PROCEDURE ORIENTED STYLE : MYSQLi

Sisoft Technologies Pvt Ltd
SRC E7, Shipra Riviera Bazar, Gyan Khand-3, Indirapuram, Ghaziabad
Website: www.sisoft.in, Email:info@sisoft.in
Phone: +91-9999-283-283

LEARNING TOPICS

- Connections
- Executing Statements
- Prepared statements
- Multiple Statements
- API Support for Transactions
- Metadata

- MYSQLi
- MYSQLi_stmt
- MYSQLi_result
- MYSQLi_driver
- MYSQLi_warning

CONNECTING TO DATABASE

```
mysqli_connect("localhost", “user", “password", “database");
```

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

if (!$link) {
    printf("Can't connect to localhost. Error: %s\n", mysqli_connect_error());
    exit();
}
```

Connection Error Handling

mysqli_connect_errno()

```
<?php
$link = mysqli_connect('localhost', 'fake_user', 'my_password', 'my_db');

if (!$link) {
    die('Connect Error: ' . mysqli_connect_errno());
}
?>
```

mysqli_connect_error()

```
<?php
$link = mysqli_connect('localhost', 'fake_user', 'my_password', 'my_db');

if (!$link) {
    die('Connect Error: ' . mysqli_connect_error());
}
?>
```

Executing Statements – mysqli_query()

```
/* Insert rows */
mysqli_query($link, "CREATE TABLE Language SELECT * from CountryLanguage");
printf("Affected rows (INSERT): %d\n", mysqli_affected_rows($link));

mysqli_query($link, "ALTER TABLE Language ADD Status int default 0");

/* update rows */
mysqli_query($link, "UPDATE Language SET Status=1 WHERE Percentage > 50");
printf("Affected rows (UPDATE): %d\n", mysqli_affected_rows($link));

/* delete rows */
mysqli_query($link, "DELETE FROM Language WHERE Percentage < 50");
printf("Affected rows (DELETE): %d\n", mysqli_affected_rows($link));
```

Executing Statements – error handling – mysqli_query()

mysqli_errno() / error()

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

if (!mysqli_query($link, "SET a=1")) {
    printf("Errorcode: %d\n", mysqli_errno($link));
}

/* close connection */
mysqli_close($link);
?>
```

Executing Statements – error handling – mysqli_query()

mysqli_error()

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world
");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

if (!mysqli_query($link, "SET a=1")) {
    printf("Error message: %s\n", mysqli_error($link));
}

/* close connection */
mysqli_close($link);
?>
```

Mysqli_insert_id ()

The mysqli_insert_id() function returns the ID generated by a query (usually INSERT) on a table with a column having the AUTO_INCREMENT attribute. If no INSERT or UPDATE statements were sent via this connection, or if the modified table does not have a column with the AUTO_INCREMENT attribute, this function will return zero.

```php
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

mysqli_query($link, "CREATE TABLE myCity LIKE City");

$query = "INSERT INTO myCity VALUES (NULL, 'Stuttgart', 'DEU', 'Stuttga
rt', 617000)";
mysqli_query($link, $query);

printf ("New Record has id %d.\n", mysqli_insert_id($link));

/* drop table */
mysqli_query($link, "DROP TABLE myCity");

/* close connection */
mysqli_close($link);
?>
```

Executing Statements – Fetch Result – mysqli_query()

mysqli_result

mysqli_num_rows():- mysqli_num_rows (mysqli_result $result) : int

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

if ($result = mysqli_query($link, "SELECT Code, Name FROM Country ORDER
BY Name")) {

    /* determine number of rows result set */
    $row_cnt = mysqli_num_rows($result);

    printf("Result set has %d rows.\n", $row_cnt);

    /* close result set */
    mysqli_free_result($result);
}

/* close connection */
mysqli_close($link);
?>
```

Executing Statements – Fetch Result – mysqli_query()

- **mysqli_fetch_array :-**

Returns an array that corresponds to the fetched row or NULL if there are no more rows for the resultset represented by the result parameter.
mysqli_fetch_array() is an extended version of the mysqli_fetch_row() function. In addition to storing the data in the numeric indices of the result array, the mysqli_fetch_array() function can also store the data in associative indices, using the field names of the result set as keys.

