jinyPHP – 18. 데이터 삭제

**18. 데이터 삭제**

=========================

데이터를 조작하는 마지막 쿼리 명령에 대해서 학습해 보도록 합니다. DELETE명령은 테이블의 데이터를 삭제하는 쿼리 명령어 입니다. DELETE 는 데이터의 전체를 삭제 합니다. 또는 WHERE 조건을 같이 사용하여 특정 한 개 또는 범위를 지정하여 삭제를 할 수 있습니다.

삭제된 데이터는 다시 되돌리 수 없습니다. 삭제명령시에는 반드시 조건과 내용을 잘 확인후에 명령을 실행하여야 합니다.

**18.1 전체삭제**

=========================

기본적으로 DELETE 쿼리 명령 뒤에 아무런 조건이 포함되지 않으면 테이블의 전체 데이터를 삭제합니다. 테이블의 전체 데이터를 삭제할 때는 꼭 다시한번 명령을 확인후에 실행을 하여야 합니다.

**쿼리문법)**

|  |
| --- |
| DELETE FROM table\_name;  또는  DELETE \* FROM table\_name; |

**18.1.1 쿼리 실습**

=========================

테이블의 데이터 모두를 삭제해 보도록 합니다. 임시로 작업한 members2 테이블의 내용을 확인해 봅니다.

|  |
| --- |
| mysql> select \* from members2;  +----+----------+-----------+-------------+-------+---------+  | Id | LastName | FirstName | Address | City | Country |  +----+----------+-----------+-------------+-------+---------+  | 1 | hojin | lee | shinchon | seoul | korea |  | 2 | jiny | NULL | NULL | NULL | korea |  | 3 | james | kim | youngdun-po | seoul | korea |  | 4 | eric | jung | seocho-gu | seoul | korea |  | 5 | hojin | NULL | NULL | NULL | NULL |  | 6 | jiny | NULL | NULL | NULL | NULL |  | 7 | james | NULL | NULL | NULL | NULL |  | 8 | eric | NULL | NULL | NULL | NULL |  +----+----------+-----------+-------------+-------+---------+  8 rows in set (0.00 sec) |

여러 개의 데이터가 포함되어 있습니다.

**DELETE FROM members2;**

위의 쿼리 명령은members2 테이블의 데이터 전체를 삭제를 합니다.

|  |
| --- |
| mysql> DELETE FROM members2;  Query OK, 0 rows affected (0.01 sec) |

명령이 정상적으로 수행이 되었습니다. 테이블의 모든 데이터가 삭제가 되었습니다. 삭제된 데이터는 복구할 수 없습니다.

또한 데이터를 삭제하였다고 해서 프라이머리 키의 Auto\_inceament 속성의 번호를 초기화 되지 않습니다. 추가로 데이터를 삽입을 하게 되면, 이전에 프라이머리 Id 값 이후부터 자동으로 증가되어 데이터가 삽입이 됩니다.

**18.1.2 PHP 실습**

=========================

PHP 코드를 이용하여 데이터를 삭제해 봅니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| // 데이터 전체를 삭제합니다.  public function deleteAll($tbname)  {  if ($tbname) {  $queryString = "DELETE FROM $tbname";  // 쿼리를 전송합니다.  if (mysqli\_query($this->dbcon, $queryString)=== TRUE) {  $this->msgEcho("쿼리성공] ".$queryString);  $this->msgEcho(" 데이터 전체 삭제!");  // 객체 반환, 매서드체인  return $this;  } else {  $this->msgEcho("Error] ".$queryString);  }  } else {  $this->msgEcho("Error] 테이블 이름이 없습니다.");  }  } |

**예제파일) delete-01.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $tbname = "members2";  $db->deleteAll($tbname);  ?> |

출력화면)

mysql connected!

쿼리성공] DELETE FROM members2

데이터 전체 삭제!

**18.2 조건삭제**

=========================

테이블의 데이터를 삭제할 때 조건을 추가로 지정하지 않으면 모든 데이터가 삭제가 됩니다. 이는 매우 위험한 삭제 방법입니다.

실제적으로 모든 데이터를 삭제를 원하는 경우는 별로 없습니다. 데이터를 삭제할 때는 항상 조건을 선택하여 삭제를 하는 것이 좋습니다.

**쿼리문법)**

|  |
| --- |
| DELETE FROM 테이블이름 WHERE 컬럼명 = 컬럼데이터 |

또한 조건을 지정하더라도 정확한 데이터의 열을 지정하였는지 꼭 확인하는 습관이 필요로 합니다.

**18.2.1 쿼리 실습**

=========================

테이블의 열을 지정하여 데이터를 삭제해 봅니다. members1 테이블에는 여러 데이터들이 들어가 있습니다.

|  |
| --- |
| mysql> select \* from members1;  +----+----------+-----------+-------------+-------+---------+  | Id | LastName | FirstName | Address | City | Country |  +----+----------+-----------+-------------+-------+---------+  | 1 | hojin | lee | shinchon | seoul | korea |  | 2 | jiny | NULL | NULL | NULL | korea |  | 3 | james | kim | youngdun-po | seoul | korea |  | 4 | eric | jung | seocho-gu | seoul | korea |  +----+----------+-----------+-------------+-------+---------+  4 rows in set (0.00 sec) |

이중에서 Id 값이 4인 데이터만 삭제해 보겠습니다.

