12장 문자열

혼자 공부하는 C 은 혼자 공부하는 C 은 혼자 공부하는 C 은 혼자 공부하는 C 은 혼자 공부하는 C 은



❖ 문자열 상수 구현 방법

문자열 상수가 주소란 증거 소스 코드 예제12-1.c

```
01 #include <stdio.h>
                                                         ₩ 실행결과
                                                                                      X
                                                         apple이 저장된 시작 주소 값: 00EA58F8
02
                                                         두 번째 문자의 주소 값: 00EA58F9
   int main(void)
                                                         첫 번째 문자 : a
04 {
                                                         두 번째 문자 : p
                                                         배열로 표현한 세 번째 문자 : p
05
       printf("apple이 저장된 시작 주소 값: %p\n", "apple");
       printf("두 번째 문자의 주소 값: %p\n", "apple" + 1);
                                                            // 주소 값 출력
06
07
       printf("첫 번째 문자 : %c\n", *"apple");
                                                             // 간접 참조 연산
       printf("두 번째 문자 : %c\n", *("apple" + 1));
08
                                                          // 포인터 연산식
       printf("배열로 표현한 세 번째 문자 : %c\n", "apple"[2]); // 배열 표현식
09
10
                                                                 문자열이 저장된 시작 위치 값
                                    printf("주소 값: %p\n", "apple");
                                                                 (끝의 두 자리만 표시)
11
       return 0;
                                          문자열은 따로 배열에 저장
12 }
                                                                    (F8) F9
                                                                          FA FB
                                                                                 FC
                                                                                    FD
                                                                                    10
                                                                     a
                                                                        p
                                                                           p
                                       첫 번째 문자의 주소 전달
                                    printf("주소 값: %p\n", F8);
```

❖ char 포인터로 문자열 사용

포인터로 문자열을 사용하는 방법 소스 코드 예제12-2.c

```
01 #include <stdio.h>
                                           첫 번째 문자의 주소 값 저장
02
                                                          100 2 101
                                                               102 103 104 105
   int main(void)
                                        100
                                                                         10
04 {
                                   dessert 포인터
                                                                문자열 상수
       char *dessert = "apple";
05
                                              첫 번째 문자를 가리킨다.
06
       printf("오늘 후식은 %s입니다.\n", dessert); // 문자열 출력
07
08
       dessert = "banana";
                                              // 새로운 문자열 대입
       printf("내일 후식은 %s입니다.\n", dessert); // 바뀐 문자열 출력
09
10
                                              ₩ 실행결과
11
       return 0;
                                               오늘 후식은 apple입니다.
12 }
                                               내일 후식은 banana입니다.
```

❖ scanf 함수를 사용한 문자열 입력

scanf 함수를 사용한 문자열 입력 소스 코드 예제12-3.c

```
01 #include <stdio.h>
                                 키보드 입력
                                                                버퍼
02
                                apple jam
03
   int main(void)
                              Enter 를 누를 때
04
   {
                              개행 문자까지 버퍼에 저장
                                                          곳백 문자 전까지의 문자열을
                                                          배열에 저장하고 널 문자를 붙입니다.
05
       char str[80];
                                          str 배열
06
                                                                 10
                                                              е
       printf("문자열 입력 : ");
07
       scanf("%s", str);
                                              // %s를 사용하고 배열명을 준다.
08
09
       printf("첫 번째 단어 : %s\n", str);
                                              // 배열에 입력된 문자열 출력
       scanf("%s", str);
10
11
       printf("버퍼에 남아 있는 두 번째 단어 : %s\n", str);
                                                       까 실행결과
                                                                                   X
12
                                                        문자열 입력 : apple jam 🕗
       return 0;
13
                                                        첫 번째 단어 : apple
14 }
                                                        버퍼에 남아 있는 두 번째 단어 : jam
```

