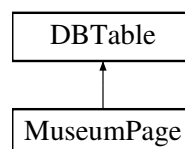
- const **sqlite3_module** * **pModule**
- const char * **zName**
- int **nRefModule**
- void * **pAux**
- void(* **xDestroy**)(void *)
- **Table** * **pEpoTab**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.72 MuseumPage Class Reference

Inheritance diagram for MuseumPage:



Public Member Functions

- void **store_create_sql** ()
store_create_sql sql statement for creates on in table
- virtual void **store_add_row_sql** ()
store_add_row_sql sql statement for row add on ining to table
- bool **add_row** (int pID, string n, string d, string p, int c)
add_row adds row to database
- vector< string > **select_elements_sql** (string)
select_elements_sql selects all elements of specified column
- vector< string > **select_cell** (string, string, string)
select_cell selects info from given cell
- vector< string > **select_cell_search** (string, string, string)
select_cell_search Uses LIKE SQL query to find valid entries in table
- **MuseumPage** * **getMuseumPageFromId** (int id)
getMuseumPageFromId returns instance of a museum page from id
- bool **addMuseum** (int, string, string, string, int, string)
MuseumPage::addMuseum (p. 51) Checks formatting of new museum entry and adds to database if everything is correct.
- vector< string > **getMuseumChildrenId** (int)
MuseumPage::getMuseumChildrenId (p. 52) Retrieves all of the museum IDs of a given parent's children museums.
- vector< string > **getMuseumChildrenNames** (int)
MuseumPage::getMuseumChildrenNames (p. 52) Retrieves all museum names of a give parent's children museums.
- vector< string > **getMuseumByCuratorId** (int)
MuseumPage::getMuseumByCuratorId (p. 51) Retrieves all of the names of the museum's given a curator's ID.
- int **getMuseumId** (string mname)
MuseumPage::getMuseumId (p. 53) Finds the museum's ID given the museum's name.
- int **getMuseumSearchId** (string mname)
MuseumPage::getMuseumSearchId (p. 53) Finds the museum's ID given the museum's name and specifically uses the like keyword to find similar searches to the inputted query.
- void **remove_entries** ()
MuseumPage::remove_row Removes a row given a column name and key value.
- **MuseumPage** (**DBTool** *db, std::string name, bool _isTesting)
MuseumPage::MuseumPage (p. 50) Constructor that initializes database class tools.
- **MuseumPage** ()
MuseumPage (p. 48) Constructor for this page.
- **MuseumPage** (int i)
MuseumPage (p. 48) represents one page for this museum.
- int **getId** ()
- void **setId** (int)
- string **getDescription** ()
- void **setDescription** (string)
- string **getName** ()
- void **setName** (string)
- string **getPicString** ()
- void **setPicString** (string)
- int **getParentId** ()
- void **setParentId** (int)
- int **getCuratorId** ()
- void **setCuratorId** (int)
- void **setDB** (**sqlite3** *)

Public Attributes

- `std::string sql_select`
sql_select used for select_cell and select_elements_sql methods

Additional Inherited Members

3.72.1 Constructor & Destructor Documentation

3.72.1.1 `MuseumPage()` [1/3]

```
MuseumPage::MuseumPage (
    DBTool * _db,
    std::string name,
    bool _isTesting )
```

MuseumPage::MuseumPage (p. 50) Constructor that initializes database class tools.

Parameters

<code>_db</code>	Reference to DBTool (p. 33) class
<code>name</code>	Name of the Database table

3.72.1.2 `MuseumPage()` [2/3]

```
MuseumPage::MuseumPage ( )
```

MuseumPage (p. 48) Constructor for this page.

MuseumPage::MuseumPage (p. 50) Default Constructor.

3.72.1.3 `MuseumPage()` [3/3]

```
MuseumPage::MuseumPage (
    int i )
```

MuseumPage (p. 48) represents one page for this museum.

MuseumPage::MuseumPage (p. 50) Constructor that initializes the ID of museum.

Parameters

<code>i</code>	the id for this page
<code>i</code>	The desired ID

3.72.2 Member Function Documentation

3.72.2.1 add_row()

```
bool MuseumPage::add_row (
    int pID,
    string n,
    string d,
    string p,
    int c )
```

add_row adds row to database

Returns

bool

3.72.2.2 addMuseum()

```
bool MuseumPage::addMuseum (
    int _pID,
    string _n,
    string _d,
    string _p,
    int _c,
    string priv )
```

MuseumPage::addMuseum (p. 51) Checks formatting of new museum entry and adds to database if everything is correct.

Parameters

<i>_pID</i>	The Parent Museum's ID
<i>_n</i>	The name of the new museum
<i>_d</i>	The description of the new museum
<i>_p</i>	The picture file name of the new museum
<i>_c</i>	The ID of the curator that created this new museum
<i>priv</i>	The privelege of the user

Returns

Whether the addition of the museum was successful

3.72.2.3 `getMuseumByCuratorId()`

```
vector< string > MuseumPage::getMuseumByCuratorId (
    int CID )
```

MuseumPage::getMuseumByCuratorId (p. 51) Retrieves all of the names of the museum's given a curator's ID.

Parameters

<i>CID</i>	Curator (p. 29)'s unique ID
------------	------------------------------------

Returns

A vector full of museum names created by the curator

3.72.2.4 `getMuseumChildrenId()`

```
vector< string > MuseumPage::getMuseumChildrenId (
    int parent )
```

MuseumPage::getMuseumChildrenId (p. 52) Retrieves all of the museum IDs of a given parent's children museums.

Parameters

<i>parent</i>	The ID of the parent museum
---------------	-----------------------------

Returns

A vector full of id's of children museum

3.72.2.5 `getMuseumChildrenNames()`

```
vector< string > MuseumPage::getMuseumChildrenNames (
    int parent )
```

MuseumPage::getMuseumChildrenNames (p. 52) Retrieves all museum names of a give parent's children museums.

Parameters

<i>parent</i>	The ID of the parent museum
---------------	-----------------------------

Returns

A vector full of names of children museum

3.72.2.6 getMuseumId()

```
int MuseumPage::getMuseumId (
    string mname )
```

MuseumPage::getMuseumId (p. 53) Finds the museum's ID given the museum's name.

Parameters

<i>mname</i>	The museum's name
--------------	-------------------

Returns

The museum's unique ID

3.72.2.7 getMuseumPageFromId()

```
MuseumPage * MuseumPage::getMuseumPageFromId (
    int _id )
```

getMuseumPageFromId returns instance of a museum page from id

MuseumPage::getMuseumPageFromId (p. 53) Searches for museum by ID and returns all data related to it.

Parameters

<i>id</i>	the id of the museum
-----------	----------------------

Returns**Parameters**

↔	Museum ID
↔	
<i>id</i>	

Returns

Reference to Museum ID object that contains all needed data

3.72.2.8 getMuseumSearchId()

```
int MuseumPage::getMuseumSearchId (
    string mname )
```

MuseumPage::getMuseumSearchId (p. 53) Finds the museum's ID given the museum's name and specifically uses the like keyword to find similar searches to the inputted query.

Parameters

<i>mname</i>	The museum's name
--------------	-------------------

Returns

The museum's unique ID

3.72.2.9 remove_entries()

```
void MuseumPage::remove_entries ( )
```

MuseumPage::remove_row Removes a row given a column name and key value.

Parameters

<i>n</i>	The key value of the row
<i>cn</i>	The column name to match the key

3.72.2.10 select_cell_search()

```
vector< string > MuseumPage::select_cell_search (
    string whichRow,
    string row,
    string col )
```

select_cell_search Uses LIKE SQL query to find valid entries in table

MuseumPage::select_cell_search (p. 54) Looks for an element in the table using the LIKE keyword.

Returns

List of possible queries

Parameters

<i>whichRow</i>	The pattern in which the LIKE keyword will be compared to
<i>row</i>	The row that is desired
<i>col</i>	The column that is desired

Returns

The result of the SQL query

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/museumpage.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/museumpage.cpp

3.73 NameContext Struct Reference

Public Attributes

- **Parse** * **pParse**
- **SrcList** * **pSrcList**
- union {
 ExprList * **pEList**
 AgglInfo * **pAgglInfo**
 Upsert * **pUpsert**
} **uNC**
- **NameContext** * **pNext**
- int **nRef**
- int **nErr**
- int **ncFlags**
- **Select** * **pWinSelect**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.74 NthValueCtx Struct Reference

Public Attributes

- i64 **nStep**
- **sqlite3_value** * **pValue**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.75 NtileCtx Struct Reference

Public Attributes

- i64 **nTotal**
- i64 **nParam**
- i64 **iRow**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.76 VdbeOp::p4union Union Reference

Public Attributes

- int **i**
- void * **p**
- char * **z**
- i64 * **pl64**
- double * **pReal**
- **FuncDef** * **pFunc**
- **sqlite3_context** * **pCtx**
- **CollSeq** * **pColl**
- **Mem** * **pMem**
- **VTable** * **pVtab**
- **KeyInfo** * **pKeyInfo**
- int * **ai**
- **SubProgram** * **pProgram**
- **Table** * **pTab**
- int(* **xAdvance**)(**BtCursor** *, int)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.77 Pager Struct Reference

Public Attributes

- **sqlite3_vfs** * **pVfs**
- u8 **exclusiveMode**
- u8 **journalMode**
- u8 **useJournal**
- u8 **noSync**
- u8 **fullSync**
- u8 **extraSync**
- u8 **syncFlags**

- u8 **walSyncFlags**
- u8 **tempFile**
- u8 **noLock**
- u8 **readOnly**
- u8 **memDb**
- u8 **eState**
- u8 **eLock**
- u8 **changeCountDone**
- u8 **setMaster**
- u8 **doNotSpill**
- u8 **subJInMemory**
- u8 **bUseFetch**
- u8 **hasHeldSharedLock**
- Pgno **dbSize**
- Pgno **dbOrigSize**
- Pgno **dbFileSize**
- Pgno **dbHintSize**
- int **errCode**
- int **nRec**
- u32 **cksumInit**
- u32 **nSubRec**
- Bitvec * **plnJournal**
- sqlite3_file * **fd**
- sqlite3_file * **jfd**
- sqlite3_file * **sjfd**
- i64 **journalOff**
- i64 **journalHdr**
- sqlite3_backup * **pBackup**
- PagerSavepoint * **aSavepoint**
- int **nSavepoint**
- u32 **iDataVersion**
- char **dbFileVers** [16]
- int **nMmapOut**
- sqlite3_int64 **szMmap**
- PgHdr * **pMmapFreelist**
- u16 **nExtra**
- i16 **nReserve**
- u32 **vfsFlags**
- u32 **sectorSize**
- int **pageSize**
- Pgno **mxPgno**
- i64 **journalSizeLimit**
- char * **zFilename**
- char * **zJournal**
- int(* **xBusyHandler**)(void *)
- void * **pBusyHandlerArg**
- int **aStat** [4]
- void(* **xReinit**)(DbPage *)
- int(* **xGet**)(Pager *, Pgno, DbPage **, int)
- char * **pTmpSpace**
- PCache * **pPCache**
- Wal * **pWal**
- char * **zWal**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.78 PagerSavepoint Struct Reference

Public Attributes

- i64 **iOffset**
- i64 **iHdrOffset**
- **Bitvec** * **pInSavepoint**
- Pgno **nOrig**
- Pgno **iSubRec**
- u32 **aWalData** [WAL_SAVEPOINT_NDATA]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.79 Parse Struct Reference

