

NOTE: for class db_manage till version 1.0 only

INSERT DB() METHOD DOCUMENTATION

Insert_db(Arg1,Arg2,Arg3)

Note: Marked * are Compulsory other can be NULL

➤ Arg1 *

Name of the table on which operation has to perform.

Data type:- String

Example:

\$table= 'test' ;

➤ Arg2

DATA to insert in the table.

Data type:- Associative Array

Example:

```
$where=array(  
    'id_of_user'=>'Rahul Sharma',  
    'email_of_user'=>'rahul66@gmail.com', );
```

➤ Arg3*

Object module from the connection of PDO connection method.

Data type:- Object

Return type: Associative array.

Acknowledgement: Is returned as **associative array** in the return array variable.

The acknowledgement returns can be accessed as following way:

```
echo $return['genre'];
```

Extra acknowledgement keys= genre, type, desc, code, **rows_affected**(Note:Only available if no error occurred in the method)

Known Exception :

Fatal error:- Call to a Member function prepare() on a non-object.

Cause:- Wrong object or no object is passed to the method.

SELECT DB() METHOD DOCUMENTATION

select_db(Arg1,Arg2,Arg3,Arg4,Arg5,Arg6,Arg7)

Note: Marked * are Compulsory other can be NULL

➤ Arg1 *

Name of the table on which operation has to perform.

Data type:- String

Example:

\$table= 'test' ;

➤ Arg2

Where clause for the query.

Data type:- Associative Array

Example:

```
$where=array(  
    'id_of_user'=>8 );
```

➤ Arg3

Skip LIMIT token for the query[LIMIT ?,20].

Data type:- Integer

Example:

```
$starting_limit=10;
```

➤ Arg4

Max LIMIT token for the query[LIMIT 1,?].

Data type:- Integer

Example:

```
$ending_limit=10;
```

➤ Arg5

Tell the function what to select from the table.

Data type:- Single dimension array

Example:

```
$select[0]='id_of_user'
```

```
$select[1]='phone_of_user'
```

➤ Arg6

Order by clause for the query, The string first word should be proper field name.

Data type:- String

Example:

```
$order_by='id_of_user DESC'
```

Note: DESC = Descending

➤ Arg7*

Object module from the connection of PDO connection method.

Data type:- Object

Return type: Associative multi dimension array.

Returns: Selected data from the database table is returned in the form of **associative multi dimensional array**.

Acknowledgement: Is also returned as **associative multi dimensional array** in the return array variable.

The acknowledgement returns can be accessed as following way:

```
echo $return['ack_help']['genre'];
```

Extra acknowledgement keys= genre,type,desc,code

Known Exception :

Fatal error:- Call to a Member function prepare() on a non-object.

Cause:- Wrong object or no object is passed to the method.

UPDATE_DB() METHOD DOCUMENTATION

update_db(Arg1,Arg2,Arg3,Arg4)

Note: Marked * are Compulsory other can be NULL

➤ Arg1 *

Name of the table on which operation has to perform.

Data type:- String

Example:

\$table= 'test' ;

➤ Arg2

Where clause for the query.

Data type:- Associative Array

Example:

```
$where=array(  
    'id_of_user'=>8 );
```

➤ Arg3*

DATA to update in the table.

Data type:- Associative Array

Example:

```
$where=array(  
    'id_of_user'=>'Rahul Sharma',  
    'email_of_user'=>'rahul66@gmail.com', );
```

➤ Arg4*

Object module from the connection of PDO connection method.

Data type:- Object

Return type: Associative array.

Acknowledgement: Is returned as **associative array** in the return array variable.

The acknowledgement returns can be accessed as following way:

```
echo $return['genre'];
```

Extra acknowledgement keys= genre, type, desc, code, **rows_affected**(Note:Only available if no error occurred in the method)

Known Exception :

Fatal error:- Call to a Member function prepare() on a non-object.

Cause:- Wrong object or no object is passed to the method.

DELETE_DB() METHOD DOCUMENTATION

delete_db(Arg1,Arg2,Arg3)

Note: Marked * are Compulsory other can be NULL

➤ Arg1 *

Name of the table on which operation has to perform

Data type:- String

Example:

\$table= 'test' ;

➤ Arg2

Where clause for the query.

Data type:- Associative Array

Example:

\$where=array(
 'id_of_user'=>8);

➤ Arg4*

Object module from the connection of PDO connection method.

Data type:- Object

Return type: Associative array.

Acknowledgement: Is returned as **associative array** in the return array variable.

The acknowledgement returns can be accessed as following way:

echo \$return['genre'];

Extra acknowledgement keys= genre, type, desc, code, **rows_affected**(Note:Only available if no error occurred in the method)

Known Exception :

Fatal error:- Call to a Member function prepare() on a non-object.

Cause:- Wrong object or no object is passed to the method.

Note 1: If rows_affected = 0 when data is deleted successfully mean there is no WHERE clause matching field in the database right now.

Note 2: If where parameter is NULL, whole columns will be deleted from database, so be carefull before proceeding.

UPLOAD_DB() METHOD DOCUMENTATION

upload_db(Arg1,Arg2,Arg3,Arg4)

This function is similar to **update_db()** function, but only these differences.

Arg3 = This argument should be used to Update/add only BLOB type data in database. Any other type of data is not tested through this function, You can use update function for string and integer values injection into database.

Known Cases/Exceptions:

Exceptions are common as previous functions

Case no 1: BLOB Field size is dependent on the database engine and type, sometimes if you try to push data greater then the field size, row_affected will be 0[row_affected==0] but acknowledgement code will be 1.

Extra Help about acknowledgement arrays

[genre]

Possible values "error" or "success".

Data type= String

[type]

Possible values are dynamic, changes time to time according to function.

Data type= String

[desc]

Possible values are dynamic, changes time to time according to function.

Data type= String

[code]

Possible values 0, 1, 107(till version 1.0)

Data type= Integer

0 When a error occurs that is pre-defined

1 When no error occurs

107 When a unknown Exception occurs

[rows_affected]

Please note this key is available only in some cases only, please refer to function docs for more info.

Data type=Integer

Best way to loop select_db() method returns:

```
$rep = select_db(...);
```

```
foreach ($rep as $key=>$value) {
```

```
    if(!strcmp($key,'ack_help')){ // all acknowledgement variables will be available here
```

```
        echo '<br>'.$value['genre'].'....'.$value['desc'];
```

```
    }
```

```
else{
```

```
    foreach($value as $key2=>$value2){//all data will be available here
```

```
    echo '<br>' . $key2.'=>'.$value2;
```

```
    }
```

```
}
```

```
}
```