jinyPHP 2권 - 010. 데이터조회

**10. 데이터조회**

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

데이터베이스 시스템에서 가장 많이 사용되는 명령은 데이터를 조회 하는 것입니다. SELECT 쿼리 명령은 테이블에서 데이터를 읽어오는 기능을 처리합니다. 전체 데이터를 읽어 오거나 조건을 통하여 특정 데이터만 읽어 올 수 있습니다.

**10.1 컬럼조회**

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

SELECT 쿼리 명령은 테이블에서 데이터를 읽어와 출력합니다. 이때 읽어올 테이블 명과 컬럼명을 지정을 하여야 합니다.

SELECT 쿼리는 기본적으로 2개의 키워드 명령을 가지고 있습니다. 첫번째는 명령의 시작 키워드 select 입니다. 테이블을 선택하는 from 키워드로 기본 구성됩니다.

**쿼리문법)**

|  |
| --- |
| **SELECT** 컬럼명1, 컬럼명2, 컬럼명3, … **from** 테이블명; |

select 와 from 키워드 사이에 읽어오고자 하는 테이블의 컬럼명을 콤마(,)로 구분하여 나열합니다. from 키워드 다음에는 조회할 테이블 명을 입력하면 됩니다.

**10.1.1 쿼리 실습**

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

테이블에서 원하는 컬럼의 데이터를 읽어와 목록으로 출력을 합니다.

|  |
| --- |
| SELECT FirstName, LastName, Country from members; |

콘솔 출력화면)

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

위의 예제를 보면 선택한 컬럼만 선택하여 결과를 출력합니다.

**10.1.2 PHP 실습**

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

PHP 코드를 통하여 테이블의 데이터를 출력합니다. 관련 메서드를 추가하여 소스에 사용을합니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| public function select($tbname,$field,$where)  {  if ($tbname) {  $queryString = "SELECT ";    if ($field) {  $queryString .= "".$field . " FROM $tbname";  } else {  $queryString .= "\* FROM $tbname";  }  if ($where) {  $queryString .= " where ".$where.";";  } else {  $queryString .= ";";  }  if ($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);  }  } else {  $this->msgEcho("Error] 테이블 이름이 없습니다.");  }  } |

**예제파일) select-01.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $tbname = "members";  // 컬럼 지정조회  echo "===== 컬럼조회 =====<br>";  if ($rowss = $db->select($tbname,"FirstName, LastName")) {  echo "tables fields = ". count($rowss) . "<br>";    for ($i=0;$i<count($rowss);$i++) {  echo $i."=";  print\_r($rowss[$i]);  echo "<br>";  }  }  // 컬럼, 조건 지정조회  echo "===== 컬럼조회 =====<br>";  if ($rowss = $db->select($tbname,"FirstName, LastName","Id<=3")) {  echo "tables fields = ". count($rowss) . "<br>";    for ($i=0;$i<count($rowss);$i++) {  echo $i."=";  print\_r($rowss[$i]);  echo "<br>";  }  }  ?> |

출력화면)

mysql connected!

===== 컬럼조회 =====

쿼리성공] SELECT FirstName, LastName FROM members;

tables fields = 6

0=stdClass Object ( [FirstName] => lee [LastName] => hojin )

1=stdClass Object ( [FirstName] => [LastName] => jiny )

2=stdClass Object ( [FirstName] => kim [LastName] => james )

3=stdClass Object ( [FirstName] => jiny [LastName] => 1234 )

4=stdClass Object ( [FirstName] => jiny [LastName] => 1234 )

5=stdClass Object ( [FirstName] => jiny [LastName] => 123400 )

===== 컬럼조회 =====

쿼리성공] SELECT FirstName, LastName FROM members where Id<=3;

tables fields = 3

0=stdClass Object ( [FirstName] => lee [LastName] => hojin )

1=stdClass Object ( [FirstName] => [LastName] => jiny )

2=stdClass Object ( [FirstName] => kim [LastName] => james )

**10.1.3 컬럼순서**

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

선택적으로 컬럼을 출력조회는 출려하고자 하는 컬럼의 순서를 변경할 수도 있습니다. SQL 쿼리상에서 지정한 컬럼명 순서대로 출력이 됩니다.

**10.2 전체조회**

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

테이블의 컬럼 전체를 출력하고자 할 때 컬럼명 모두를 나열하는 것은 불편한 작성일 것입니다. 만일 모든 컬럼을 출력할 때 쉽게 처리할 수 있는 와일드 기호를 지원합니다.

\* 와일드 기호는 일반적으로 모두 선택을 의미합니다. SELECT 쿼리에서도 컬럼명 대신에 \* 기호를 사용하게 되면 모든 컬럼을 선택하는 효과를 보실 수 있습니다.

**쿼리문법)**

|  |
| --- |
| SELECT \* from 테이블명; |

**10.2.1 쿼리 실습**

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

컬럼을 하나씩 선택하는 방법 이외에도 컬럼 전체 데이터를 쉽게 출력할 수 있습니다.

|  |
| --- |
| SELECT \* from members; |

다수의 컬럼명을 와일드 기호 \* 로 대체하여 모든 컬럼을 선택할 수 있습니다.

|  |
| --- |
| mysql> **SELECT \* from members;**  +----+----------+-----------+-------------+-------+---------+  | 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) |

위의 예제를 보면 와일드 기호 \* 로 모든 컬럼을 선택하여 데이터 결과를 출력합니다.

**10.2.2 PHP 실습**

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

PHP 코드를 이용하여 전체 컬럼 데이터 목록을 출력합니다.

**예제파일) select-02.php**

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

출력화면)

mysql connected!