Public Attributes

- **sqlite3** * **db**
- char * **zErrMsg**
- **Vdbe** * **pVdbe**
- int **rc**
- u8 **colNamesSet**
- u8 **checkSchema**
- u8 **nested**
- u8 **nTempReg**
- u8 **isMultiWrite**
- u8 **mayAbort**
- u8 **hasCompound**
- u8 **okConstFactor**
- u8 **disableLookaside**
- u8 **disableVtab**
- int **nRangeReg**
- int **iRangeReg**
- int **nErr**
- int **nTab**
- int **nMem**
- int **szOpAlloc**
- int **iSelfTab**
- int **nLabel**
- int **nLabelAlloc**
- int * **aLabel**
- **ExprList** * **pConstExpr**
- **Token** **constraintName**
- yDbMask **writeMask**
- yDbMask **cookieMask**
- int **regRowid**
- int **regRoot**
- int **nMaxArg**
- int **nSelect**

- int **nTableLock**
- **TableLock** * **aTableLock**
- **AutoincInfo** * **pAinc**
- **Parse** * **pToplevel**
- **Table** * **pTriggerTab**
- **Parse** * **pParentParse**
- int **addrCrTab**
- u32 **nQueryLoop**
- u32 **oldmask**
- u32 **newmask**
- u8 **eTriggerOp**
- u8 **eOrconf**
- u8 **disableTriggers**
- int **aTempReg** [8]
- **Token** **sNameToken**
- **Token** **sLastToken**
- ynVar **nVar**
- u8 **iPkSortOrder**
- u8 **explain**
- u8 **eParseMode**
- int **nVtabLock**
- int **nHeight**
- int **addrExplain**
- VList * **pVList**
- **Vdbe** * **pReprepare**
- const char * **zTail**
- **Table** * **pNewTable**
- **Index** * **pNewIndex**
- **Trigger** * **pNewTrigger**
- const char * **zAuthContext**
- **Token** **sArg**
- **Table** ** **apVtabLock**
- **Table** * **pZombieTab**
- **TriggerPrg** * **pTriggerPrg**
- **With** * **pWith**
- **With** * **pWithToFree**
- **RenameToken** * **pRename**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.80 PCache Struct Reference

Public Attributes

- **PgHdr** * **pDirty**
- **PgHdr** * **pDirtyTail**
- **PgHdr** * **pSynced**
- int **nRefSum**
- int **szCache**

- int **szSpill**
- int **szPage**
- int **szExtra**
- u8 **bPurgeable**
- u8 **eCreate**
- int(* **xStress**)(void *, **PgHdr** *)
- void * **pStress**
- sqlite3_pcache * **pCache**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.81 PCache1 Struct Reference

Public Attributes

- **PGroup** * **pGroup**
- unsigned int * **pnPurgeable**
- int **szPage**
- int **szExtra**
- int **szAlloc**
- int **bPurgeable**
- unsigned int **nMin**
- unsigned int **nMax**
- unsigned int **n90pct**
- unsigned int **iMaxKey**
- unsigned int **nPurgeableDummy**
- unsigned int **nRecyclable**
- unsigned int **nPage**
- unsigned int **nHash**
- **PgHdr1** ** **apHash**
- **PgHdr1** * **pFree**
- void * **pBulk**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.82 PgFreeslot Struct Reference

Public Attributes

- **PgFreeslot** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.83 PgHdr Struct Reference

Public Attributes

- **sqlite3_pcache_page** * **pPage**
- void * **pData**
- void * **pExtra**
- **PCache** * **pCache**
- **PgHdr** * **pDirty**
- **Pager** * **pPager**
- Pgno **pgno**
- u16 **flags**
- i16 **nRef**
- **PgHdr** * **pDirtyNext**
- **PgHdr** * **pDirtyPrev**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.84 PgHdr1 Struct Reference

Public Attributes

- **sqlite3_pcache_page** **page**
- unsigned int **iKey**
- u16 **isBulkLocal**
- u16 **isAnchor**
- **PgHdr1** * **pNext**
- **PCache1** * **pCache**
- **PgHdr1** * **pLruNext**
- **PgHdr1** * **pLruPrev**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.85 PGroup Struct Reference

Public Attributes

- **sqlite3_mutex** * **mutex**
- unsigned int **nMaxPage**
- unsigned int **nMinPage**
- unsigned int **mxPinned**
- unsigned int **nPurgeable**
- **PgHdr1** **lru**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.86 PmaReader Struct Reference

Public Attributes

- i64 **iReadOff**
- i64 **iEof**
- int **nAlloc**
- int **nKey**
- **sqlite3_file** * **pFd**
- u8 * **aAlloc**
- u8 * **aKey**
- u8 * **aBuffer**
- int **nBuffer**
- u8 * **aMap**
- **IncrMerger** * **pIncr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.87 PmaWriter Struct Reference

Public Attributes

- int **eFWErr**
- u8 * **aBuffer**
- int **nBuffer**
- int **iBufStart**
- int **iBufEnd**
- i64 **iWriteOff**
- **sqlite3_file** * **pFd**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.88 PragmaName Struct Reference

Public Attributes

- const char *const **zName**
- u8 **ePragTyp**
- u8 **mPragFlg**
- u8 **iPragCName**
- u8 **nPragCName**
- u64 **iArg**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.89 PragmaVtab Struct Reference

Public Attributes

- **sqlite3_vtab** base
- **sqlite3** * db
- const **PragmaName** * pName
- u8 nHidden
- u8 iHidden

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.90 PragmaVtabCursor Struct Reference

Public Attributes

- **sqlite3_vtab_cursor** base
- sqlite3_stmt * pPragma
- sqlite_int64 iRowid
- char * azArg [2]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.91 PreUpdate Struct Reference

Public Attributes

- **Vdbe** * v
- **VdbeCursor** * pCsr
- int op
- u8 * aRecord
- **KeyInfo** keyinfo
- **UnpackedRecord** * pUnpacked
- **UnpackedRecord** * pNewUnpacked
- int iNewReg
- i64 iKey1
- i64 iKey2
- **Mem** * aNew
- **Table** * pTab
- **Index** * pPk

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.92 PrintfArguments Struct Reference

Public Attributes

- int **nArg**
- int **nUsed**
- **sqlite3_value** ** **apArg**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.93 RenameCtx Struct Reference

Public Attributes

- **RenameToken** * **pList**
- int **nList**
- int **iCol**
- **Table** * **pTab**
- const char * **zOld**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.94 RenameToken Struct Reference

Public Attributes

- void * **p**
- **Token** **t**
- **RenameToken** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.95 ReusableSpace Struct Reference

Public Attributes

- u8 * **pSpace**
- sqlite3_int64 **nFree**
- sqlite3_int64 **nNeeded**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.96 RowLoadInfo Struct Reference

Public Attributes

- int **regResult**
- u8 **ecelFlags**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.97 RowSet Struct Reference

Public Attributes

- struct **RowSetChunk** * **pChunk**
- **sqlite3** * **db**
- struct **RowSetEntry** * **pEntry**
- struct **RowSetEntry** * **pLast**
- struct **RowSetEntry** * **pFresh**
- struct **RowSetEntry** * **pForest**
- u16 **nFresh**
- u16 **rsFlags**
- int **iBatch**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.98 RowSetChunk Struct Reference

Public Attributes

- struct **RowSetChunk** * **pNextChunk**
- struct **RowSetEntry** **aEntry** [ROWSET_ENTRY_PER_CHUNK]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.99 RowSetEntry Struct Reference

Public Attributes

- i64 **v**
- struct **RowSetEntry** * **pRight**
- struct **RowSetEntry** * **pLeft**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.100 Savepoint Struct Reference

Public Attributes

- char * **zName**
- i64 **nDeferredCons**
- i64 **nDeferredImmCons**
- **Savepoint** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.101 ScanStatus Struct Reference

Public Attributes

- int **addrExplain**
- int **addrLoop**
- int **addrVisit**
- int **iSelectID**
- LogEst **nEst**
- char * **zName**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.102 Schema Struct Reference

Public Attributes

- int **schema_cookie**
- int **iGeneration**
- Hash **tblHash**
- Hash **idxHash**
- Hash **trigHash**
- Hash **fkeyHash**
- Table * **pSeqTab**
- u8 **file_format**
- u8 **enc**
- u16 **schemaFlags**
- int **cache_size**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.103 FKey::sColMap Struct Reference

Public Attributes

- int **iFrom**
- char * **zCol**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.104 Select Struct Reference

Public Attributes

- u8 **op**
- LogEst **nSelectRow**
- u32 **selFlags**
- int **iLimit**
- int **iOffset**
- u32 **selId**
- int **addrOpenEphm** [2]
- ExprList * **pEList**
- SrcList * **pSrc**
- Expr * **pWhere**
- ExprList * **pGroupBy**
- Expr * **pHaving**
- ExprList * **pOrderBy**
- Select * **pPrior**
- Select * **pNext**
- Expr * **pLimit**
- With * **pWith**
- Window * **pWin**
- Window * **pWinDefn**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.105 SelectDest Struct Reference

Public Attributes

- u8 **eDest**
- int **iSDParm**
- int **iSdst**
- int **nSdst**
- char * **zAffSdst**
- **ExprList** * **pOrderBy**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.106 SortCtx Struct Reference

Public Attributes

- **ExprList** * **pOrderBy**
- int **nOBSat**
- int **iECursor**
- int **regReturn**
- int **labelBkOut**
- int **addrSortIndex**
- int **labelDone**
- int **labelOBLOpt**
- u8 **sortFlags**
- struct **RowLoadInfo** * **pDeferredRowLoad**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.107 SorterFile Struct Reference

Public Attributes

- **sqlite3_file** * **pFd**
- i64 **iEof**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.108 SorterList Struct Reference

Public Attributes

- **SorterRecord * pList**
- **u8 * aMemory**
- **int szPMA**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.109 SorterRecord Struct Reference

Public Attributes

- **int nVal**
- union {
 SorterRecord * pNext
 int iNext
} **u**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.110 SortSubtask Struct Reference

Public Attributes

- **SQLiteThread * pThread**
- **int bDone**
- **VdbeSorter * pSorter**
- **UnpackedRecord * pUnpacked**
- **SorterList list**
- **int nPMA**
- **SorterCompare xCompare**
- **SorterFile file**
- **SorterFile file2**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.111 sqlite3 Struct Reference

Classes

- struct **sqlite3InitInfo**

Public Attributes

- **sqlite3_vfs** * pVfs
- struct **Vdbe** * pVdbe
- **CollSeq** * pDfltColl
- **sqlite3_mutex** * mutex
- **Db** * aDb
- int **nDb**
- u32 **mDbFlags**
- u64 **flags**
- i64 **lastRowid**
- i64 **szMmap**
- u32 **nSchemaLock**
- unsigned int **openFlags**
- int **errCode**
- int **errMask**
- int **iSysErrno**
- u16 **dbOptFlags**
- u8 **enc**
- u8 **autoCommit**
- u8 **temp_store**
- u8 **mallocFailed**
- u8 **bBenignMalloc**
- u8 **dfltLockMode**
- signed char **nextAutovac**
- u8 **suppressErr**
- u8 **vtabOnConflict**
- u8 **isTransactionSavepoint**
- u8 **mTrace**
- u8 **noSharedCache**
- u8 **nSqlExec**
- int **nextPagesize**
- u32 **magic**
- int **nChange**
- int **nTotalChange**
- int **aLimit** [SQLITE_N_LIMIT]
- int **nMaxSorterMmap**
- struct **sqlite3::sqlite3InitInfo** init
- int **nVdbeActive**
- int **nVdbeRead**
- int **nVdbeWrite**
- int **nVdbeExec**
- int **nVDestroy**
- int **nExtension**
- void ** **aExtension**
- int(* **xTrace**)(u32, void *, void *, void *)
- void * **pTraceArg**

