

THE  
**C**  
PROGRAMMING  
LANGUAGE

# 12차시 문자/문자열

# 문자단위 입출력 함수

- 문자 입력 함수 : `int getchar()`
- 문자 출력 함수 : `int putchar(int c)`
  
- `int c1;`  
`c1 = getchar();`  
`putchar(c1);`

# 문자열의 저장

- 배열을 이용하여 문자열을 저장한다
- 마지막에 반드시 널(\0)이 삽입된다.
- `char s[6] = "hello";`  
또는 `char s[] = "hello";` //배열길이 자동 계산

s[0]	s[1]	s[2]	s[3]	s[4]	s[5]	
h	e	l	l	o	\0	

널(null)문  
자

# 문자열의 사용자 입력

➤ %s 서식문자를 이용하여 입력 받는다

➤ `char s[6];`  
`scanf("%s", s);` //예) 12345 입력  
`printf("%s \n", s);`

& 붙이지 않는다

s[0]	s[1]	s[2]	s[3]	s[4]	s[5]	
1	2	3	4	5	\0	

널(null)문자  
자동삽입

# 문자열 입출력 함수

- 문자 입력 함수 : `char* gets(char* s)`
- 문자 출력 함수 : `int puts(const char* s)`
  
- `char name[20];`  
`gets(name);`  
`puts(name);` //자동 줄바꿈

# 그외 문자열 관련 함수

- <string.h>
- 문자열 복사 : `char * strcpy(*to_str, *from_str2)`
- 문자열 연결 : `char * strcat(*str1 , *str2);`
- 문자열 비교 : `char * strcmp(*str1, *str2);`
- 문자열 길이 : `int strlen(*str);`
- <stdlib.h>
- 문자열을 int로 변환 : `int atoi(*str);`
- 문자열을 double로 변환 : `double atof(*str);`
- 등등
- <stdio.h>
- `sprintf`함수를 이용해 다양한 형 변환가능

# 실습

- 문자열을 입력 받고 그 문자열의 값과 길이와 주소값을 출력해 보시오
- 예)  
문자열을 입력하세요 : **call by reference**  
값: call by reference  
길이 : 17  
주소값 : 13630056