===== 전체조회 =====

쿼리성공] SELECT \* FROM members;

tables fields = 6

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] => jiny [FirstName] => [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] => [manager] => [email] => )

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

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

**10.3 지정값 출력**

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

지금까지의 SELECT는 테이블의 컬럼 데이터 값을 읽어와서 출력을 하였습니다. SELECT는 컬럼 데이터 이외에 직접 정의한 상수도 출력할 수 있습니다.

**쿼리문법)**

|  |
| --- |
| SELECT 상수값, 상수값 ….; |

**10.3.1 쿼리 실습**

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

SELECT 쿼리 명령어 다음에 단일 따옴표를 통하여 상수값을 입력합니다.

|  |
| --- |
| select 'aaa','bbb','ccc'; |

상수값이 여러 개일 경우 콤마로 구분을 합니다.

|  |
| --- |
| mysql> select 'aaa','bbb','ccc';  +-----+-----+-----+  | aaa | bbb | ccc |  +-----+-----+-----+  | aaa | bbb | ccc |  +-----+-----+-----+  1 row in set (0.00 sec) |

위의 출력결과를 보면 SELECT 다음에 입력한 상수값을 그냥 출력합니다. 또한 간단한 연산식도 입력할 수 있습니다.

|  |
| --- |
| mysql> select 1+1;  +-----+  | 1+1 |  +-----+  | 2 |  +-----+  1 row in set (0.00 sec) |

연산식은 간단한 계산이나 함수값을 출력할때 사용하면 편리합니다.

**05.2 \* 와일드카드**

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

아스테리스크 (\*) 와일드 카드는 앞의 전체 컬럼을 출력할 때 한번 사용을 해보았습니다. \* 기호는 전체를 대신하는 기호 입니다. SELECT 컬럼명을 나열하는 \* 와일드 카드를 적용하면, 전체의 컬럼 지정을 의미합니다.

**쿼리문법)**

|  |
| --- |
| SELECT \* FROM 테이블명; |

**쿼리예제)**

|  |
| --- |
| SELECT \* FROM members; |

다음 select 쿼리는 회원 테이블에서 전체의 컬럼필드의 정보를 가지고 옵니다.

**05.3 중복제거**

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

SELECT 쿼리는 테이블의 모든 데이터를 가지고 와서 출력을 합니다. 테이블에 여러 임의의 데이터들이 무작위로 들어가다 보면 **중복된 데이터**들이 발생하게 됩니다.

이때 DISTINCT 키워드는 SELECT 쿼리를 통하여 데이터를 출력할 때 중복된 데이터를 배제하고 출력을 할 수 있습니다. SELECT DISTINCT 쿼리명령은 데이터에서 중복된 값은 배제하고 결과를 출력합니다.

**쿼리문법)**

|  |
| --- |
| SELECT DISTINCT 컬럼1, 컬럼2, … FROM 테이블명; |

**05.3.1 쿼리 실습**

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

먼저 아래 쿼리는 고객명단에서 국가 컬럼필드 만 출력합니다. 하지만, 각 나라별 회원수는 아주 많을 것입니다.

|  |
| --- |
| SELECT Country FROM members; |

그런데 회원정보를 기준으로 가입된 국가의 정보를 알고 싶다고 해봅시다. 국가의 목록들은 회원들로 인하여 중복될 것입니다. 중복된 국가를 제외하고, 1개의 국가들만 출력하고 싶습니다.

|  |
| --- |
| SELECT DISTINCT Country FROM members; |

DISTINCT 를 사용하면 중복된 내용을 배제하기 때문에 회원수의 국가목록을 하나씩만 출력합니다.

DISTINCT 키워드는 SQL 내장함수의 인자로도 넣을 수 있습니다.

|  |
| --- |
| SELECT COUNT(DISTINCT Country) FROM members; |

위의 예제쿼리는 가입된 회원의 국가명의 갯수를 확인할 수 있습니다.

**05.3.2 PHP실습**

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

PHP 코드를 통하여 중복을 배제한 목록을 출력해 봅니다.

**PHP 예제)**

mysql.class.php 파일에 메서드 예제를 추가합니다.

|  |
| --- |
| public function selectDistinct($tbname,$field,$where)  {  if ($tbname) {  $queryString = "SELECT DISTINCT ";    if ($field) {  $queryString .= "".$field . " FROM $tbname";  } else {  $queryString .= "\* FROM $tbname";  }  if ($where) {  $queryString .= " where ".$where.";";  } else {  $queryString .= ";";  }  if ($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);  }  } else {  $this->msgEcho("Error] 테이블 이름이 없습니다.");  }  } |

**예제파일) select-04.php**

|  |
| --- |
| <?php  include "dbinfo.php";  include "mysql.class.php";    // ++ Mysqli DB 연결.  $db = new JinyMysql();  $tbname = "members";  // 컬럼, 조건 지정조회  echo "===== 컬럼조회 =====<br>";  if ($rowss = $db->selectDistinct($tbname,"FirstName, LastName","Id<=3")) {  echo "tables fields = ". count($rowss) . "<br>";    for ($i=0;$i<count($rowss);$i++) {  echo $i."=";  print\_r($rowss[$i]);  echo "<br>";  }  }  ?> |

출력화면)

mysql connected!

===== 컬럼조회 =====

쿼리성공] SELECT DISTINCT FirstName, LastName FROM members where Id<=3;

tables fields = 3

0=stdClass Object ( [FirstName] => lee [LastName] => hojin )

1=stdClass Object ( [FirstName] => [LastName] => jiny )

2=stdClass Object ( [FirstName] => kim [LastName] => james )