DELETE FROM members1 where Id=4;

|  |
| --- |
| mysql> DELETE FROM members1 where Id=4;  Query OK, 1 row affected (0.01 sec) |

삭제가 정상적으로 처리되었습니다. 다시 한번 select 를 통하여 데이터의 목록을 확인해 봅니다.

|  |
| --- |
| mysql> select \* from members1;  +----+----------+-----------+-------------+-------+---------+  | Id | LastName | FirstName | Address | City | Country |  +----+----------+-----------+-------------+-------+---------+  | 1 | hojin | lee | shinchon | seoul | korea |  | 2 | jiny | NULL | NULL | NULL | korea |  | 3 | james | kim | youngdun-po | seoul | korea |  +----+----------+-----------+-------------+-------+---------+  3 rows in set (0.00 sec) |

**18.2.2 PHP 실습**

=========================

PHP 코드를 통하여 선택한 데이터만 삭제를 해봅니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| // 선택한 데이터 Id를 삭제합니다.  public function deleteID($tbname, $Id)  {  if ($tbname) {  if($Id){  $queryString = "DELETE FROM $tbname where Id = '$Id'";  // 쿼리를 전송합니다.  if (mysqli\_query($this->dbcon, $queryString)=== TRUE) {  $this->msgEcho("쿼리성공] ".$queryString);  $this->msgEcho(" 데이터 $Id 삭제!");  // 객체 반환, 매서드체인  return $this;  } else {  $this->msgEcho("Error] ".$queryString);  }  } else {  $this->msgEcho("Error] 삭제할 Id값이 없습니다.");  }  } else {  $this->msgEcho("Error] 테이블 이름이 없습니다.");  }  } |

**예제파일) delete-02.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $tbname = "members2";  $db->deleteID($tbname, 2);  ?> |

출력화면)

mysql connected!

쿼리성공] DELETE FROM members2 where Id = '2'

데이터 2 삭제!

이처럼 항상 데이터를 삭제를 할때는 고유 Id 값 또는 조건을 이용하여 삭제를 하는 것이 안전합니다. 실수로 전체 데이터를 삭제한다면, 복구하기 힘듭니다.

또는 특정 문자열을 검색하여 삭제를 할 수도 있습니다.

**DELETE FROM Customers WHERE city='seoul';**

조건으로 문자열을 지정을 하는 것은 데이터의 조건 범위를 지정하는 것과 같습니다. 고객테이블에서 도시가 seoul인 모든 데이터를 삭제를 합니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| // 조건에 맞는 데이터를 모두 삭제합니다.  public function deleteWhere($tbname, $where)  {  if ($tbname){  if($where){  $queryString = "DELETE FROM $tbname where $where";  // 쿼리를 전송합니다.  if (mysqli\_query($this->dbcon, $queryString)=== TRUE){  $this->msgEcho("쿼리성공] ".$queryString);  $this->msgEcho(" 데이터 조건 삭제!");  // 객체 반환, 매서드체인  return $this;  } else {  $this->msgEcho("Error] ".$queryString);  }  } else {  $this->msgEcho("Error] 삭제조건이 없습니다.");  }  } else {  $this->msgEcho("Error] 테이블 이름이 없습니다.");  }  } |

**예제파일) delete-03.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $tbname = "members2";  $db->deleteWhere($tbname, "FirstName = 'jiny'");  ?> |

출력화면)

mysql connected!

쿼리성공] DELETE FROM members2 where FirstName = 'jiny'

데이터 조건 삭제!

**17.3 정렬삭제**

=========================

데이터를 삭제할 때 또한 범위를 지정한 영역의 데이터만 삭제를 하게 됩니다. 데이터를 정렬 키워드 또는 제한 키워드를 같이 사용하여 데이터를 삭제할 수 있습니다.

**쿼리문법)**

|  |
| --- |
| DELETE FROM 테이블이름 ORDER BY 컬럼명 LIMIT 제한갯수; |

**17.3.1 PHP 코드**

=========================

PHP 코드를 통해서 데이터를 삭제해 봅니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| // 최상의 데이터 몇개만 삭제합니다.  public function deleteLimit($tbname, $num, $orderby)  {  if ($tbname) {  if($num) {  if ($orderby){  $queryString = "DELETE FROM $tbname ORDER BY $orderby LIMIT $num";  } else {  $queryString = "DELETE FROM $tbname LIMIT $num";  }    // 쿼리를 전송합니다.  if (mysqli\_query($this->dbcon, $queryString)=== TRUE) {  $this->msgEcho("쿼리성공] ".$queryString);  $this->msgEcho(" 데이터 $Id 삭제!");  // 객체 반환, 매서드체인  return $this;  } else {  $this->msgEcho("Error] ".$queryString);  }  } else {  $this->msgEcho("Error] 삭제할 테이터 갯수가 없습니다.");  }  } else {  $this->msgEcho("Error] 테이블 이름이 없습니다.");  }  } |

**예제파일) delete-04.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $tbname = "members3";  // 최상위 데이터 2개만 삭제합니다.  $db->deleteLimit($tbname, 2);  // LastName 정렬후에 데이터 2개만 삭제합니다.  $db->deleteLimit($tbname, 2, "LastName desc");  ?> |

출력화면)

mysql connected!

쿼리성공] DELETE FROM members3 ORDER BY LastName desc LIMIT 2

데이터 삭제!