Parameters

result

Procedural style only: A result set identifier returned by mysqli_query(), mysqli_store_result() or mysqli_use_result().

resulttype

This optional parameter is a constant indicating what type of array should be produced from the current row data. The possible values for this parameter are the constants MYSQLI_ASSOC, MYSQLI_NUM, or MYSQLI_BOTH.

- **mysqli_fetch_all**

mysqli_fetch_all() fetches all result rows and returns the result set as an associative array, a numeric array, or both.

Parameters

result

Procedural style only: A result set identifier returned by mysqli_query(), mysqli_store_result() or mysqli_use_result().

resulttype

This optional parameter is a constant indicating what type of array should be produced from the current row data. The possible values for this parameter are the constants MYSQLI_ASSOC, MYSQLI_NUM, or MYSQLI_BOTH

Note:- Never use this method for fetching the data as it will affect the website or software.

- **mysqli_fetch_assoc**

Returns an associative array that corresponds to the fetched row or NULL if there are no more rows.

Parameters

result

Procedural style only: A result set identifier returned by mysqli_query(), mysqli_store_result() or mysqli_use_result().

- **mysqli_fetch_row**

Fetches one row of data from the result set and returns it as an enumerated array, where each column is stored in an array offset starting from 0 (zero). Each subsequent call to this function will return the next row within the result set, or NULL if there are no more rows.

Parameters

result

Procedural style only: A result set identifier returned by mysqli_query(), mysqli_store_result() or mysqli_use_result().

mysqli_fetch_all():- mysqli_fetch_all (mysqli_result $result [, int $resulttype = MYSQLI_NUM]) : mixed

mysqli_fetch_all() fetches all result rows and returns the result set as an associative array, a numeric array, or both.

Note:- Never use this method for fetching the data as it will affect the website or software.

mysqli_fetch_array():- mysqli_fetch_array (mysqli_result $result [, int $resulttype = MYSQLI_BOTH]) : mixed

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

$query = "SELECT Name, CountryCode FROM City ORDER by ID LIMIT 3";
$result = mysqli_query($link, $query);

/* numeric array */
$row = mysqli_fetch_array($result, MYSQLI_NUM);
printf ("%s (%s)\n", $row[0], $row[1]);

/* associative array */
$row = mysqli_fetch_array($result, MYSQLI_ASSOC);
printf ("%s (%s)\n", $row["Name"], $row["CountryCode"]);

/* associative and numeric array */
$row = mysqli_fetch_array($result, MYSQLI_BOTH);
printf ("%s (%s)\n", $row[0], $row["CountryCode"]);

/* free result set */
mysqli_free_result($result);

/* close connection */
mysqli_close($link);
?>
```

mysqli_fetch_assoc() :-   mysqli_fetch_assoc (mysqli_result $result) : array

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

$query = "SELECT Name, CountryCode FROM City ORDER by ID DESC
LIMIT 50,5";

if ($result = mysqli_query($link, $query)) {

    /* fetch associative array */
    while ($row = mysqli_fetch_assoc($result)) {
        printf ("%s (%s)\n", $row["Name"], $row["CountryCode"]);
    }

    /* free result set */
    mysqli_free_result($result);
}

/* close connection */
mysqli_close($link);
?>
```

mysqli_fetch_row():- mysqli_fetch_row (mysqli_result $result) : mixed

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

$query = "SELECT Name, CountryCode FROM City ORDER by ID DESC LIMIT 50,5";

if ($result = mysqli_query($link, $query)) {

    /* fetch associative array */
    while ($row = mysqli_fetch_row($result)) {
        printf ("%s (%s)\n", $row[0], $row[1]);
    }

    /* free result set */
    mysqli_free_result($result);
}

/* close connection */
mysqli_close($link);
?>
```

The MySQLi Extension Function Summary

Procedure oriented style

```
/* select all rows */
$result = mysqli_query($link, "SELECT CountryCode FROM Language");
printf("Affected rows (SELECT): %d\n", mysqli_affected_rows($link));

mysqli_free_result($result);

/* Delete table Language */
mysqli_query($link, "DROP TABLE Language");

