begin-transaction

mysqli::begin_transaction

mysqli_begin_transaction

Starts a transaction

ENTITY

ENTITY

public

bool

mysqli::begin_transaction

int

flags

0

string

null

name

ENTITY

ENTITY:

bool

mysqli_begin_transaction

mysqli

mysql

int

flags

0

string

null

name

ENTITY

Begins a transaction. Requires the InnoDB engine (it is enabled by default). For additional details about how MySQL transactions work, see ENTITY.

ENTITY

`flags`
Valid flags are:

- `MYSQLI_TRANS_START_READ_ONLY`: Start the transaction as "START TRANSACTION READ ONLY". Requires MySQL 5.6 and above.
- `MYSQLI_TRANS_START_READ_WRITE`: Start the transaction as "START TRANSACTION READ WRITE". Requires MySQL 5.6 and above.
- `MYSQLI_TRANS_START_WITH_CONSISTENT_SNAPSHOT`: Start the transaction as "START TRANSACTION WITH CONSISTENT SNAPSHOT".

`name`
Savepoint name for the transaction.

ENTITY

ENTITY

ENTITY

ENTITY	ENTITY
8.0.0	`name` is now nullable.

ENTITY

mysqli::begin_transaction example

ENTITY

```
<?php

/* Tell mysqli to throw an exception if an error occurs */
mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);

$mysqli = new mysqli("localhost", "my_user", "my_password", "world");

/* The table engine has to support transactions */
$mysqli->query("CREATE TABLE IF NOT EXISTS language (
    Code text NOT NULL,
    Speakers int(11) NOT NULL
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;");

/* Start transaction */
$mysqli->begin_transaction();

try {
    /* Insert some values */
    $mysqli->query("INSERT INTO language(Code, Speakers) VALUES ('DE', 42000123)");

    /* Try to insert invalid values */
    $language_code = 'FR';
    $native_speakers = 'Unknown';
    $stmt = $mysqli->prepare('INSERT INTO language(Code, Speakers) VALUES (?,?)');
    $stmt->bind_param('ss', $language_code, $native_speakers);
    $stmt->execute();

    /* If code reaches this point without errors then commit the data in the database */
    $mysqli->commit();
} catch (mysqli_sql_exception $exception) {
    $mysqli->rollback();

    throw $exception;
}
```

ENTITY

```
<?php

/* Tell mysqli to throw an exception if an error occurs */
mysqli_report(MYSQLI_REPORT_ERROR | MYSQLI_REPORT_STRICT);

$mysqli = mysqli_connect("localhost", "my_user", "my_password", "world");

/* The table engine has to support transactions */
mysqli_query($mysqli, "CREATE TABLE IF NOT EXISTS language (
    Code text NOT NULL,
    Speakers int(11) NOT NULL
    ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;");

/* Start transaction */
mysqli_begin_transaction($mysqli);

try {
    /* Insert some values */
    mysqli_query($mysqli, "INSERT INTO language(Code, Speakers) VALUES ('DE', 42000123)");

    /* Try to insert invalid values */
    $language_code = 'FR';
    $native_speakers = 'Unknown';
    $stmt = mysqli_prepare($mysqli, 'INSERT INTO language(Code, Speakers) VALUES (?,?)');
    mysqli_stmt_bind_param($stmt, 'ss', $language_code, $native_speakers);
    mysqli_stmt_execute($stmt);

    /* If code reaches this point without errors then commit the data in the database */
    mysqli_commit($mysqli);
} catch (mysqli_sql_exception $exception) {
    mysqli_rollback($mysqli);

    throw $exception;
}
```

ENTITY

> This function does not work with non transactional table types (like MyISAM or ISAM).

ENTITY

`mysqli_autocommit`, `mysqli_commit`, `mysqli_rollback`

□□□ #2
□ □□□ □□□ 2026-10-31 17:55:28 UTC
□ □□□ □□□ 2026-10-31 17:55:29 UTC