jinyPHP 2권 - 019. union

**19. UNION**

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

지금까지의 데이터 작업들은 모두 1개의 테이블의 기준으로 작업을 하였습니다. 관계형 데이터베이스의 특징은 데이터를 다수의 테이블에 분산하여 저장을 한다는 것입니다. 또한 분산되어 있는 자료들은 관계를 맺고 데이터를 검색 처리를 합니다.

MYSQL에서는 여러 개의 테이블을 결합하여 처리할 수 있는 UNION과 JOIN 명령 키워드를 지원 합니다.

**19.1 결합**

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

UNION 은 여러 개의 테이블 데이터를 묽어 하나의 결과로 출력합니다. 각각의 테이블에 대해서 SELECT결과를 통하여 서로 결합된 내용을 출력합니다.

* 두개 또는 여러개의 SELECT 결과를 결합니다.
* 결합되는 테이블의 컬럼의 자료형은 동일해야 합니다.
* 구조가 다른 테이블도 결과를 결합할 수 있습니다.
* 각각의 SELECT 문의 컬럼 또한 동일한 순서로 있어야합니다

**쿼리문법)**

|  |
| --- |
| SELECT 컬럼1, 컬럼2 ... FROM 테이블1  **UNION**  SELECT 컬럼1, 컬럼2 ... FROM 테이블2; |

두개의 SELECT 문장 사이에 UNION 키워드를 삽입을 하면 됩니다. 두개의 SELECT 결과를 모두 출력 하게 됩니다.

**19.1.2 쿼리 실습**

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

다음은 UNION 결합을 통하여 두개의 회원테이블을 하나의 데이터로 출력해 보는 실습 입니다. 아래 쿼리를 콘솔 모드에서 입력해 봅니다.

select

FirstName, LastName

from members

**union**

select

FirstName, LastName

from members1;

두개의 Select 쿼리 문장이 union 으로 결합되어 있습니다. 실행을 하면 다음과 같습니다.

|  |
| --- |
| mysql> **select FirstName, LastName from members union select FirstName, LastName from members1;**  +-----------+----------+  | FirstName | LastName |  +-----------+----------+  | lee | hojin |  | NULL | jiny |  | kim | james |  | jung | eric |  +-----------+----------+  4 rows in set (0.00 sec) |

위의 예제는 두개의 SELECT 내용이 결합되어 하나의 결과로 출력되었습니다.

만일 SELECT로 결합 출력을 하려고 하는 두개의 테이블의 컬럼의 수 가 동일 하다면 전체를 의미하는 와일드카드(\*) 기호를 이용을 할 수도 있습니다.

(select **\*** from members)

**union**

(select **\*** from members1);

여러 개의 union을 결합할 때 혼동을 방지하기 위하여 괄호를 같이 사용하면 편리합니다.

**19.2 조건식 결합**

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

UNION의 특징은 여러 개의 select 문장을 결합하여 출력을 합니다. 이때 작성되는 Select 문장에 추가 where 조건절을 포함하여 작성을 할 수 있습니다.

**쿼리문법)**

|  |
| --- |
| (select  FirstName, LastName  from members where 조건)  **union**  (select  FirstName, LastName  from members1 where 조건 );  **union**  (select  FirstName, LastName  from members3 where 조건 ); |

각각의 조건이 들어간 Select 결과들을 기준으로 union 결합을 하여 데이터를 출력을 할 수 있습니다.

**19.3 중복허용**

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

테이블의 데이터를 UNION으로 결합을 할 때 중복되는 데이터의 내용은 출력이 되지 않습니다. UNION은 기본적으로 중복 데이터를 허용하지 않는 형태로 필터링을 처리합니다.

UNION의 기본 중복 필터링을 허용하지 않고 전체 데이터를 출력 할 수도 있습니다. UNION ALL을 사용하게 되면 모든 전체의 데이터가 출력됩니다.

**쿼리문법)**

|  |
| --- |
| SELECT 컬럼1, 컬럼2 ... FROM 테이블1  **UNION ALL**  SELECT 컬럼1, 컬럼2 ... FROM 테이블2; |

**19.3.1 쿼리 실습**

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

앞에서 실습한 두 테이블의 결합의 예를 보면 각각의 테이블에는 중복된 데이터 테이블 입니다. 동일한 값이 있는데도 출력 결과가 2개가 표시되지 않고 한 개씩만 표시가 됩니다.

|  |
| --- |
| mysql> select FirstName, LastName from members union select FirstName, LastName from members1;  +-----------+----------+  | FirstName | LastName |  +-----------+----------+  | lee | hojin |  | NULL | jiny |  | kim | james |  | jung | eric |  +-----------+----------+  4 rows in set (0.00 sec) |