- void(* **xProfile**)(void *, const char *, u64)
- void * **pProfileArg**
- void * **pCommitArg**
- int(* **xCommitCallback**)(void *)
- void * **pRollbackArg**
- void(* **xRollbackCallback**)(void *)
- void * **pUpdateArg**
- void(* **xUpdateCallback**)(void *, int, const char *, const char *, sqlite_int64)
- **Parse** * **pParse**
- int(* **xWalCallback**)(void *, **sqlite3** *, const char *, int)
- void * **pWalArg**
- void(* **xCollNeeded**)(void *, **sqlite3** *, int eTextRep, const char *)
- void(* **xCollNeeded16**)(void *, **sqlite3** *, int eTextRep, const void *)
- void * **pCollNeededArg**
- **sqlite3_value** * **pErr**
-
- union {
 volatile int **isInterrupted**
 double **notUsed1**
 } **u1**
- **Lookaside** **lookaside**
- **sqlite3_xauth** **xAuth**
- void * **pAuthArg**
- int(* **xProgress**)(void *)
- void * **pProgressArg**
- unsigned **nProgressOps**
- int **nVTrans**
- **Hash** **aModule**
- **VtabCtx** * **pVtabCtx**
- **VTable** ** **aVTrans**
- **VTable** * **pDisconnect**
- **Hash** **aFunc**
- **Hash** **aCollSeq**
- **BusyHandler** **busyHandler**
- **Db** **aDbStatic** [2]
- **Savepoint** * **pSavepoint**
- int **busyTimeout**
- int **nSavepoint**
- int **nStatement**
- i64 **nDeferredCons**
- i64 **nDeferredImmCons**
- int * **pnBytesFreed**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.112 sqlite3_api_routines Struct Reference

Public Attributes

- void **(* aggregate_context)**(**sqlite3_context** *, int nBytes)
- int **(* aggregate_count)**(**sqlite3_context** *)
- int **(* bind_blob)**(sqlite3_stmt *, int, const void *, int n, void(*) (void *))
- int **(* bind_double)**(sqlite3_stmt *, int, double)
- int **(* bind_int)**(sqlite3_stmt *, int, int)
- int **(* bind_int64)**(sqlite3_stmt *, int, sqlite_int64)
- int **(* bind_null)**(sqlite3_stmt *, int)
- int **(* bind_parameter_count)**(sqlite3_stmt *)
- int **(* bind_parameter_index)**(sqlite3_stmt *, const char *zName)
- const char **(* bind_parameter_name)**(sqlite3_stmt *, int)
- int **(* bind_text)**(sqlite3_stmt *, int, const char *, int n, void(*) (void *))
- int **(* bind_text16)**(sqlite3_stmt *, int, const void *, int, void(*) (void *))
- int **(* bind_value)**(sqlite3_stmt *, int, const **sqlite3_value** *)
- int **(* busy_handler)**(**sqlite3** *, int(*) (void *, int), void *)
- int **(* busy_timeout)**(**sqlite3** *, int ms)
- int **(* changes)**(**sqlite3** *)
- int **(* close)**(**sqlite3** *)
- int **(* collation_needed)**(**sqlite3** *, void *, void(*) (void *, **sqlite3** *, int eTextRep, const char *))
- int **(* collation_needed16)**(**sqlite3** *, void *, void(*) (void *, **sqlite3** *, int eTextRep, const void *))
- const void **(* column_blob)**(sqlite3_stmt *, int iCol)
- int **(* column_bytes)**(sqlite3_stmt *, int iCol)
- int **(* column_bytes16)**(sqlite3_stmt *, int iCol)
- int **(* column_count)**(sqlite3_stmt *pStmt)
- const char **(* column_database_name)**(sqlite3_stmt *, int)
- const void **(* column_database_name16)**(sqlite3_stmt *, int)
- const char **(* column_decltype)**(sqlite3_stmt *, int i)
- const void **(* column_decltype16)**(sqlite3_stmt *, int)
- double **(* column_double)**(sqlite3_stmt *, int iCol)
- int **(* column_int)**(sqlite3_stmt *, int iCol)
- sqlite_int64 **(* column_int64)**(sqlite3_stmt *, int iCol)
- const char **(* column_name)**(sqlite3_stmt *, int)
- const void **(* column_name16)**(sqlite3_stmt *, int)
- const char **(* column_origin_name)**(sqlite3_stmt *, int)
- const void **(* column_origin_name16)**(sqlite3_stmt *, int)
- const char **(* column_table_name)**(sqlite3_stmt *, int)
- const void **(* column_table_name16)**(sqlite3_stmt *, int)
- const unsigned char **(* column_text)**(sqlite3_stmt *, int iCol)
- const void **(* column_text16)**(sqlite3_stmt *, int iCol)
- int **(* column_type)**(sqlite3_stmt *, int iCol)
- **sqlite3_value** **(* column_value)**(sqlite3_stmt *, int iCol)
- void **(* commit_hook)**(**sqlite3** *, int(*) (void *, void *)
- int **(* complete)**(const char *sql)
- int **(* complete16)**(const void *sql)
- int **(* create_collation)**(**sqlite3** *, const char *, int, void *, int(*) (void *, int, const void *, int, const void *))
- int **(* create_collation16)**(**sqlite3** *, const void *, int, void *, int(*) (void *, int, const void *, int, const void *))
- int **(* create_function)**(**sqlite3** *, const char *, int, int, void *, void(*) (xFunc)(**sqlite3_context** *, int, **sqlite3_value** **), void(*) (xStep)(**sqlite3_context** *, int, **sqlite3_value** **), void(*) (xFinal)(**sqlite3_context** *))

- `int(* create_function16)(sqlite3 *, const void *, int, int, void *, void(*xFunc)(sqlite3_context *, int, sqlite3_value **), void(*xStep)(sqlite3_context *, int, sqlite3_value **), void(*xFinal)(sqlite3_context *))`
- `int(* create_module)(sqlite3 *, const char *, const sqlite3_module *, void *)`
- `int(* data_count)(sqlite3_stmt *pStmt)`
- `sqlite3 *(* db_handle)(sqlite3_stmt *)`
- `int(* declare_vtab)(sqlite3 *, const char *)`
- `int(* enable_shared_cache)(int)`
- `int(* errcode)(sqlite3 *db)`
- `const char *(* errmsg)(sqlite3 *)`
- `const void *(* errmsg16)(sqlite3 *)`
- `int(* exec)(sqlite3 *, const char *, sqlite3_callback, void *, char **)`
- `int(* expired)(sqlite3_stmt *)`
- `int(* finalize)(sqlite3_stmt *pStmt)`
- `void(* free)(void *)`
- `void(* free_table)(char **result)`
- `int(* get_autocommit)(sqlite3 *)`
- `void *(* get_auxdata)(sqlite3_context *, int)`
- `int(* get_table)(sqlite3 *, const char *, char ***, int *, int *, char **)`
- `int(* global_recover)(void)`
- `void(* interruptx)(sqlite3 *)`
- `sqlite_int64(* last_insert_rowid)(sqlite3 *)`
- `const char *(* libversion)(void)`
- `int(* libversion_number)(void)`
- `void *(* malloc)(int)`
- `char *(* mprintf)(const char *,...)`
- `int(* open)(const char *, sqlite3 **)`
- `int(* open16)(const void *, sqlite3 **)`
- `int(* prepare)(sqlite3 *, const char *, int, sqlite3_stmt **, const char **)`
- `int(* prepare16)(sqlite3 *, const void *, int, sqlite3_stmt **, const void **)`
- `void *(* profile)(sqlite3 *, void(*) (void *, const char *, sqlite_uint64), void *)`
- `void(* progress_handler)(sqlite3 *, int, int(*) (void *), void *)`
- `void *(* realloc)(void *, int)`
- `int(* reset)(sqlite3_stmt *pStmt)`
- `void(* result_blob)(sqlite3_context *, const void *, int, void(*) (void *))`
- `void(* result_double)(sqlite3_context *, double)`
- `void(* result_error)(sqlite3_context *, const char *, int)`
- `void(* result_error16)(sqlite3_context *, const void *, int)`
- `void(* result_int)(sqlite3_context *, int)`
- `void(* result_int64)(sqlite3_context *, sqlite_int64)`
- `void(* result_null)(sqlite3_context *)`
- `void(* result_text)(sqlite3_context *, const char *, int, void(*) (void *))`
- `void(* result_text16)(sqlite3_context *, const void *, int, void(*) (void *))`
- `void(* result_text16be)(sqlite3_context *, const void *, int, void(*) (void *))`
- `void(* result_text16le)(sqlite3_context *, const void *, int, void(*) (void *))`
- `void(* result_value)(sqlite3_context *, sqlite3_value *)`
- `void *(* rollback_hook)(sqlite3 *, void(*) (void *), void *)`
- `int(* set_authorizer)(sqlite3 *, int(*) (void *, int, const char *, const char *, const char *, const char *), void *)`
- `void(* set_auxdata)(sqlite3_context *, int, void *, void(*) (void *))`
- `char *(* xsnprintf)(int, char *, const char *,...)`
- `int(* step)(sqlite3_stmt *)`
- `int(* table_column_metadata)(sqlite3 *, const char *, const char *, const char *, char const **, char const **, int *, int *, int *)`
- `void(* thread_cleanup)(void)`

- `int(* total_changes)(sqlite3 *)`
- `void (* trace)(sqlite3 *, void(*xTrace)(void *, const char *), void *)`
- `int(* transfer_bindings)(sqlite3_stmt *, sqlite3_stmt *)`
- `void (* update_hook)(sqlite3 *, void(*) (void *, int, char const *, char const *, sqlite_int64), void *)`
- `void (* user_data)(sqlite3_context *)`
- `const void (* value_blob)(sqlite3_value *)`
- `int(* value_bytes)(sqlite3_value *)`
- `int(* value_bytes16)(sqlite3_value *)`
- `double(* value_double)(sqlite3_value *)`
- `int(* value_int)(sqlite3_value *)`
- `sqlite_int64(* value_int64)(sqlite3_value *)`
- `int(* value_numeric_type)(sqlite3_value *)`
- `const unsigned char (* value_text)(sqlite3_value *)`
- `const void (* value_text16)(sqlite3_value *)`
- `const void (* value_text16be)(sqlite3_value *)`
- `const void (* value_text16le)(sqlite3_value *)`
- `int(* value_type)(sqlite3_value *)`
- `char (* vmprintf)(const char *, va_list)`
- `int(* overload_function)(sqlite3 *, const char *zFuncName, int nArg)`
- `int(* prepare_v2)(sqlite3 *, const char *, int, sqlite3_stmt **, const char **)`
- `int(* prepare16_v2)(sqlite3 *, const void *, int, sqlite3_stmt **, const void **)`
- `int(* clear_bindings)(sqlite3_stmt *)`
- `int(* create_module_v2)(sqlite3 *, const char *, const sqlite3_module *, void *, void(*xDestroy)(void *))`
- `int(* bind_zeroblob)(sqlite3_stmt *, int, int)`
- `int(* blob_bytes)(sqlite3_blob *)`
- `int(* blob_close)(sqlite3_blob *)`
- `int(* blob_open)(sqlite3 *, const char *, const char *, const char *, sqlite_int64, int, sqlite3_blob **)`
- `int(* blob_read)(sqlite3_blob *, void *, int, int)`
- `int(* blob_write)(sqlite3_blob *, const void *, int, int)`
- `int(* create_collation_v2)(sqlite3 *, const char *, int, void *, int(*) (void *, int, const void *, int, const void *), void(*) (void *))`
- `int(* file_control)(sqlite3 *, const char *, int, void *)`
- `sqlite_int64(* memory_highwater)(int)`
- `sqlite_int64(* memory_used)(void)`
- `sqlite3_mutex *(* mutex_alloc)(int)`
- `void(* mutex_enter)(sqlite3_mutex *)`
- `void(* mutex_free)(sqlite3_mutex *)`
- `void(* mutex_leave)(sqlite3_mutex *)`
- `int(* mutex_try)(sqlite3_mutex *)`
- `int(* open_v2)(const char *, sqlite3 **, int, const char *)`
- `int(* release_memory)(int)`
- `void(* result_error_nomem)(sqlite3_context *)`
- `void(* result_error_toobig)(sqlite3_context *)`
- `int(* sleep)(int)`
- `void(* soft_heap_limit)(int)`
- `sqlite3_vfs *(* vfs_find)(const char *)`
- `int(* vfs_register)(sqlite3_vfs *, int)`
- `int(* vfs_unregister)(sqlite3_vfs *)`
- `int(* xthreadsafe)(void)`
- `void(* result_zeroblob)(sqlite3_context *, int)`
- `void(* result_error_code)(sqlite3_context *, int)`
- `int(* test_control)(int,...)`
- `void(* randomness)(int, void *)`
- `sqlite3 *(* context_db_handle)(sqlite3_context *)`

