

# 07

## CSV

CSV 파일은 엑셀 등 데이터베이스의 자료들을 서로 주고받을 때 자주 사용하는 순수 데이터 포맷 파일의 일종입니다. CSV는 데이터를 외부 파일로 출력하고, 외부 데이터를 일괄 처리하여 입력받을 때 많이 사용합니다.

### 07.1 샘플 데이터

엑셀 프로그램 등을 통하여 작성한 데이터를 CVS 포맷으로 저장하면 됩니다. 파일 저장 형식을 선택하면 다양한 파일 형식을 볼 수 있습니다.

```
Excel 통합 문서
Excel 매크로 사용 통합 문서
Excel 바이너리 통합 문서
Excel 97 - 2003 통합 문서
CSV UTF-8(실표로 분리)
XML 데이터
웹 보관 파일
웹 페이지
Excel 서식 파일
Excel 매크로 사용 서식 파일
Excel 97 - 2003 서식 파일
텍스트 (탭으로 분리)
유니코드 텍스트
XML 스프레드시트 2003
Microsoft Excel 5.0/95 통합 문서
CSV (실표로 분리)
텍스트 (공백으로 분리)
DIF (Data Interchange Format)
```

CSV 파일로 엑셀 파일을 저장할 때 각각의 데이터는 콤마(,)로 구분하여 한 줄에 1개의 레코드 데이터를 저장합니다.

## 07.2 CSV 쓰기

PHP에서 생성한 데이터를 CSV 형태의 파일로 출력할 수 있습니다. 출력된 CSV 파일은 엑셀 등의 응용프로그램에서 읽고, 데이터를 가공할 수 있습니다.

fputcsv() 함수는 입력된 Array 배열을 CSV 형식으로 줄을 만들고 파일 포인터에 씁니다.

### | 내장 함수 |

```
int fputcsv ( resource $handle , array $fields [, string $delimiter = "," [, string $enclosure = "" [, string $escape_char = "\" ]]] )
```

fputcsv()는 필드 배열로 전달된 행을 CSV로 형식으로 지정합니다. 지정된 파일 핸들에 새로운 행으로 끝내도록 출력합니다.

#### 예제 파일 | csv\_write.php

```
1  <?php
2
3      $list = array (
4          array(1, "aaa", 18, "Seoul"),
5          array(2, "bbb", 20, "Daejeon"),
6          array(3, "ccc", 30, "Incheon ")
7      );
8
9      $csvFile = "sample.csv";
10
11     $fp = fopen($csvFile, 'w');
12     if (!is_resource($fp)) {
13         die("저장할 파일 포인터를 생성할 수 없습니다.");
14     } else {
```

```

15
16         // 파일 독점 잠금 설정
17         flock($fp, LOCK_EX);
18
19         // 배열을 CSV 파일로 저장
20         foreach ($list as $fields) {
21             fputcsv($fp, $fields);
22         }
23
24         // 파일 잠금 해제
25         fflush($fp);
26         flock($fp, LOCK_UN);
27
28         fclose($fp);
29     }
30
31     ?>

```

지정한 Array 데이터를 sample.csv 파일로 출력을 합니다.

#### 파일 내용

```

1,aaa,18,Seoul
2,bbb,20,Daejeon
3,ccc,30,"Incheon "

```

## 07.3 CSV 읽기

PHP는 CVS 파일의 데이터를 쉽게 읽고 처리하기 위해서 fgetcsv() 함수를 제공합니다. fgetcsv() 함수는 파일 포인터에서 한 줄 단위로 데이터를 읽어와 \$delimiter 기호를 기준으로 CSV 필드를 구문 분석합니다.

#### | 내장 함수 |

```

array fgetcsv ( resource $handle [, int $length = 0 [, string $delimiter = "," [, string $enclosure = "" [, string $escape = "\" ]]] ] )

```

fgetcsv()가 CSV 형식의 필드에 대해 읽은 행을 구문 분석하고 읽은 필드가 들어 있습니다. 배열을 반환한다는 점을 제외하면 fgets()와 비슷한 동작을 수행합니다.

#### 예제 파일 | csv\_read.php.php

```
1  <?php
2      $csvFile = "sample.csv";
3
4      // 먼저 CSV 파일 존재 유무를 확인합니다.
5      if (!file_exists($csvFile)) {
6          die("CSV 파일이 존재하지 않습니다.");
7      } else {
8
9          // CSV 파일을 읽어옵니다.
10         $row = 1;
11         if (($fp = fopen($csvFile, "r")) !== FALSE) {
12             // CSV 데이터 한 줄을 읽어옵니다.
13             $length = 1000;
14             $delimiter = ",";
15             while (($data = fgetcsv($fp, $length, $delimiter)) !==
16                 FALSE) {
17                 $columns = count($data);
18
19                 echo "라인($row): 컬럼수($columns)<br/>";
20                 $row++;
21                 for ($i=0; $i < $columns; $i++) {
22                     echo $data[$i] . "<br />";
23                 }
24             }
25             fclose($fp);
26         }
27     }
28
29  ?>
```

#### 화면 출력

라인(1): 컬럼수(4)  
1  
aaa

```
18
Seoul
라인(2): 컬럼수(4)
2
bbb
20
Daejeon
라인(3): 컬럼수(4)
3
ccc
30
Incheon
```

### | 내장 함수 |

```
array str_getcsv ( string $input [, string $delimiter = "," [, string $enclosure = "" [,
string $escape = "\\" ]]] )
```

내장 함수 `str_getcsv()`는 CSV 문자열을 배열로 구문 분석합니다.

#### 예제 파일 | `str_getcsv.php`

```
1 <?php
2     $csv = array_map('str_getcsv', file('sample.csv'));
3     print_r($csv);
4
5 ?>
```

#### 화면 출력

```
Array ( [0] => Array ( [0] => 1 [1] => aaa [2] => 18 [3] => Seoul ) [1] => Array (
[0] => 2 [1] => bbb [2] => 20 [3] => Daejeon ) [2] => Array ( [0] => 3 [1] =>
ccc [2] => 30 [3] => Incheon ) )
```