❖ gets 함수를 사용한 문자열 입력

gets 함수로 한 줄의 문자열 입력 소스 코드 예제12-4.c

```
01
   #include <stdio.h>
                              키보드 입력
                                                          버퍼
                                                p
                              apple jam 🕗
                                                   p
                                                                   m
02
                            Enter 를 누를 때
   int main(void)
                            개행 문자까지 버퍼에 저장
                                                            모두 배열에 저장하지만
04
       char str[80];
05
                                       str 배열
                                                  p
                                             a
                                               p
                                                                   m
06
       printf("공백이 포함된 문자열 입력 : ");
07
08
       gets(str);
                              // 배열명으로 주고 함수 호출
       printf("입력한 문자열은 %s입니다.", str);
09
10
                                      ₩ 실행결과
                                                                          X
11
       return 0;
                                       공백이 포함된 문자열 입력 : apple jam 🕗
12
                                       입력한 문자열은 apple jam입니다.
```

❖ fgets 함수를 사용한 문자열 입력

fgets 함수의 문자열 입력 방법 소스 코드 예제12-5.c

```
#include <stdio.h>
                                 키보드 입력
                                                                  버퍼
   // 나중에 입력할 공간입니다.
                                                  a
                                                     p
                                                        p
                                                                        a
                                                                           m
                                                                              \n
                                apple jam 🕗
03
                              Enter 를 누를 때
                              개행 문자까지 버퍼에 저장
   int main(void)
                                                                   개행 문자까지 배열에 저장하고
                                                                   마지막에 널 문자를 붙입니다.
05
   {
06
       char str[80];
                                           str 배열
                                                        p
                                                                              \n
                                                              е
                                                                        a
                                                                           m
07
08
       printf("공백이 포함된 문자열 입력 : ");
                                                        str[strlen(str) - 1] = '\0'
       fgets(str, sizeof(str), stdin);
09
                                                        널 문자 이전까지 문자 수 Io개
10
       // 나중에 입력할 공간입니다.
                                                   9는 배열에서 개행 문자가 저장된 곳의 위치
11
       printf("입력된 문자열은 %s입니다\n", str);
                                                            개행 문자를 널 문자로 바꿈
12
                            7 실행결과
13
       return 0;
                             공백이 포함된 문자열 입력 : apple jam ❷
                             입력된 문자열은 apple jam
14 }
                             입니다
```

❖ 표준 입력 함수의 버퍼 공유 문제

개행 문자로 인해 gets 함수가 입력을 못하는 경우 소스 코드 예제12-6.c

```
01 #include <stdio.h>
                                      키보드 입력
                                                              버퍼
02
                                        17
   int main(void)
                                   Enter 를 누를 때
04 {
                                   개행 문자까지 버퍼에 저장
                                                                 gets
                                                          scanf
05
       int age;
       char name[20];
                                                                         10
                                                                                       name 배열
06
                                 age 변수
                                             17
                                               문자열 "17"을 정수 17로
07
                                               변환하여 2진수로 저장
                                                                                      getchar();
       printf("나이 입력: ");
08
09
       scanf("%d", &age);
                                       // scanf 함수로 나이 입력
                                                                                      scanf("%*c");
10
       printf("이름 입력 : ");
11
                                                                                      fgetc(stdin);
12
       gets(name);
                                       // gets 함수로 이름 입력
       printf("나이 : %d, 이름 : %s\n", age, name);
13
                                                          ₩ 실행결과
                                                                                     X
14
                                                          나이 입력 : 17 🔊
15
       return 0;
                                                          이름 입력 : 나이 : 17, 이름 :
16 }
```

❖ 문자열을 출력하는 puts, fputs 함수

문자열을 출력하는 puts와 fputs 함수 소스 코드 예제12-7.c

```
#include <stdio.h>
01
                                    배열명str
02
   int main(void)
                                                          ₹200₹ 201 202
   {
                                                                           10
04
                                               200
                                    포인터ps
                                                                     n
                                                                        a
05
       char str[80] = "apple juice";
                                           ₹300₹ 301 302
06
       char *ps = "banana";
                             문자열 상수
07
       puts(str); // apple juice 출력하고 줄 바꿈
08
09
       fputs(ps, stdout);// banana만 출력
10
       puts("milk"); // banana에 이어 milk 출력
                                                     ₩ 실행결과
                                                                             ×
11
                                                      apple juice
       return 0;
12
                                                      bananamilk
13 }
```



키워드로 끝내는 핵심 포인트

- **❖ 문자열**은 첫 번째 문자가 저장된 메모리의 주소로 바뀐다.
- ❖ scanf 함수는 중간에 공백이 포함된 문자열을 입력할 수 없다.
- ❖ gets 함수는 한 줄의 데이터를 char 배열에 저장한다.
- ❖ fgets 함수는 배열의 크기를 검사하는 문자열 입력 함수다.