- `int(* extended_result_codes)(sqlite3 *, int)`
- `int(* limit)(sqlite3 *, int, int)`
- `sqlite3_stmt *(* next_stmt)(sqlite3 *, sqlite3_stmt *)`
- `const char *(* sql)(sqlite3_stmt *)`
- `int(* status)(int, int *, int *, int)`
- `int(* backup_finish)(sqlite3_backup *)`
- `sqlite3_backup *(* backup_init)(sqlite3 *, const char *, sqlite3 *, const char *)`
- `int(* backup_pagecount)(sqlite3_backup *)`
- `int(* backup_remaining)(sqlite3_backup *)`
- `int(* backup_step)(sqlite3_backup *, int)`
- `const char *(* compileoption_get)(int)`
- `int(* compileoption_used)(const char *)`
- `int(* create_function_v2)(sqlite3 *, const char *, int, int, void *, void(*xFunc)(sqlite3_context *, int, sqlite3_value **), void(*xStep)(sqlite3_context *, int, sqlite3_value **), void(*xFinal)(sqlite3_context *, void(*xDestroy)(void *))`
- `int(* db_config)(sqlite3 *, int,...)`
- `sqlite3_mutex *(* db_mutex)(sqlite3 *)`
- `int(* db_status)(sqlite3 *, int, int *, int *, int)`
- `int(* extended_errcode)(sqlite3 *)`
- `void(* log)(int, const char *,...)`
- `sqlite3_int64(* soft_heap_limit64)(sqlite3_int64)`
- `const char *(* sourceid)(void)`
- `int(* stmt_status)(sqlite3_stmt *, int, int)`
- `int(* strnicmp)(const char *, const char *, int)`
- `int(* unlock_notify)(sqlite3 *, void(*) (void **, int), void *)`
- `int(* wal_autocheckpoint)(sqlite3 *, int)`
- `int(* wal_checkpoint)(sqlite3 *, const char *)`
- `void *(* wal_hook)(sqlite3 *, int(*) (void *, sqlite3 *, const char *, int), void *)`
- `int(* blob_reopen)(sqlite3_blob *, sqlite3_int64)`
- `int(* vtab_config)(sqlite3 *, int op,...)`
- `int(* vtab_on_conflict)(sqlite3 *)`
- `int(* close_v2)(sqlite3 *)`
- `const char *(* db_filename)(sqlite3 *, const char *)`
- `int(* db_readonly)(sqlite3 *, const char *)`
- `int(* db_release_memory)(sqlite3 *)`
- `const char *(* errstr)(int)`
- `int(* stmt_busy)(sqlite3_stmt *)`
- `int(* stmt_readonly)(sqlite3_stmt *)`
- `int(* stricmp)(const char *, const char *)`
- `int(* uri_boolean)(const char *, const char *, int)`
- `sqlite3_int64(* uri_int64)(const char *, const char *, sqlite3_int64)`
- `const char *(* uri_parameter)(const char *, const char *)`
- `char *(* xvsnprintf)(int, char *, const char *, va_list)`
- `int(* wal_checkpoint_v2)(sqlite3 *, const char *, int, int *, int *)`
- `int(* auto_extension)(void(*) (void))`
- `int(* bind_blob64)(sqlite3_stmt *, int, const void *, sqlite3_uint64, void(*) (void *))`
- `int(* bind_text64)(sqlite3_stmt *, int, const char *, sqlite3_uint64, void(*) (void *), unsigned char)`
- `int(* cancel_auto_extension)(void(*) (void))`
- `int(* load_extension)(sqlite3 *, const char *, const char *, char **)`
- `void *(* malloc64)(sqlite3_uint64)`
- `sqlite3_uint64(* msize)(void *)`
- `void *(* realloc64)(void *, sqlite3_uint64)`
- `void(* reset_auto_extension)(void)`
- `void(* result_blob64)(sqlite3_context *, const void *, sqlite3_uint64, void(*) (void *))`
- `void(* result_text64)(sqlite3_context *, const char *, sqlite3_uint64, void(*) (void *), unsigned char)`

- `int(* strglob)(const char *, const char *)`
- `sqlite3_value *(* value_dup)(const sqlite3_value *)`
- `void(* value_free)(sqlite3_value *)`
- `int(* result_zeroblob64)(sqlite3_context *, sqlite3_uint64)`
- `int(* bind_zeroblob64)(sqlite3_stmt *, int, sqlite3_uint64)`
- `unsigned int(* value_subtype)(sqlite3_value *)`
- `void(* result_subtype)(sqlite3_context *, unsigned int)`
- `int(* status64)(int, sqlite3_int64 *, sqlite3_int64 *, int)`
- `int(* striike)(const char *, const char *, unsigned int)`
- `int(* db_cacheflush)(sqlite3 *)`
- `int(* system_errno)(sqlite3 *)`
- `int(* trace_v2)(sqlite3 *, unsigned, int*)(unsigned, void *, void *, void *), void *)`
- `char *(* expanded_sql)(sqlite3_stmt *)`
- `void(* set_last_insert_rowid)(sqlite3 *, sqlite3_int64)`
- `int(* prepare_v3)(sqlite3 *, const char *, int, unsigned int, sqlite3_stmt **, const char **)`
- `int(* prepare16_v3)(sqlite3 *, const void *, int, unsigned int, sqlite3_stmt **, const void **)`
- `int(* bind_pointer)(sqlite3_stmt *, int, void *, const char *, void*)(void *)`
- `void(* result_pointer)(sqlite3_context *, void *, const char *, void*)(void *)`
- `void *(* value_pointer)(sqlite3_value *, const char *)`
- `int(* vtab_nochange)(sqlite3_context *)`
- `int(* value_nochange)(sqlite3_value *)`
- `const char *(* vtab_collation)(sqlite3_index_info *, int)`
- `int(* keyword_count)(void)`
- `int(* keyword_name)(int, const char **, int *)`
- `int(* keyword_check)(const char *, int)`
- `sqlite3_str *(* str_new)(sqlite3 *)`
- `char *(* str_finish)(sqlite3_str *)`
- `void(* str_appendf)(sqlite3_str *, const char *zFormat,...)`
- `void(* str_vappendf)(sqlite3_str *, const char *zFormat, va_list)`
- `void(* str_append)(sqlite3_str *, const char *zIn, int N)`
- `void(* str_appendall)(sqlite3_str *, const char *zIn)`
- `void(* str_appendchar)(sqlite3_str *, int N, char C)`
- `void(* str_reset)(sqlite3_str *)`
- `int(* str_errcode)(sqlite3_str *)`
- `int(* str_length)(sqlite3_str *)`
- `char *(* str_value)(sqlite3_str *)`
- `int(* create_window_function)(sqlite3 *, const char *, int, int, void *, void(*xStep)(sqlite3_context *, int, sqlite3_value **), void(*xFinal)(sqlite3_context *, void(*xValue)(sqlite3_context *, void(*xInv)(sqlite3_context *, int, sqlite3_value **), void(*xDestroy)(void *)`
- `const char *(* normalized_sql)(sqlite3_stmt *)`
- `int(* stmt_isexplain)(sqlite3_stmt *)`
- `int(* value_frombind)(sqlite3_value *)`
- `int(* drop_modules)(sqlite3 *, const char **)`
- `sqlite3_int64(* hard_heap_limit64)(sqlite3_int64)`
- `const char *(* uri_key)(const char *, int)`
- `const char *(* filename_database)(const char *)`
- `const char *(* filename_journal)(const char *)`
- `const char *(* filename_wal)(const char *)`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.113 sqlite3_backup Struct Reference

Public Attributes

- **sqlite3 * pDestDb**
- **Btree * pDest**
- **u32 iDestSchema**
- **int bDestLocked**
- **Pgno iNext**
- **sqlite3 * pSrcDb**
- **Btree * pSrc**
- **int rc**
- **Pgno nRemaining**
- **Pgno nPagecount**
- **int isAttached**
- **sqlite3_backup * pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.114 sqlite3_context Struct Reference

Public Attributes

- **Mem * pOut**
- **FuncDef * pFunc**
- **Mem * pMem**
- **Vdbe * pVdbe**
- **int iOp**
- **int isError**
- **u8 skipFlag**
- **u8 argc**
- **sqlite3_value * argv [1]**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.115 sqlite3_file Struct Reference

Public Attributes

- **const struct sqlite3_io_methods * pMethods**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.116 sqlite3_index_info::sqlite3_index_constraint Struct Reference

Public Attributes

- int **iColumn**
- unsigned char **op**
- unsigned char **usable**
- int **iTermOffset**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.117 sqlite3_index_info::sqlite3_index_constraint_usage Struct Reference

Public Attributes

- int **argvIndex**
- unsigned char **omit**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.118 sqlite3_index_info Struct Reference

Classes

- struct **sqlite3_index_constraint**
- struct **sqlite3_index_constraint_usage**
- struct **sqlite3_index_orderby**

Public Attributes

- int **nConstraint**
- struct **sqlite3_index_info::sqlite3_index_constraint * aConstraint**
- int **nOrderBy**
- struct **sqlite3_index_info::sqlite3_index_orderby * aOrderBy**
- struct **sqlite3_index_info::sqlite3_index_constraint_usage * aConstraintUsage**
- int **idxNum**
- char * **idxStr**
- int **needToFreeIdxStr**
- int **orderByConsumed**
- double **estimatedCost**
- sqlite3_int64 **estimatedRows**
- int **idxFlags**
- sqlite3_uint64 **colUsed**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.119 sqlite3_index_info::sqlite3_index_orderby Struct Reference

Public Attributes

- int **iColumn**
- unsigned char **desc**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.120 sqlite3_io_methods Struct Reference