즉, 중복되는 이름은 제외가 된 것입니다. 다음은 두개의 테이블의 중복 값을 포함한 모든 데이터를 출력합니다. 만일 중복된 내용을 모두 출력을 하기 위해서는 UNION 보다 UNION ALL 로 사용하시면 됩니다.

select FirstName, LastName from members

**union ALL**

select FirstName, LastName from members1;

UNION ALL을 적용하여 입력을 해보도록 합니다.

|  |
| --- |
| mysql> **select FirstName, LastName from members union ALL select FirstName, LastName from members1;**  +-----------+----------+  | FirstName | LastName |  +-----------+----------+  | lee | hojin |  | NULL | jiny |  | kim | james |  | jung | eric |  | lee | hojin |  | NULL | jiny |  | kim | james |  +-----------+----------+  7 rows in set (0.00 sec) |

두개의 테이블 모든 데이터가 결합되어 출력이 됩니다. UNION을 통하여 결과를 결합할 때의 출력은 첫번째 SELECT 의 컬럼의 순번형태로 출력됩니다.

**19.3.2 PHP 실습**

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

PHP 코드를 통하여 UNION 출력 실습을 해보도록 합니다. UNION 결과를 처리할 수 있는 메서드를 추가하여 실습해 봅니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| // 다수의 query를 union 연결 문장을 실행합니다.  // query 문장을 가진 배열을 입력받습니다.  public function union($queryARR)  {  for ($i=0;$i<count($queryARR);$i++) {  $queryString .= "(".$queryARR[$i].") union ";  }  $queryString .= ";";  $queryString = str\_replace("union ;","",$queryString);  if ($result = mysqli\_query($this->dbcon, $queryString)) {  $this->msgEcho("쿼리성공] ".$queryString);  $rowss = "";  $row\_cnt = mysqli\_num\_rows($result);  for ($i=0; $i<$row\_cnt; $i++) {  $rowss[$i] = mysqli\_fetch\_object($result);  }  $result->free();  return $rowss;  } else {  $this->msgEcho("Error] ".$queryString);  }  } |

**예제파일) union-01.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $queryARR = array(  'select \* from members',  'select \* from members\_all' );  if($rowss = $db->union($queryARR)){  echo "tables fields = ". count($rowss) . "<br>";    for($i=0;$i<count($rowss);$i++){  echo $i."=";  print\_r($rowss[$i]);  echo "<br>";  }  }  ?> |

출력화면)

mysql connected!

쿼리성공] (select \* from members) union (select \* from members\_all)

tables fields = 12

0=stdClass Object ( [Id] => 1 [LastName] => hojin [FirstName] => lee [Address] => shinchon [City] => seoul [Country] => Korea [manager] => infohojin [email] => hojin@jinyphp.com )

1=stdClass Object ( [Id] => 2 [LastName] => lee [FirstName] => hojin [Address] => [City] => [Country] => Korea [manager] => infohojin [email] => jiny@jinyphp.com )

2=stdClass Object ( [Id] => 3 [LastName] => james [FirstName] => kim [Address] => youngdun-po [City] => seoul [Country] => Korea [manager] => info [email] => james@jinyphp.com )

3=stdClass Object ( [Id] => 5 [LastName] => 1234 [FirstName] => jiny [Address] => [City] => [Country] => Korea [manager] => [email] => )

4=stdClass Object ( [Id] => 6 [LastName] => 1234 [FirstName] => jiny [Address] => [City] => [Country] => Korea [manager] => [email] => )

5=stdClass Object ( [Id] => 7 [LastName] => 123400 [FirstName] => jiny [Address] => [City] => [Country] => Korea [manager] => [email] => )

6=stdClass Object ( [Id] => 1 [LastName] => 0000-00-00 [FirstName] => 2017-06-02 00:00:00 [Address] => s [City] => seoul [Country] => Korea [manager] => infohojin [email] => hojin@jinyphp.com )

7=stdClass Object ( [Id] => 2 [LastName] => 0000-00-00 [FirstName] => 2017-06-02 00:00:00 [Address] => [City] => [Country] => Korea [manager] => infohojin [email] => jiny@jinyphp.com )

8=stdClass Object ( [Id] => 3 [LastName] => 0000-00-00 [FirstName] => 2017-06-02 00:00:00 [Address] => y [City] => seoul [Country] => Korea [manager] => info [email] => james@jinyphp.com )

9=stdClass Object ( [Id] => 5 [LastName] => 0000-00-00 [FirstName] => 2017-06-02 00:00:00 [Address] => [City] => [Country] => Korea [manager] => [email] => )

10=stdClass Object ( [Id] => 6 [LastName] => 0000-00-00 [FirstName] => 2017-06-02 00:00:00 [Address] => [City] => [Country] => Korea [manager] => [email] => )

11=stdClass Object ( [Id] => 7 [LastName] => 0000-00-00 [FirstName] => 2017-06-02 00:00:00 [Address] => [City] => [Country] => Korea [manager] => [email] => )