/* close connection */
mysqli_close($link);
?>
```

Executing Statements – Prepare Result – mysqli_query()

mysqli_field_count()

Returns the number of columns for the most recent query on the connection represented by the link parameter. This function can be useful when using the mysqli_store_result() function to determine if the query should have produced a non-empty result set or not without knowing the nature of the query.

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "test");

mysqli_query($link, "DROP TABLE IF EXISTS friends");
mysqli_query($link, "CREATE TABLE friends (id int, name varchar(20))");

mysqli_query($link, "INSERT INTO friends VALUES (1,'Hartmut'), (2, 'Ulf')")
;

mysqli_real_query($link, "SELECT * FROM friends");

if (mysqli_field_count($link)) {
    /* this was a select/show or describe query */
    $result = mysqli_store_result($link);

    /* process resultset */
    $row = mysqli_fetch_row($result);

    /* free resultset */
    mysqli_free_result($result);
}

/* close connection */
mysqli_close($link);
?>
```

Summary of mysqli_result Methods

mysqli_result

mysqli_num_rows():- mysqli_num_rows (mysqli_result $result) : int

The behaviour of mysqli_num_rows() depends on whether buffered or unbuffered result sets are being used. For unbuffered result sets, mysqli_num_rows() will not return the correct number of rows until all the rows in the result have been retrieved.

```php
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

if ($result = mysqli_query($link, "SELECT Code, Name FROM Country ORDER
BY Name")) {

    /* determine number of rows result set */
    $row_cnt = mysqli_num_rows($result);

    printf("Result set has %d rows.\n", $row_cnt);

    /* close result set */
    mysqli_free_result($result);
}

/* close connection */
mysqli_close($link);
?>
```

My sqli Methods

Mysqli_autocommit()

mysqli_autocommit (mysqli $link , bool $mode) : bool

Turns on or off auto-commit mode on queries for the database connection.
To determine the current state of autocommit use the SQL command *SELECT @@autocommit*.

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

if (!$link) {
    printf("Can't connect to localhost. Error: %s\n", mysqli_connect_error());
    exit();
}

/* turn autocommit on */
mysqli_autocommit($link, TRUE);

if ($result = mysqli_query($link, "SELECT @@autocommit")) {
    $row = mysqli_fetch_row($result);
    printf("Autocommit is %s\n", $row[0]);
    mysqli_free_result($result);
}

/* close connection */
mysqli_close($link);
?>
```

Mysqli_query()

mysqli_query (mysqli $link , string $query [, int $resultmode = MYSQLI_STORE_RESULT]) : mixed

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* Create table doesn't return a resultset */
if (mysqli_query($link, "CREATE TEMPORARY TABLE myCity LIKE City")
=== TRUE) {
    printf("Table myCity successfully created.\n");
}

/* Select queries return a resultset */
if ($result = mysqli_query($link, "SELECT Name FROM City LIMIT 10")) {
    printf("Select returned %d rows.\n", mysqli_num_rows($result));
```

Mysqli_query() continued…

mysqli_query (mysqli $link , string $query [, int $resultmode = MYSQLI_STORE_RESULT]) : mixed

```
/* free result set */
    mysqli_free_result($result);
}

/* If we have to retrieve large amount of data we use MYSQLI_USE_RESU
LT */
if ($result = mysqli_query($link, "SELECT * FROM City", MYSQLI_USE_R
ESULT)) {

    /* Note, that we can't execute any functions which interact with the
       server until result set was closed. All calls will return an
       'out of sync' error */
    if (!mysqli_query($link, "SET @a:='this will not work'")) {
        printf("Error: %s\n", mysqli_error($link));
    }
    mysqli_free_result($result);
}

mysqli_close($link);
?>
```

Mysqli_multi_query()

mysqli_multi_query (mysqli $link , string $query) : bool ‘

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}
$query  = "SELECT CURRENT_USER();";
$query .= "SELECT Name FROM City ORDER BY ID LIMIT 20, 5";