Public Attributes

- int **iVersion**
- int(* **xClose**)(**sqlite3_file** *)
- int(* **xRead**)(**sqlite3_file** *, void *, int iAmt, sqlite3_int64 iOfst)
- int(* **xWrite**)(**sqlite3_file** *, const void *, int iAmt, sqlite3_int64 iOfst)
- int(* **xTruncate**)(**sqlite3_file** *, sqlite3_int64 size)
- int(* **xSync**)(**sqlite3_file** *, int flags)
- int(* **xFileSize**)(**sqlite3_file** *, sqlite3_int64 *pSize)
- int(* **xLock**)(**sqlite3_file** *, int)
- int(* **xUnlock**)(**sqlite3_file** *, int)
- int(* **xCheckReservedLock**)(**sqlite3_file** *, int *pResOut)
- int(* **xFileControl**)(**sqlite3_file** *, int op, void *pArg)
- int(* **xSectorSize**)(**sqlite3_file** *)
- int(* **xDeviceCharacteristics**)(**sqlite3_file** *)
- int(* **xShmMap**)(**sqlite3_file** *, int iPg, int pgsz, int, void volatile **)
- int(* **xShmLock**)(**sqlite3_file** *, int offset, int n, int flags)
- void(* **xShmBarrier**)(**sqlite3_file** *)
- int(* **xShmUnmap**)(**sqlite3_file** *, int deleteFlag)
- int(* **xFetch**)(**sqlite3_file** *, sqlite3_int64 iOfst, int iAmt, void **pp)
- int(* **xUnfetch**)(**sqlite3_file** *, sqlite3_int64 iOfst, void *p)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.121 sqlite3_mem_methods Struct Reference

Public Attributes

- void *****(*** xMalloc**)(int)
- void(*** xFree**)(void *****)
- void *****(*** xRealloc**)(void *****, int)
- int(*** xSize**)(void *****)
- int(*** xRoundup**)(int)
- int(*** xInit**)(void *****)
- void(*** xShutdown**)(void *****)
- void ***** **pAppData**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.122 sqlite3_module Struct Reference

Public Attributes

- int **iVersion**
- int(*** xCreate**)(**sqlite3** *****, void *****pAux, int argc, const char *****const *****argv, **sqlite3_vtab** ****ppVTab**, char ******)
- int(*** xConnect**)(**sqlite3** *****, void *****pAux, int argc, const char *****const *****argv, **sqlite3_vtab** ****ppVTab**, char ******)
- int(*** xBestIndex**)(**sqlite3_vtab** *****pVTab, **sqlite3_index_info** *****)
- int(*** xDisconnect**)(**sqlite3_vtab** *****pVTab)
- int(*** xDestroy**)(**sqlite3_vtab** *****pVTab)
- int(*** xOpen**)(**sqlite3_vtab** *****pVTab, **sqlite3_vtab_cursor** ****ppCursor**)
- int(*** xClose**)(**sqlite3_vtab_cursor** *****)
- int(*** xFilter**)(**sqlite3_vtab_cursor** *****, int idxNum, const char *****idxStr, int argc, **sqlite3_value** ****argv**)
- int(*** xNext**)(**sqlite3_vtab_cursor** *****)
- int(*** xEOF**)(**sqlite3_vtab_cursor** *****)
- int(*** xColumn**)(**sqlite3_vtab_cursor** *****, **sqlite3_context** *****, int)
- int(*** xRowid**)(**sqlite3_vtab_cursor** *****, **sqlite3_int64** *****pRowid)
- int(*** xUpdate**)(**sqlite3_vtab** *****, int, **sqlite3_value** ******, **sqlite3_int64** *****)
- int(*** xBegin**)(**sqlite3_vtab** *****pVTab)
- int(*** xSync**)(**sqlite3_vtab** *****pVTab)
- int(*** xCommit**)(**sqlite3_vtab** *****pVTab)
- int(*** xRollback**)(**sqlite3_vtab** *****pVTab)
- int(*** xFindFunction**)(**sqlite3_vtab** *****pVTab, int nArg, const char *****zName, void(****pFunc**)(**sqlite3_context** *****, int, **sqlite3_value** ******), void ****ppArg**)
- int(*** xRename**)(**sqlite3_vtab** *****pVTab, const char *****zNew)
- int(*** xSavepoint**)(**sqlite3_vtab** *****pVTab, int)
- int(*** xRelease**)(**sqlite3_vtab** *****pVTab, int)
- int(*** xRollbackTo**)(**sqlite3_vtab** *****pVTab, int)
- int(*** xShadowName**)(const char *****)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.123 sqlite3_mutex Struct Reference

Public Attributes

- pthread_mutex_t **mutex**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.124 sqlite3_mutex_methods Struct Reference

Public Attributes

- int(* **xMutexInit**)(void)
- int(* **xMutexEnd**)(void)
- sqlite3_mutex *(* **xMutexAlloc**)(int)
- void(* **xMutexFree**)(sqlite3_mutex *)
- void(* **xMutexEnter**)(sqlite3_mutex *)
- int(* **xMutexTry**)(sqlite3_mutex *)
- void(* **xMutexLeave**)(sqlite3_mutex *)
- int(* **xMutexHeld**)(sqlite3_mutex *)
- int(* **xMutexNotheld**)(sqlite3_mutex *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.125 sqlite3_pcache_methods Struct Reference

Public Attributes

- void * **pArg**
- int(* **xInit**)(void *)
- void(* **xShutdown**)(void *)
- sqlite3_pcache *(* **xCreate**)(int szPage, int bPurgeable)
- void(* **xCachesize**)(sqlite3_pcache *, int nCachesize)
- int(* **xPagecount**)(sqlite3_pcache *)
- void(* **xFetch**)(sqlite3_pcache *, unsigned key, int createFlag)
- void(* **xUnpin**)(sqlite3_pcache *, void *, int discard)
- void(* **xRekey**)(sqlite3_pcache *, void *, unsigned oldKey, unsigned newKey)
- void(* **xTruncate**)(sqlite3_pcache *, unsigned iLimit)
- void(* **xDestroy**)(sqlite3_pcache *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.126 sqlite3_pcache_methods2 Struct Reference

Public Attributes

- int **iVersion**
- void * **pArg**
- int(* **xInit**)(void *)
- void(* **xShutdown**)(void *)
- sqlite3_pcache *(* **xCreate**)(int szPage, int szExtra, int bPurgeable)
- void(* **xCachesize**)(sqlite3_pcache *, int nCachesize)
- int(* **xPagecount**)(sqlite3_pcache *)
- **sqlite3_pcache_page** *(* **xFetch**)(sqlite3_pcache *, unsigned key, int createFlag)
- void(* **xUnpin**)(sqlite3_pcache *, **sqlite3_pcache_page** *, int discard)
- void(* **xRekey**)(sqlite3_pcache *, **sqlite3_pcache_page** *, unsigned oldKey, unsigned newKey)
- void(* **xTruncate**)(sqlite3_pcache *, unsigned iLimit)
- void(* **xDestroy**)(sqlite3_pcache *)
- void(* **xShrink**)(sqlite3_pcache *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.127 sqlite3_pcache_page Struct Reference

Public Attributes

- void * **pBuf**
- void * **pExtra**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.128 sqlite3_rtree_geometry Struct Reference

Public Attributes

- void * **pContext**
- int **nParam**
- sqlite3_rtree_dbl * **aParam**
- void * **pUser**
- void(* **xDelUser**)(void *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.129 sqlite3_rtree_query_info Struct Reference

Public Attributes

- void * **pContext**
- int **nParam**
- sqlite3_rtree_dbl * **aParam**
- void * **pUser**
- void(* **xDelUser**)(void *)
- sqlite3_rtree_dbl * **aCoord**
- unsigned int * **anQueue**
- int **nCoord**
- int **iLevel**
- int **mxLevel**
- sqlite3_int64 **iRowid**
- sqlite3_rtree_dbl **rParentScore**
- int **eParentWithin**
- int **eWithin**
- sqlite3_rtree_dbl **rScore**
- **sqlite3_value** ** **apSqlParam**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.130 sqlite3_snapshot Struct Reference

Public Attributes

- unsigned char **hidden** [48]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.131 sqlite3_str Struct Reference

Public Attributes

- **sqlite3** * **db**
- char * **zText**
- u32 **nAlloc**
- u32 **mxAlloc**
- u32 **nChar**
- u8 **accError**
- u8 **printfFlags**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.132 sqlite3_value Struct Reference

Classes

- union **MemValue**

Public Attributes

- union **sqlite3_value::MemValue** u
- u16 **flags**
- u8 **enc**
- u8 **eSubtype**
- int **n**
- char * **z**
- char * **zMalloc**
- int **szMalloc**
- u32 **uTemp**
- **sqlite3** * **db**
- void(* **xDel**)(void *)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.133 sqlite3_vfs Struct Reference

Public Attributes

- int **iVersion**
- int **szOsFile**
- int **mxPathname**
- **sqlite3_vfs** * **pNext**
- const char * **zName**
- void * **pAppData**
- int(* **xOpen**)(**sqlite3_vfs** *, const char *zName, **sqlite3_file** *, int flags, int *pOutFlags)
- int(* **xDelete**)(**sqlite3_vfs** *, const char *zName, int syncDir)
- int(* **xAccess**)(**sqlite3_vfs** *, const char *zName, int flags, int *pResOut)
- int(* **xFullPathname**)(**sqlite3_vfs** *, const char *zName, int nOut, char *zOut)
- void(* **xDIOpen**)(**sqlite3_vfs** *, const char *zFilename)
- void(* **xDLError**)(**sqlite3_vfs** *, int nByte, char *zErrMsg)
- void(* **xDISym**)(**sqlite3_vfs** *, void *, const char *zSymbol)(void)
- void(* **xDIClose**)(**sqlite3_vfs** *, void *)
- int(* **xRandomness**)(**sqlite3_vfs** *, int nByte, char *zOut)
- int(* **xSleep**)(**sqlite3_vfs** *, int microseconds)
- int(* **xCurrentTime**)(**sqlite3_vfs** *, double *)
- int(* **xGetLastError**)(**sqlite3_vfs** *, int, char *)
- int(* **xCurrentTimeInt64**)(**sqlite3_vfs** *, sqlite3_int64 *)
- int(* **xSetSystemCall**)(**sqlite3_vfs** *, const char *zName, sqlite3_syscall_ptr)
- sqlite3_syscall_ptr(* **xGetSystemCall**)(**sqlite3_vfs** *, const char *zName)
- const char *(* **xNextSystemCall**)(**sqlite3_vfs** *, const char *zName)

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.134 sqlite3_vtab Struct Reference

Public Attributes

- const **sqlite3_module** * **pModule**
- int **nRef**
- char * **zErrMsg**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.135 sqlite3_vtab_cursor Struct Reference

Public Attributes

- **sqlite3_vtab** * **pVtab**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.h

3.136 Sqlite3Config Struct Reference

Public Attributes

- int **bMemstat**
- u8 **bCoreMutex**
- u8 **bFullMutex**
- u8 **bOpenUri**
- u8 **bUseCis**
- u8 **bSmallMalloc**
- u8 **bExtraSchemaChecks**
- int **mxStrlen**
- int **neverCorrupt**
- int **szLookaside**
- int **nLookaside**
- int **nStmtSpill**
- **sqlite3_mem_methods** **m**
- **sqlite3_mutex_methods** **mutex**
- **sqlite3_pcache_methods2** **pcache2**
- void * **pHeap**
- int **nHeap**
- int **mnReq**
- int **mxReq**
- **sqlite3_int64** **szMmap**

