

```
1  struct 구조체의 이름 {
2      멤버 변수;
3  };

1  #include <stdio.h>
2
3  struct student {
4      int id;
5      char *name;
6      float percentage;
7  }; // 구조체 뒤에 세미콜론이 와야함
8
9  int main() {
10     struct student s;
11     s.id=1;
12     s.name = "김철수";
13     s.percentage = 90.5;
14     printf("아이디: %d \n", s.id);
15     printf("이름: %s \n", s.name);
16     printf("백분율: %f \n", s.percentage);
17     return 0;
18 }
```

```
1  /* www.roseindia.net/c-tutorials/c-structure-pointer.shtml */
2  #include <stdio.h>
3
4  int main() {
5      struct st {
6          int id;
7          char *name;
8          char *address;
9      };
10     struct st employee, *stptr;
11     stptr = &employee;
12     stptr->id = 1;
13     stptr->name = "홍길동";
14     stptr->address = "서울";
15     printf("직원 안내: 아이디=%d\n%s\n%s\n", stptr->id, stptr->name,
16     stptr->address);
17     return 0;
18 }
```

<const가 (*) 왼쪽에 있는 경우>

: **const** char *pText;

: char **const** *pText;

: pText 포인터가 가리키고 있는 데이터에게 상수성을 부여
: 즉, 포인터 자체는 변경가능.

const char* pText = MainText; // const가 * 왼쪽에 있으므로 데이터 변경 불가

const char* pSubsText = "BBB";

pText = pSubsText; // 주소 변경이므로 OK

pText = "ByeBye"; // 역시, 주소 변경이므로 OK

*pText = 'X'; // 데이터 변경불가이므로 Error

<const가 (*) 오른쪽에 있는 경우>

: char* **const** pText;

: pText 포인터 자체에 상수성을 부여

: 즉, 데이터 변경 가능

char* const pText = MainText; // const가 * 오른쪽에 있으므로 포인터 변경 불가

pText = "ByeBye"; // 포인터 변경불가이므로 Error

*pText = 'X'; // 데이터 변경이므로 OK