/* execute multi query */
if (mysqli_multi_query($link, $query)) {
    do {
        /* store first result set */
        if ($result = mysqli_store_result($link)) {
            while ($row = mysqli_fetch_row($result)) {
                printf("%s\n", $row[0]);
            }
            mysqli_free_result($result);
        }
        /* print divider */
        if (mysqli_more_results($link)) {
            printf("-----------------\n");
        }
    } while (mysqli_next_result($link));
}
/* close connection */
mysqli_close($link);
?>
```

Mysqli_commit()

mysqli_commit (mysqli $link [, int $flags = 0 [, string $name]]) : bool

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "test");

/* check connection */
if (!$link) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* set autocommit to off */
mysqli_autocommit($link, FALSE);

mysqli_query($link, "CREATE TABLE Language LIKE CountryLanguage");

/* Insert some values */
mysqli_query($link, "INSERT INTO Language VALUES ('DEU', 'Bavarian', 'F', 11.
2)");
mysqli_query($link, "INSERT INTO Language VALUES ('DEU', 'Swabian', 'F', 9.4
)");

/* commit transaction */
if (!mysqli_commit($link)) {
    print("Transaction commit failed\n");
    exit();
}

/* close connection */
mysqli_close($link);
?>
```

mysqli_real_escape_string()

mysqli_real_escape_string (mysqli $link , string $escapestr) : string

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

mysqli_query($link, "CREATE TEMPORARY TABLE myCity LIKE City");

$city = "'s Hertogenbosch";

/* this query will fail, cause we didn't escape $city */
if (!mysqli_query($link, "INSERT into myCity (Name) VALUES ('$city')")) {
    printf("Error: %s\n", mysqli_sqlstate($link));
}

$city = mysqli_real_escape_string($link, $city);

/* this query with escaped $city will work */
if (mysqli_query($link, "INSERT into myCity (Name) VALUES ('$city')")) {
    printf("%d Row inserted.\n", mysqli_affected_rows($link));
}

mysqli_close($link);
?>
```

mysqli_select_db()

mysqli_select_db (mysqli $link , string $dbname) : bool

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "test");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

/* return name of current default database */
if ($result = mysqli_query($link, "SELECT DATABASE()")) {
    $row = mysqli_fetch_row($result);
    printf("Default database is %s.\n", $row[0]);
    mysqli_free_result($result);
}

/* change db to world db */
mysqli_select_db($link, "world");

/* return name of current default database */
if ($result = mysqli_query($link, "SELECT DATABASE()")) {
    $row = mysqli_fetch_row($result);
    printf("Default database is %s.\n", $row[0]);
    mysqli_free_result($result);
}

mysqli_close($link);
?>
```

PREPARED STATEMENTS

Summary of mysqli_stmt methods

mysqli_stmt_affected_rows()

mysqli_stmt_affected_rows (mysqli_stmt $stmt) : int

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}
/* create temp table */
mysqli_query($link, "CREATE TEMPORARY TABLE myCountry LIKE Country");
$query = "INSERT INTO myCountry SELECT * FROM Country WHERE Code LIKE ?";

/* prepare statement */
if ($stmt = mysqli_prepare($link, $query)) {

    /* Bind variable for placeholder */
    $code = 'A%';
    mysqli_stmt_bind_param($stmt, "s", $code);

    /* execute statement */
    mysqli_stmt_execute($stmt);
    printf("rows inserted: %d\n", mysqli_stmt_affected_rows($stmt));