- `sqlite3_int64 mxMmap`
- `void * pPage`
- `int szPage`
- `int nPage`
- `int mxParserStack`
- `int sharedCacheEnabled`
- `u32 szPma`
- `int isInit`
- `int inProgress`
- `int isMutexInit`
- `int isMallocInit`
- `int isPCacheInit`
- `int nRefInitMutex`
- `sqlite3_mutex * plnitMutex`
- `void(* xLog)(void *, int, const char *)`
- `void * pLogArg`
- `int(* xTestCallback)(int)`
- `int bLocaltimeFault`
- `int iOnceResetThreshold`
- `u32 szSorterRef`
- `unsigned int iPngSeed`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.137 `sqlite3::sqlite3InitInfo` Struct Reference

Public Attributes

- `int newTnum`
- `u8 iDb`
- `u8 busy`
- `unsigned orphanTrigger: 1`
- `unsigned imposterTable: 1`
- `unsigned reopenMemdb: 1`
- `char ** azInit`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.138 `SQLiteThread` Struct Reference

Public Attributes

- `pthread_t tid`
- `int done`
- `void * pOut`
- `void(* xTask)(void *)`
- `void * pln`

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

- `CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c`

3.139 SrcCount Struct Reference

Public Attributes

- **SrcList * pSrc**
- **int nThis**
- **int nOther**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.140 SrcList Struct Reference

Classes

- struct **SrcList_item**

Public Attributes

- **int nSrc**
- **u32 nAlloc**
- **struct SrcList::SrcList_item a [1]**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.141 SrcList::SrcList_item Struct Reference

Public Attributes

- **Schema * pSchema**
- **char * zDatabase**
- **char * zName**
- **char * zAlias**
- **Table * pTab**
- **Select * pSelect**
- **int addrFillSub**
- **int regReturn**
- **int regResult**
-

```

struct {
    u8 jointype
    unsigned notIndexed:1
    unsigned isIndexedBy:1
    unsigned isTabFunc:1
    unsigned isCorrelated:1
    unsigned viaCoroutine:1
    unsigned isRecursive:1
    unsigned fromDDL:1
} fg

• int iCursor
• Expr * pOn
• IdList * pUsing
• Bitmask colUsed
•

union {
    char * zIndexedBy
    ExprList * pFuncArg
} u1

• Index * pIBIndex

```

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.142 Stat4Accum Struct Reference

Public Attributes

- tRowcnt **nRow**
- tRowcnt **nPSample**
- int **nCol**
- int **nKeyCol**
- int **mxSample**
- **Stat4Sample current**
- u32 **iPrn**
- **Stat4Sample * aBest**
- int **iMin**
- int **nSample**
- int **nMaxEqZero**
- int **iGet**
- **Stat4Sample * a**
- **sqlite3 * db**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.143 Stat4Sample Struct Reference

Public Attributes

- tRowcnt * **anEq**
- tRowcnt * **anDLt**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.144 SubProgram Struct Reference

Public Attributes

- **VdbeOp** * **aOp**
- int **nOp**
- int **nMem**
- int **nCsr**
- u8 * **aOnce**
- void * **token**
- **SubProgram** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.145 SubstContext Struct Reference

Public Attributes

- **Parse** * **pParse**
- int **iTable**
- int **iNewTable**
- int **isLeftJoin**
- **ExprList** * **pEList**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.146 SumCtx Struct Reference

Public Attributes

- double **rSum**
- i64 **iSum**
- i64 **cnt**
- u8 **overflow**
- u8 **approx**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.147 Table Struct Reference

Public Attributes

- char * **zName**
- Column * **aCol**
- Index * **pIndex**
- Select * **pSelect**
- FKey * **pFKey**
- char * **zColAff**
- ExprList * **pCheck**
- int **tnum**
- u32 **nTabRef**
- u32 **tabFlags**
- i16 **iPKey**
- i16 **nCol**
- i16 **nNVCol**
- LogEst **nRowLogEst**
- LogEst **szTabRow**
- u8 **keyConf**
- int **addColOffset**
- int **nModuleArg**
- char ** **azModuleArg**
- VTable * **pVTable**
- Trigger * **pTrigger**
- Schema * **pSchema**
- Table * **pNextZombie**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.148 TableLock Struct Reference

Public Attributes

- int **iDb**
- int **iTab**
- u8 **isWriteLock**
- const char * **zLockName**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.149 TabResult Struct Reference

Public Attributes

- char ** **azResult**
- char * **zErrMsg**
- u32 **nAlloc**
- u32 **nRow**
- u32 **nColumn**
- u32 **nData**
- int **rc**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.150 Token Struct Reference

Public Attributes

- const char * **z**
- unsigned int **n**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.151 TrigEvent Struct Reference

Public Attributes

- int **a**
- **IdList** * **b**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.152 Trigger Struct Reference

Public Attributes

- char * **zName**
- char * **table**
- u8 **op**
- u8 **tr_tm**
- **Expr** * **pWhen**
- **IdList** * **pColumns**
- **Schema** * **pSchema**
- **Schema** * **pTabSchema**
- **TriggerStep** * **step_list**
- **Trigger** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.153 TriggerPrg Struct Reference

Public Attributes

- **Trigger** * **pTrigger**
- **TriggerPrg** * **pNext**
- **SubProgram** * **pProgram**
- int **orconf**
- u32 **aColmask** [2]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.154 TriggerStep Struct Reference

Public Attributes

- **u8 op**
- **u8 orconf**
- **Trigger * pTrig**
- **Select * pSelect**
- **char * zTarget**
- **Expr * pWhere**
- **ExprList * pExprList**
- **IdList * pldList**
- **Upsert * pUpsert**
- **char * zSpan**
- **TriggerStep * pNext**
- **TriggerStep * pLast**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.155 unixFile Struct Reference

Public Attributes

- **sqlite3_io_methods const * pMethod**
- **sqlite3_vfs * pVfs**
- **unixInodeInfo * pNode**
- **int h**
- **unsigned char eFileLock**
- **unsigned short int ctrlFlags**
- **int lastErrno**
- **void * lockingContext**
- **UnixUnusedFd * pPreallocatedUnused**
- **const char * zPath**
- **unixShm * pShm**
- **int szChunk**
- **int sectorSize**
- **int deviceCharacteristics**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.156 unixFileId Struct Reference

Public Attributes

- dev_t **dev**
- u64 **ino**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.157 unixNodeInfo Struct Reference

Public Attributes

- struct **unixFileId** **fileId**
- **sqlite3_mutex** * **pLockMutex**
- int **nShared**
- int **nLock**
- unsigned char **eFileLock**
- unsigned char **bProcessLock**
- **UnixUnusedFd** * **pUnused**
- int **nRef**
- **unixShmNode** * **pShmNode**
- **unixNodeInfo** * **pNext**
- **unixNodeInfo** * **pPrev**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.158 unixShm Struct Reference

Public Attributes

- **unixShmNode** * **pShmNode**
- **unixShm** * **pNext**
- u8 **hasMutex**
- u8 **id**
- u16 **sharedMask**
- u16 **exclMask**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.159 unixShmNode Struct Reference

Public Attributes

- **unixNodeInfo** * **pNode**
- **sqlite3_mutex** * **pShmMutex**
- char * **zFilename**
- int **hShm**
- int **szRegion**
- u16 **nRegion**
- u8 **isReadonly**
- u8 **isUnlocked**
- char ** **apRegion**
- int **nRef**
- **unixShm** * **pFirst**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.160 UnixUnusedFd Struct Reference

Public Attributes

- int **fd**
- int **flags**
- **UnixUnusedFd** * **pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.161 UnpackedRecord Struct Reference

Public Attributes

- **KeyInfo** * **pKeyInfo**
- **Mem** * **aMem**
- u16 **nField**
- i8 **default_rc**
- u8 **errCode**
- i8 **r1**
- i8 **r2**
- u8 **eqSeen**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.162 Upsert Struct Reference

Public Attributes

- **ExprList * pUpsertTarget**
- **Expr * pUpsertTargetWhere**
- **ExprList * pUpsertSet**
- **Expr * pUpsertWhere**
- **Index * pUpsertIdx**
- **SrcList * pUpsertSrc**
- **int regData**
- **int iDataCur**
- **int ildxCur**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.163 ValueNewStat4Ctx Struct Reference

Public Attributes

- **Parse * pParse**
- **Index * pIdx**
- **UnpackedRecord ** ppRec**
- **int iVal**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.164 Vdbe Struct Reference

Public Attributes

- **sqlite3 * db**
- **Vdbe * pPrev**
- **Vdbe * pNext**
- **Parse * pParse**
- **ynVar nVar**
- **u32 magic**
- **int nMem**
- **int nCursor**
- **u32 cacheCtr**
- **int pc**
- **int rc**
- **int nChange**
- **int iStatement**

- i64 **iCurrentTime**
- i64 **nFkConstraint**
- i64 **nStmtDefCons**
- i64 **nStmtDeflmmCons**
- **Mem * aMem**
- **Mem ** apArg**
- **VdbeCursor ** apCsr**
- **Mem * aVar**
- **Op * aOp**
- int **nOp**
- int **nOpAlloc**
- **Mem * aColName**
- **Mem * pResultSet**
- char * **zErrMsg**
- VList * **pVList**
- i64 **startTime**
- u16 **nResColumn**
- u8 **errorAction**
- u8 **minWriteFileFormat**
- u8 **prepFlags**
- bft **expired:2**
- bft **explain:2**
- bft **doingRerun:1**
- bft **changeCntOn:1**
- bft **runOnlyOnce:1**
- bft **usesStmtJournal:1**
- bft **readOnly:1**
- bft **blsReader:1**
- yDbMask **btreeMask**
- yDbMask **lockMask**
- u32 **aCounter** [7]
- char * **zSql**
- void * **pFree**
- **VdbeFrame * pFrame**
- **VdbeFrame * pDelFrame**
- int **nFrame**
- u32 **expmask**
- **SubProgram * pProgram**
- **AuxData * pAuxData**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.165 VdbeCursor Struct Reference

Public Attributes

- **u8 eCurType**
- **i8 iDb**
- **u8 nullRow**
- **u8 deferredMoveto**
- **u8 isTable**
- **Bool isEphemeral:1**
- **Bool useRandomRowid:1**
- **Bool isOrdered:1**
- **Bool seekHit:1**
- **Btree * pBtx**
- **i64 seqCount**
- **int * aAltMap**
- **u32 cacheStatus**
- **int seekResult**
- **VdbeCursor * pAltCursor**
-
- union {
 - BtCursor * pCursor**
 - sqlite3_vtab_cursor * pVCur**
 - VdbeSorter * pSorter**
- } **uc**
-
- **KeyInfo * pKeyInfo**
- **u32 iHdrOffset**
- **Pgno pgnoRoot**
- **i16 nField**
- **u16 nHdrParsed**
- **i64 movetoTarget**
- **u32 * aOffset**
- **const u8 * aRow**
- **u32 payloadSize**
- **u32 szRow**
- **u32 aType [1]**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.166 VdbeFrame Struct Reference

Public Attributes

- **Vdbe * v**
- **VdbeFrame * pParent**
- **Op * aOp**
- **i64 * anExec**
- **Mem * aMem**

- **VdbeCursor** ** **apCsr**
- u8 * **aOnce**
- void * **token**
- i64 **lastRowid**
- **AuxData** * **pAuxData**
- int **nCursor**
- int **pc**
- int **nOp**
- int **nMem**
- int **nChildMem**
- int **nChildCsr**
- int **nChange**
- int **nDbChange**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.167 VdbeOp Struct Reference