마무리

표로 정리하는 핵심 포인트

표 12-1 문자열을 저장하는 배열과 포인터의 차이

구분	char 포인터	char 배열
초기 <mark>화</mark>	<pre>char *pc = "mango";</pre>	char str[80] = "mango";
대입	pc = "banana";	<pre>strcpy(str, "banana");</pre>
크기	sizeof(pc) → 4바이트	sizeof(str) → 80H이트
수정	pc[0] = 't'; (X)	str[0] = 't'; (0)
입력	scanf("%s", pc); (X)	scanf("%s", str); (0)

❖ 문자열을 대입하는 strcpy 함수 (1/2)

strcpy 함수의 사용법 소스 코드 예제12-8.c

```
#include <stdio.h>
   #include <string.h>
                                      str1
                                                           W
                                                               b
                                                       a
                                                                  е
03
   int main(void)
05
   {
                                       포인터 ps2
                                                 100
                                                       str2
06
       char str1[80] = "strawberry";
       char str2[80] = "apple";
07
                                                               200
                                                                  201 202
08
       char *ps1 = "banana";
                                             포인터 psl
                                                      200
                                                                                     10
                                                                                  a
09
       char *ps2 = str2;
                                                                        문자열 상수
10
       printf("최초 문자열: %s\n", str1);
11
                                                           strcpy(str1, str2)
12
       strcpy(str1, str2);
13
       printf("바뀐 문자열: %s\n", str1);
14
                                                             복사 받을 곳 복사할 내용
```

❖ 문자열을 대입하는 strcpy 함수 (2/2)

strcpy 함수의 사용법 소스 코드 예제12-8.c

```
strcpy(str1, ps1);
15
                                                                   10
        printf("바뀐 문자열 : %s\n", str1); <sup>str1</sup>
                                                    p
                                                        p
                                                               е
                                                                       е
16
17
                                                                     201
                                                                         202
                                                                  200
18
        strcpy(str1, ps2);
                                               포인터 psl
                                                         200
                                                                                         10
                                                                   b
                                                                      a
                                                                              a
                                                                                     a
                                                                          n
                                                                                 n
19
        printf("바뀐 문자열: %s\n", str1);
                                                                           문자열 상수
20
21
        strcpy(str1, "banana");
                                               포인터 ps2
                                                        100
                                                              str2
                                                                                 е
        printf("바뀐 문자열: %s\n", str1);
22
23
24
        return 0;
                                                                  ₩ 실행결과
                                                                                         ×
25 }
                                                                  최초 문자열 : strawberry
                                                                  바뀐 문자열 : apple
                                                                  바뀐 문자열 : banana
                                                                  바뀐 문자열 : apple
                                                                  바뀐 문자열 : banana
```

❖ 원하는 개수의 문자만을 복사하는 strncpy 함수

strncpy 함수를 사용한 문자열 복사 소스 코드 예제12-9.c

```
01 #include <stdio.h>
                                        strncpy(str, "apple-pie", 5);
   #include <string.h>
03
                                              복사 받을 복사할 복사할
배열명 문자열 문자 수
   int main(void)
05 {
06
       char str[20] = "mango tree"; // 배열 초기화
07
       strncpy(str, "apple-pie", 5); // "apple-pie"에서 다섯 문자만 복사
08
09
       printf("%s\n", str);
10
                                   // 복사 받은 문자열 출력
11

    ✓ 실행결과

                                                                          X
12
       return 0;
                                                         apple tree
13 }
```