    /* close statement */
    mysqli_stmt_close($stmt);
}
/* close connection */
mysqli_close($link);
?>
```

mysqli_stmt_prepare

mysqli_stmt_prepare (mysqli_stmt $stmt , string $query) : bool

Prepares the SQL query pointed to by the null-terminated string query.
The parameter markers must be bound to application variables using mysqli_stmt_bind_param() and/ormysqli_stmt_bind_result() before executing the statement or fetching rows.

Parameters

stmt
Procedural style only: A statement identifier returned by mysqli_stmt_init().

query
The query, as a string. It must consist of a single SQL statement.
You can include one or more parameter markers in the SQL statement by embedding question mark (?) characters at the appropriate positions.

Create Prepared statement

mysqli_stmt_init

mysqli_stmt_init (mysqli $link) : mysqli_stmt

Allocates and initializes a statement object suitable for mysqli_stmt_prepare().

Parameters

link

Procedural style only: A link identifier returned by mysqli_connect() or mysqli_init()

Bind statement

mysqli_stmt_bind_param

mysqli_stmt_bind_param (mysqli_stmt $stmt , string $types , mixed &$var1 [, mixed &$...]) : bool

Bind variables for the parameter markers in the SQL statement that was passed to mysqli_prepare().

Parameters

stmt

Procedural style only: A statement identifier returned by mysqli_stmt_init().

types

A string that contains one or more characters which specify the types for the corresponding bind variables:

Execute Statement

mysqli_stmt_execute

mysqli_stmt_execute (mysqli_stmt $stmt) : bool

Executes a query that has been previously prepared using the mysqli_prepare() function. When executed any parameter markers which exist will automatically be replaced with the appropriate data.
If the statement is *UPDATE*, *DELETE*, or *INSERT*, the total number of affected rows can be determined by using the mysqli_stmt_affected_rows() function. Likewise, if the query yields a result set the mysqli_stmt_fetch() function is used.

Parameters

stmt

Procedural style only: A statement identifier returned by mysqli_stmt_init().

mysqli_stmt_bind_result

mysqli_stmt_bind_result (mysqli_stmt $stmt , mixed &$var1 [, mixed &$...]) : bool

Binds columns in the result set to variables.
When mysqli_stmt_fetch() is called to fetch data, the MySQL client/server protocol places the data for the bound columns into the specified variables var1, ...

Parameters

stmt

Procedural style only: A statement identifier returned by mysqli_stmt_init().

var1

The variable to be bound.

mysqli_stmt_get_result

mysqli_stmt_get_result (mysqli_stmt $stmt) : mysqli_result

Call to return a result set from a prepared statement query.

Parameters

stmt

Procedural style only: A statement identifier returned by mysqli_stmt_init().

Fetch statement

mysqli_stmt_fetch

mysqli_stmt_fetch (mysqli_stmt $stmt) : bool

Fetch the result from a prepared statement into the variables bound by mysqli_stmt_bind_result().

Parameters

stmt

Procedural style only: A statement identifier returned by mysqli_stmt_init().

mysqli_stmt_close

mysqli_stmt_close (mysqli_stmt $stmt) : bool

Closes a prepared statement. mysqli_stmt_close() also deallocates the statement handle. If the current statement has pending or unread results, this function cancels them so that the next query can be executed.

Parameters

stmt

Procedural style only: A statement identifier returned by mysqli_stmt_init().

Example of Prepared Statements

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");
/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}
$city = "Amersfoort";
/* create a prepared statement */
$stmt = mysqli_stmt_init($link);
if (mysqli_stmt_prepare($stmt, 'SELECT District FROM City WHERE Name=?')) {
/* bind parameters for markers */
    mysqli_stmt_bind_param($stmt, "s", $city);
  /* execute query */
    mysqli_stmt_execute($stmt);

    /* bind result variables */
    mysqli_stmt_bind_result($stmt, $district);

    /* fetch value */
    mysqli_stmt_fetch($stmt);

    printf("%s is in district %s\n", $city, $district);

    /* close statement */
    mysqli_stmt_close($stmt);
}
/* close connection */
mysqli_close($link);
?>
```

mysqli_stmt_errno():- mysqli_stmt_errno (mysqli_stmt $stmt) : int

```
<?php
/* Open a connection */
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

mysqli_query($link, "CREATE TABLE myCountry LIKE Country");
mysqli_query($link, "INSERT INTO myCountry SELECT * FROM Country");


$query = "SELECT Name, Code FROM myCountry ORDER BY Name";
if ($stmt = mysqli_prepare($link, $query)) {

    /* drop table */
    mysqli_query($link, "DROP TABLE myCountry");

    /* execute query */
    mysqli_stmt_execute($stmt);

    printf("Error: %d.\n", mysqli_stmt_errno($stmt));

    /* close statement */
    mysqli_stmt_close($stmt);
}
/* close connection */
mysqli_close($link);
?>
```

mysqli_stmt_error():- mysqli_stmt_error (mysqli_stmt $stmt) : string

Returns the error code for the most recently invoked statement function that can succeed or fail.

```
<?php
/* Open a connection */
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

mysqli_query($link, "CREATE TABLE myCountry LIKE Country");
mysqli_query($link, "INSERT INTO myCountry SELECT * FROM Country");

$query = "SELECT Name, Code FROM myCountry ORDER BY Name";
if ($stmt = mysqli_prepare($link, $query)) {

    /* drop table */
    mysqli_query($link, "DROP TABLE myCountry");

    /* execute query */
    mysqli_stmt_execute($stmt);

    printf("Error: %s.\n", mysqli_stmt_error($stmt));

    /* close statement */
    mysqli_stmt_close($stmt);
}
/* close connection */
mysqli_close($link);
?>
```

mysqli_stmt_field_count()

mysqli_stmt_field_count (mysqli_stmt $stmt) : int

Warning:- This function is currently not documented; only its argument list is available.