Classes

- union **p4union**

Public Attributes

- u8 **opcode**
- signed char **p4type**
- u16 **p5**
- int **p1**
- int **p2**
- int **p3**
- union **VdbeOp::p4union** **p4**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.168 VdbeOpList Struct Reference

Public Attributes

- u8 **opcode**
- signed char **p1**
- signed char **p2**
- signed char **p3**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.169 VdbeSorter Struct Reference

Public Attributes

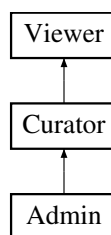
- int **mnPmaSize**
- int **mxPmaSize**
- int **mxKeysize**
- int **pgsz**
- **PmaReader** * **pReader**
- **MergeEngine** * **pMerger**
- **sqlite3** * **db**
- **KeyInfo** * **pKeyInfo**
- **UnpackedRecord** * **pUnpacked**
- **SorterList** **list**
- int **iMemory**
- int **nMemory**
- u8 **bUsePMA**
- u8 **bUseThreads**
- u8 **iPrev**
- u8 **nTask**
- u8 **typeMask**
- **SortSubtask** **aTask** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.170 Viewer Class Reference

Inheritance diagram for Viewer:



Public Member Functions

- **Viewer** ()
***Viewer::Viewer** (p. 100) Default Constructor.*
- **Viewer** (int _userID, string _un, string _pw, string _priv, string _desc, string _selfPic)
***Viewer::Viewer** (p. 100) Constructor that initializes class instance variables with parameters.*
- int **getUserID** ()
- void **setUserID** (int)
- string **getUn** ()
- void **setUn** (string)

- string **getPw** ()
- void **setPw** (string)
- string **getPrivelege** ()
- void **setPrivelege** (string)
- string **getDescription** ()
- void **setDescription** (string)
- string **getSelfPic** ()
- void **setSelfPic** (string)
- string **getUniquePW** ()
- void **setUniquePW** (string)

Protected Attributes

- string **pw**
- string **un**
- string **privelege**
- int **userID**
- string **description**
- string **selfPic**
- string **uniquePW**

3.170.1 Constructor & Destructor Documentation

3.170.1.1 Viewer()

```
Viewer::Viewer (
    int _userID,
    string _un,
    string _pw,
    string _priv,
    string _desc,
    string _selfPic )
```

Viewer::Viewer (p. 100) Constructor that initializes class instance variables with parameters.

Parameters

_userID	The Viewer (p. 100)'s ID
_un	The Viewer (p. 100)'s Username
_pw	The Viewer (p. 100)'s Password
_priv	The Viewer (p. 100)'s Privelege level
_desc	The Viewer (p. 100)'s self description
_selfPic	The Viewer (p. 100)'s Profile picture

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/viewer.h
- CS_205_Software_Engineering/Group Project/project/museum/museum/project/viewer.cpp

3.171 VtabCtx Struct Reference

Public Attributes

- **VTable * pVTable**
- **Table * pTab**
- **VtabCtx * pPrior**
- **int bDeclared**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.172 VTable Struct Reference

Public Attributes

- **sqlite3 * db**
- **Module * pMod**
- **sqlite3_vtab * pVtab**
- **int nRef**
- **u8 bConstraint**
- **u8 eVtabRisk**
- **int iSavepoint**
- **VTable * pNext**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.173 vxworksFileId Struct Reference

Public Attributes

- **struct vxworksFileId * pNext**
- **int nRef**
- **int nName**
- **char * zCanonicalName**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.174 Wal Struct Reference

Public Attributes

- **sqlite3_vfs** * **pVfs**
- **sqlite3_file** * **pDbFd**
- **sqlite3_file** * **pWalFd**
- **u32** **iCallback**
- **i64** **mxWalSize**
- **int** **nWiData**
- **int** **szFirstBlock**
- **volatile u32** ** **apWiData**
- **u32** **szPage**
- **i16** **readLock**
- **u8** **syncFlags**
- **u8** **exclusiveMode**
- **u8** **writeLock**
- **u8** **ckptLock**
- **u8** **readOnly**
- **u8** **truncateOnCommit**
- **u8** **syncHeader**
- **u8** **padToSectorBoundary**
- **u8** **bShmUnreliable**
- **WalIndexHdr** **hdr**
- **u32** **minFrame**
- **u32** **iReCksum**
- **const char** * **zWalName**
- **u32** **nCkpt**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.175 WalCkptInfo Struct Reference

Public Attributes

- **u32** **nBackfill**
- **u32** **aReadMark** [WAL_NREADER]
- **u8** **aLock** [SQLITE_SHM_NLOCK]
- **u32** **nBackfillAttempted**
- **u32** **notUsed0**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.176 WalHashLoc Struct Reference

Public Attributes

- volatile ht_slot * **aHash**
- volatile u32 * **aPgno**
- u32 **iZero**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.177 WalIndexHdr Struct Reference

Public Attributes

- u32 **iVersion**
- u32 **unused**
- u32 **iChange**
- u8 **isInit**
- u8 **bigEndCksum**
- u16 **szPage**
- u32 **mxFrame**
- u32 **nPage**
- u32 **aFrameCksum** [2]
- u32 **aSalt** [2]
- u32 **aCksum** [2]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.178 WalIterator Struct Reference

Classes

- struct **WalSegment**

Public Attributes

- int **iPrior**
- int **nSegment**
- struct **WalIterator::WalSegment** **aSegment** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.179 Walker Struct Reference

Public Attributes

- **Parse** * **pParse**
- int(* **xExprCallback**)(**Walker** *, **Expr** *)
- int(* **xSelectCallback**)(**Walker** *, **Select** *)
- void(* **xSelectCallback2**)(**Walker** *, **Select** *)
- int **walkerDepth**
- u16 **eCode**
-

```

union {
    NameContext * pNC
    int n
    int iCur
    SrcList * pSrcList
    struct SrcCount * pSrcCount
    struct CCurHint * pCCurHint
    int * aiCol
    struct IdxCover * pldxCover
    struct IdxExprTrans * pldxTrans
    ExprList * pGroupBy
    Select * pSelect
    struct WindowRewrite * pRewrite
    struct WhereConst * pConst
    struct RenameCtx * pRename
    struct Table * pTab
} u

```

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.180 WalIterator::WalSegment Struct Reference

Public Attributes

- int **iNext**
- ht_slot * **aIndex**
- u32 * **aPgno**
- int **nEntry**
- int **iZero**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.181 WalWriter Struct Reference

Public Attributes

- **Wal * pWal**
- **sqlite3_file * pFd**
- **sqlite3_int64 iSyncPoint**
- **int syncFlags**
- **int szPage**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.182 WhereAndInfo Struct Reference

Public Attributes

- **WhereClause wc**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.183 WhereClause Struct Reference

Public Attributes

- **WhereInfo * pWInfo**
- **WhereClause * pOuter**
- **u8 op**
- **u8 hasOr**
- **int nTerm**
- **int nSlot**
- **WhereTerm * a**
- **WhereTerm aStatic [8]**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.184 WhereConst Struct Reference

Public Attributes

- **Parse** * **pParse**
- int **nConst**
- int **nChng**
- **Expr** ** **apExpr**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.185 WhereExprMod Struct Reference

Public Attributes

- **WhereExprMod** * **pNext**
- **Expr** * **pExpr**
- **Expr** **orig**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.186 WhereInfo Struct Reference

Public Attributes

- **Parse** * **pParse**
- **SrcList** * **pTabList**
- **ExprList** * **pOrderBy**
- **ExprList** * **pResultSet**
- **Expr** * **pWhere**
- int **aiCurOnePass** [2]
- int **iContinue**
- int **iBreak**
- int **savedNQueryLoop**
- u16 **wctrlFlags**
- LogEst **iLimit**
- u8 **nLevel**
- i8 **nOBSat**
- u8 **eOnePass**
- u8 **eDistinct**
- unsigned **bDeferredSeek**:1
- unsigned **untestedTerms**:1
- unsigned **bOrderedInnerLoop**:1
- unsigned **sorted**:1

- LogEst **nRowOut**
- int **iTop**
- **WhereLoop** * **pLoops**
- **WhereExprMod** * **pExprMods**
- Bitmask **revMask**
- **WhereClause** **sWC**
- **WhereMaskSet** **sMaskSet**
- **WhereLevel** **a** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.187 WhereLevel Struct Reference

Public Attributes

- int **iLeftJoin**
- int **iTabCur**
- int **ildxCur**
- int **addrBrk**
- int **addrNxt**
- int **addrSkip**
- int **addrCont**
- int **addrFirst**
- int **addrBody**
- int **regBignull**
- int **addrBignull**
- u32 **iLikeRepCntr**
- int **addrLikeRep**
- u8 **iFrom**
- u8 **op**
- u8 **p3**
- u8 **p5**
- int **p1**
- int **p2**
-
- union {
 struct {
 int **nIn**
 struct **InLoop** {
 int **iCur**
 int **addrInTop**
 int **iBase**
 int **nPrefix**
 u8 **eEndLoopOp**
 } **alnLoop**
 } **in**
Index * **pCovidx**
 } **u**
- struct **WhereLoop** * **pWLoop**
- Bitmask **notReady**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.188 WhereLoop Struct Reference

Public Attributes

- Bitmask **prereq**
- Bitmask **maskSelf**
- u8 **iTab**
- u8 **iSortIdx**
- LogEst **rSetup**
- LogEst **rRun**
- LogEst **nOut**
-
- union {
 struct {
 u16 **nEq**
 u16 **nBtm**
 u16 **nTop**
 u16 **nDistinctCol**
 Index * **pIndex**
 } **btree**
 struct {
 int **idxNum**
 u8 **needFree**
 i8 **isOrdered**
 u16 **omitMask**
 char * **idxStr**
 } **vtab**
 } **u**
- u32 **wsFlags**
- u16 **nLTerm**
- u16 **nSkip**
- u16 **nLSlot**
- WhereTerm ** **aLTerm**
- WhereLoop * **pNextLoop**
- WhereTerm * **aLTermSpace** [3]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.189 WhereLoopBuilder Struct Reference

Public Attributes

- WhereInfo * **pWInfo**
- WhereClause * **pWC**
- ExprList * **pOrderBy**
- WhereLoop * **pNew**
- WhereOrSet * **pOrSet**
- unsigned int **bldFlags**
- unsigned int **iPlanLimit**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.190 WhereMaskSet Struct Reference

Public Attributes

- int **bVarSelect**
- int **n**
- int **ix** [BMS]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.191 WhereOrCost Struct Reference

Public Attributes

- Bitmask **prereq**
- LogEst **rRun**
- LogEst **nOut**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.192 WhereOrInfo Struct Reference

Public Attributes

- **WhereClause wc**
- Bitmask **indexable**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.193 WhereOrSet Struct Reference

Public Attributes

- u16 **n**
- **WhereOrCost a** [N_OR_COST]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.194 WherePath Struct Reference

Public Attributes

- Bitmask **maskLoop**
- Bitmask **revLoop**
- LogEst **nRow**
- LogEst **rCost**
- LogEst **rUnsorted**
- i8 **isOrdered**
- **WhereLoop** ** **aLoop**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.195 WhereScan Struct Reference