❖ 문자열을 붙이는 strcat, strncat 함수

strcat, strncat 함수를 사용한 문자열 붙이기 소스 코드 예제12-10.c

```
01
    #include <stdio.h>
02
    #include <string.h>
                                    a
                                       W
                                           b
03
                                             strncat 함수
                                                                     마지막에 널 문자를 붙임
    int main(void)
04
05
    {
06
        char str[80] = "straw";
                                     // 문자열 초기화
07
08
        strcat(str, "berry");
                                     // str 배열에 문자열 붙이기
09
        printf("%s\n", str);
10
        strncat(str, "piece", 3); // str 배열에 3개의 문자 붙이기
11
        printf("%s\n", str);
                                                          ₩ 실행결과
                                                                                 ×
12
                                                           strawberry
        return 0;
13
                                                           strawberrypie
14 }
```

❖ 문자열 길이를 계산하는 strlen 함수

두 문자열 중 길이가 긴 단어 출력 소스 코드 예제12-11.c

```
01 #include <stdio.h>
02 #include <string.h>
                                                          strlen(str)
03
   int main(void)
                                                                       - 크기를 확인할 배열명
05 {
      char str1[80], str2[80];
                                       // 두 문자열을 입력할 배열
06
      char *resp;
                                         // 문자열이 긴 배열을 선택할 포인터
07
08
09
      printf("2개의 과일 이름 입력: ");
10
      scanf("%s%s", str1, str2);
                                  // 2개의 문자열 입력
      if (strlen(str1) > strlen(str2)) // 배열에 입력된 문자열의 길이 비교
11
12
          resp = str1;
                                        // 첫 번째 배열이 긴 경우 선택
13
      else
14
          resp = str2;
                                        // 두 번째 배열이 긴 경우 선택
15
      printf("이름이 긴 과일은 : %s\n", resp); // 선택된 배열의 문자열 출력
16
                                                        5 실행결과
                                                                                           X
      return 0;
17
                                                        2개의 과일 이름 입력 : banana strawberry 🔊
18 }
                                                        이름이 긴 과일은 : strawberry
```

❖ 문자열을 비교하는 strcmp, strncmp 함수

strcmp, strncmp 함수를 사용한 문자열 비교 소스 코드 예제12-12.c

```
사전의 뒤에 나오는 문자열
01 #include <stdio.h>
                                                           10
                                                                                        10
                                                        е
   #include <string.h>
                                               →아스키 코드 값 97
                                                                         →아스키 코드 값 98
03
                                                  사전의 뒤에 나오는 문자열
   int main(void)
                                                                                     h \0
                                                                            е
                                                                         p
                                                                               a
                                                                                   C
05
  {
       char str1[80] = "pear";
06
                                                       아스키 코드 값 114
                                                                              아스키 코드 값 99
                                                                          사전의 뒤에 나오는 문자열
       char str2[80] = "peach";
07
08
       printf("사전에 나중에 나오는 과일 이름 : ");
09
                                                                              아스키 코드 값 108
                                                      아스키코드 값 0
       if (strcmp(str1, str2) > 0) // str1이 str2보다 사전에 나중에 나오면 1 반환
10
11
           printf("%s\n", str1); // str1이 str2보다 사전에 먼저 나오면 -1 반환
12
       else
                                     // str1과 str2가 같은 문자열이면 0 반환
           printf("%s\n", str2);
13
14
                                                     ₩ 실행결과
                                                                                        X
       return 0;
15
                                                     사전에 나중에 나오는 과일 이름 : pear
16 }
```



키워드로 끝내는 핵심 포인트

- ❖ strcpy 함수에서 문자열을 복사 받는 곳은 배열이어야 한다.
- ❖ strcat 함수로 문자열을 최초로 붙일 때는 초기화를 해야 한다.
- ❖ strlen 함수로 배열에 저장된 문자열의 길이를 알 수 있다.
- ❖ strcmp 함수로 문자열의 사전 등록 순서를 확인할 수 있다.

마무리

표로 정리하는 핵심 포인트

표 12-2 기본적인 문자열 연산 함수

연산 가능	사용 방법	실행결과
대입	strcpy(str1, str2);	문자열 str2를 str1에 복사
길이 계산	strlen(str);	문자열 str의 길이(문자 수)를 구해 반환
붙이기	strcat(str1, str2);	문자열 str2를 str1문자열 뒤에 이어 붙임
비교	strcmp(str1, str2);	문자열 str1이 str2보다 크면 1 반환 문자열 str1이 str2보다 작으면 -1 반환 str1과 str2가 같은 문자열이면 0 반환