mysqli_stmt_insert_id()

mysqli_stmt_insert_id (mysqli_stmt $stmt) : mixed

Warning:- This function is currently not documented; only its argument list is available.

mysqli_stmt_num_rows():- mysqli_stmt_num_rows (mysqli_stmt $stmt) : int

Returns a string containing the error message for the most recently invoked statement function that can succeed or fail.

```php
<?php
/* Open a connection */
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

$query = "SELECT Name, CountryCode FROM City ORDER BY Name LIMIT 20
";
if ($stmt = mysqli_prepare($link, $query)) {

    /* execute query */
    mysqli_stmt_execute($stmt);

    /* store result */
    mysqli_stmt_store_result($stmt);

    printf("Number of rows: %d.\n", mysqli_stmt_num_rows($stmt));

    /* close statement */
    mysqli_stmt_close($stmt);
}

/* close connection */
mysqli_close($link);
?>
```

mysqli_stmt_execute()

mysqli_stmt_execute (mysqli_stmt $stmt) : bool

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

mysqli_query($link, "CREATE TABLE myCity LIKE City");

/* Prepare an insert statement */
$query = "INSERT INTO myCity (Name, CountryCode, District) VALUES (?,?,?)"
;
$stmt = mysqli_prepare($link, $query);

mysqli_stmt_bind_param($stmt, "sss", $val1, $val2, $val3);

$val1 = 'Stuttgart';
$val2 = 'DEU';
$val3 = 'Baden-Wuerttemberg';

/* Execute the statement */
mysqli_stmt_execute($stmt);

$val1 = 'Bordeaux';
$val2 = 'FRA';
$val3 = 'Aquitaine';
```

mysqli_stmt_execute() continued….

mysqli_stmt_execute (mysqli_stmt $stmt) : bool

```
/* Execute the statement */
mysqli_stmt_execute($stmt);

/* close statement */
mysqli_stmt_close($stmt);

/* retrieve all rows from myCity */
$query = "SELECT Name, CountryCode, District FROM myCity";
if ($result = mysqli_query($link, $query)) {
    while ($row = mysqli_fetch_row($result)) {
        printf("%s (%s,%s)\n", $row[0], $row[1], $row[2]);
    }
    /* free result set */
    mysqli_free_result($result);
}

/* remove table */
mysqli_query($link, "DROP TABLE myCity");

/* close connection */
mysqli_close($link);
?>
```

mysqli_stmt_fetch() :- mysqli_stmt_fetch (mysqli_stmt $stmt) : bool

```
<?php
$link = mysqli_connect("localhost", "my_user", "my_password", "world");

/* check connection */
if (mysqli_connect_errno()) {
    printf("Connect failed: %s\n", mysqli_connect_error());
    exit();
}

$query = "SELECT Name, CountryCode FROM City ORDER by ID DESC LIMIT
150,5";

if ($stmt = mysqli_prepare($link, $query)) {

    /* execute statement */
    mysqli_stmt_execute($stmt);

    /* bind result variables */
    mysqli_stmt_bind_result($stmt, $name, $code);

    /* fetch values */
    while (mysqli_stmt_fetch($stmt)) {
        printf ("%s (%s)\n", $name, $code);
    }

    /* close statement */
    mysqli_stmt_close($stmt);
}

/* close connection */
mysqli_close($link);
?>
```

Summary of mysqli_driver methods

mysqli_embedded_server_end

mysqli_embedded_server_end (void) : void

mysqli_embedded_server_start

mysqli_embedded_server_start (int $start , array $arguments , array $groups) : bool

FOR OTHER SQLI METHODS AND STATEMENTS SUMMARY PLEASE CHECK THE BELOW LINK AND IMPLEMENT THEM

https://www.php.net/manual/en/mysqli.summary.php

THANK YOU