Public Attributes

- **WhereClause** * **pOrigWC**
- **WhereClause** * **pWC**
- const char * **zCollName**
- **Expr** * **pIdxExpr**
- char **idxaff**
- unsigned char **nEquiv**
- unsigned char **iEquiv**
- u32 **opMask**
- int **k**
- int **aiCur** [11]
- i16 **aiColumn** [11]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.196 WhereTerm Struct Reference

Public Attributes

- **Expr** * **pExpr**
- **WhereClause** * **pWC**
- LogEst **truthProb**
- u16 **wtFlags**
- u16 **eOperator**
- u8 **nChild**
- u8 **eMatchOp**
- int **iParent**

- int **leftCursor**
- int **iField**
-
- union {
 - int **leftColumn**
 - WhereOrInfo** * **pOrInfo**
 - WhereAndInfo** * **pAndInfo**
- } **u**
- Bitmask **prereqRight**
- Bitmask **prereqAll**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.197 Window Struct Reference

Public Attributes

- char * **zName**
- char * **zBase**
- **ExprList** * **pPartition**
- **ExprList** * **pOrderBy**
- u8 **eFrmType**
- u8 **eStart**
- u8 **eEnd**
- u8 **blmplicitFrame**
- u8 **eExclude**
- **Expr** * **pStart**
- **Expr** * **pEnd**
- **Window** ** **ppThis**
- **Window** * **pNextWin**
- **Expr** * **pFilter**
- **FuncDef** * **pFunc**
- int **iEphCsr**
- int **regAccum**
- int **regResult**
- int **csrApp**
- int **regApp**
- int **regPart**
- **Expr** * **pOwner**
- int **nBufferCol**
- int **iArgCol**
- int **regOne**
- int **regStartRowid**
- int **regEndRowid**
- u8 **bExprArgs**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.198 WindowCodeArg Struct Reference

Public Attributes

- **Parse** * **pParse**
- **Window** * **pMWin**
- **Vdbe** * **pVdbe**
- int **addrGosub**
- int **regGosub**
- int **regArg**
- int **eDelete**
- **WindowCsrAndReg** start
- **WindowCsrAndReg** current
- **WindowCsrAndReg** end

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.199 WindowCsrAndReg Struct Reference

Public Attributes

- int **csr**
- int **reg**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.200 WindowRewrite Struct Reference

Public Attributes

- **Window** * **pWin**
- **SrcList** * **pSrc**
- **ExprList** * **pSub**
- **Table** * **pTab**
- **Select** * **pSubSelect**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.201 With Struct Reference

Classes

- struct **Cte**

Public Attributes

- int **nCte**
- **With** * **pOuter**
- struct **With::Cte** **a** [1]

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.202 YYMINORTYPE Union Reference

Public Attributes

- int **yyinit**
- sqlite3ParserTOKENTYPE **yy0**
- **SrcList** * **yy47**
- u8 **yy58**
- struct **FrameBound** **yy77**
- **With** * **yy131**
- int **yy192**
- **Expr** * **yy202**
-
- struct {
 - int **value**
 - int **mask**
 } **yy207**
- struct **TrigEvent** **yy230**
- **ExprList** * **yy242**
- **Window** * **yy303**
- **Upsert** * **yy318**
- const char * **yy436**
- **TriggerStep** * **yy447**
- **Select** * **yy539**
- **IdList** * **yy600**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.203 yyParser Struct Reference

Public Attributes

- **yyStackEntry** * **yytos**
- sqlite3ParserARG_SDECL sqlite3ParserCTX_SDECL **yyStackEntry** **yystack** [YYSTACKDEPTH]
- **yyStackEntry** * **yystackEnd**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

3.204 yyStackEntry Struct Reference

Public Attributes

- YYACTIONTYPE **stato**
- YYCODETYPE **major**
- YYMINORTYPE **minor**

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

- CS_205_Software_Engineering/Group Project/project/museum/museum/project/sqlite3.c

Index

- Account, 9
 - Account, 10
 - add_row, 11
 - addNewUser, 11
 - getAllUsernames, 12
 - getNameFromId, 12
 - login, 12
 - newPIN, 13
 - remove_entries, 13
 - removeUser, 13
 - select_all, 14
 - select_cell, 14
 - select_elements_sql, 14
 - setAllUsernames, 14
 - update_value, 15
 - userData, 15
- add_row
 - Account, 11
 - Comment, 25
 - MuseumPage, 51
- addMuseum
 - MuseumPage, 51
- addNewUser
 - Account, 11
- Admin, 16
 - Admin, 16
- AggInfo, 17
- AggInfo::AggInfo_col, 17
- AggInfo::AggInfo_func, 18
- analysisInfo, 18
- AuthContext, 18
- AutoincInfo, 18
- AuxData, 19
- Bitvec, 19
- BtCursor, 19
- BtLock, 20
- Btree, 20
- BtreePayload, 21
- BtShared, 21
- BusyHandler, 22
- CallCount, 22
- CellArray, 22
- CellInfo, 23
- CollSeq, 23
- Column, 23
- Comment, 24
 - add_row, 25
 - Comment, 24
 - newComment, 25
 - remove_entries, 26
 - removeComment, 26
 - select_cells, 26
 - select_elements_sql, 27
 - setComment, 27
 - update_value, 27
- compareInfo, 28
- CountCtx, 28
- Curator, 29
 - Curator, 29
- DateTime, 30
- Db, 31
- DbFixer, 31
- DblquoteStr, 31
- DBTable, 32
- DBTool, 33
 - open_db, 33
- DistinctCtx, 34
- et_info, 34
- Expr, 35
- ExprList, 35
- ExprList::ExprList_item, 36
- FileChunk, 36
- FilePoint, 37
- FKey, 37
- FKey::sColMap, 67
- FrameBound, 37
- fts5_api, 38
- fts5_tokenizer, 38
- Fts5ExtensionApi, 38
- Fts5Phraselater, 39
- FuncDef, 39
- FuncDefHash, 40
- FuncDestructor, 40
- getAllUsernames
 - Account, 12
- getMuseumByCuratorId
 - MuseumPage, 51
- getMuseumChildrenId
 - MuseumPage, 52
- getMuseumChildrenNames
 - MuseumPage, 52
- getMuseumId
 - MuseumPage, 53
- getMuseumPageFromId

- MuseumPage, 53
- getMuseumSearchId
 - MuseumPage, 53
- getNameFromId
 - Account, 12
- Hash, 40
- Hash::_ht, 9
- HashElem, 41
- HiddenIndexInfo, 41
- IdList, 41
- IdList::IdList_item, 42
- IdxCover, 42
- IdxExprTrans, 42
- Incrblob, 43
- IncrMerger, 43
- Index, 43
- IndexSample, 44
- InitData, 44
- IntegrityCk, 45
- KeyInfo, 45
- LastValueCtx, 45
- login
 - Account, 12
- Lookaside, 46
- LookasideSlot, 46
- MemJournal, 46
- MemPage, 47
- MergeEngine, 48
- Module, 48
- MuseumPage, 48
 - add_row, 51
 - addMuseum, 51
 - getMuseumByCuratorId, 51
 - getMuseumChildrenId, 52
 - getMuseumChildrenNames, 52
 - getMuseumId, 53
 - getMuseumPageFromId, 53
 - getMuseumSearchId, 53
 - MuseumPage, 50
 - remove_entries, 54
 - select_cell_search, 54
- NameContext, 55
- newComment
 - Comment, 25
- newPIN
 - Account, 13
- NthValueCtx, 55
- NtileCtx, 56
- open_db
 - DBTool, 33
- Pager, 56
- PagerSavepoint, 58
- Parse, 58
- PCache, 59
- PCache1, 60
- PgFreeslot, 60
- PgHdr, 61
- PgHdr1, 61
- PGroup, 61
- PmaReader, 62
- PmaWriter, 62
- PragmaName, 62
- PragmaVtab, 63
- PragmaVtabCursor, 63
- PreUpdate, 63
- PrintfArguments, 64
- remove_entries
 - Account, 13
 - Comment, 26
 - MuseumPage, 54
- removeComment
 - Comment, 26
- removeUser
 - Account, 13
- RenameCtx, 64
- RenameToken, 64
- ReusableSpace, 64
- RowLoadInfo, 65
- RowSet, 65
- RowSetChunk, 65
- RowSetEntry, 66
- Savepoint, 66
- ScanStatus, 66
- Schema, 67
- Select, 67
- select_all
 - Account, 14
- select_cell
 - Account, 14
- select_cell_search
 - MuseumPage, 54
- select_cells
 - Comment, 26
- select_elements_sql
 - Account, 14
 - Comment, 27
- SelectDest, 68
- setAllUsernames
 - Account, 14
- setComment
 - Comment, 27
- SortCtx, 68
- SorterFile, 68
- SorterList, 69
- SorterRecord, 69
- SortSubtask, 69
- sqlite3, 70
- sqlite3::sqlite3InitInfo, 86
- sqlite3_api_routines, 72

sqlite3_backup, 77
sqlite3_context, 77
sqlite3_file, 77
sqlite3_index_info, 78
sqlite3_index_info::sqlite3_index_constraint, 78
sqlite3_index_info::sqlite3_index_constraint_usage, 78
sqlite3_index_info::sqlite3_index_orderby, 79
sqlite3_io_methods, 79
sqlite3_mem_methods, 80
sqlite3_module, 80
sqlite3_mutex, 81
sqlite3_mutex_methods, 81
sqlite3_pcache_methods, 81
sqlite3_pcache_methods2, 82
sqlite3_pcache_page, 82
sqlite3_rtree_geometry, 82
sqlite3_rtree_query_info, 83
sqlite3_snapshot, 83
sqlite3_str, 83
sqlite3_value, 84
sqlite3_value::MemValue, 47
sqlite3_vfs, 84
sqlite3_vtab, 85
sqlite3_vtab_cursor, 85
Sqlite3Config, 85
SQLiteThread, 86
SrcCount, 87
SrcList, 87
SrcList::SrcList_item, 87
Stat4Accum, 88
Stat4Sample, 89
SubProgram, 89
SubstContext, 89
SumCtx, 90

Table, 90
TableLock, 91
TabResult, 91
Token, 91
TrigEvent, 92
Trigger, 92
TriggerPrg, 92
TriggerStep, 93

unixFile, 93
unixFileId, 94
unixNodeInfo, 94
unixShm, 94
unixShmNode, 95
UnixUnusedFd, 95
UnpackedRecord, 95
update_value
 Account, 15
 Comment, 27
Upsert, 96
userData
 Account, 15

ValueNewStat4Ctx, 96

Vdbe, 96
VdbeCursor, 98
VdbeFrame, 98
VdbeOp, 99
VdbeOp::p4union, 56
VdbeOpList, 99
VdbeSorter, 100
Viewer, 100
 Viewer, 101
VtabCtx, 102
VTable, 102
vxworksFileId, 102

Wal, 103
WalCkptInfo, 103
WalHashLoc, 104
WalIndexHdr, 104
WalIterator, 104
WalIterator::WalSegment, 105
Walker, 105
WalWriter, 106
WhereAndInfo, 106
WhereClause, 106
WhereConst, 107
WhereExprMod, 107
WhereInfo, 107
WhereLevel, 108
WhereLoop, 109
WhereLoopBuilder, 109
WhereMaskSet, 110
WhereOrCost, 110
WhereOrInfo, 110
WhereOrSet, 110
WherePath, 111
WhereScan, 111
WhereTerm, 111
Window, 112
WindowCodeArg, 113
WindowCsrAndReg, 113
WindowRewrite, 113
With, 114
With::Cte, 29

YYMINORTYPE, 114
yyParser, 115
yyStackEntry, 115