

## 포인터 배열과 배열 포인터의 구분

앞선 예제의 배열 포인터에서 괄호(())를 생략하면 전혀 다른 의미가 됩니다.

### 예제

```
1. int (*pArr)[3];  
2. int* pArr[3];
```

위의 예제 중에서 첫 번째는 int형 데이터를 저장할 수 있는 2차원 배열을 가리키는 배열 포인터입니다.

하지만 두 번째는 int형 데이터를 가리킬 수 있는 포인터 변수를 모아 놓은 배열을 가리키는 포인터 배열이 됩니다.

따라서 괄호의 유무가 중요하며, 포인터 배열과 배열 포인터 사이의 차이점을 정확히 이해하고 넘어가야 합니다.

## 포인터 배열

포인터 배열이란 배열 요소로 포인터 변수를 가지는 배열을 의미합니다.

즉, 포인터 변수를 저장할 수 있는 배열을 의미합니다.

다음 예제는 세 개의 int형 포인터 변수를 저장할 수 있는 포인터 배열을 선언하는 예제입니다.

### 예제

```
int i, arr_len;  
int num01 = 10, num02 = 20, num03 = 30;  
int* arr[3] = {&num01, &num02, &num03}; // int형 포인터 배열 선언  
  
arr_len = sizeof(arr)/sizeof(arr[0]);  
for (i = 0; i < arr_len; i++)  
{  
    printf("%d\n", *arr[i]);  
}
```

코딩연습 ▶

### 실행 결과

```
10  
20  
30
```

다음 그림은 위의 예제에서 사용된 포인터 배열이 메모리 상에서 어떻게 동작하는지를 보여줍니다.

