<자주 쓰는 용어>

변수 : 숫자, 문자와 같은 값들을 저장하는 공간

자료형 : 변수의 종류스크린샷이(가) 표시된 사진

자동 생성된 설명

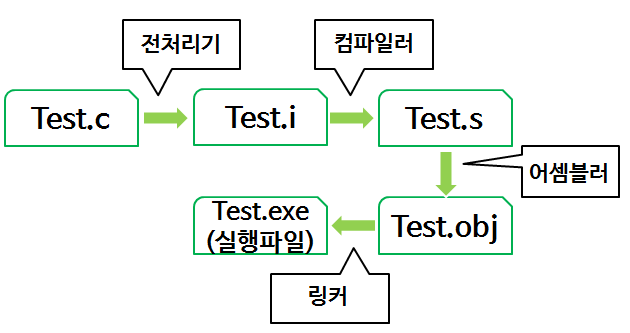
char (character) : 문자, %c

int (Integer) : 정수, %d

float (float) : 부동소수점, 不動 이 아니다. 소수점이 이리저리 떠다닌다는 부동의 의미, %f

Tip) String : 문자열, 문자 여러 개가 합쳐진 것을 의미 C에서는 사용할 수 없다.

<자주 쓰는 용어>



#include <stdio.h>

int main(int argc, char\* argv[]){

return 0;

}

#include <stdio.h> :

#은 전처리기를 의미하여 컴파일 전에 미리 처리해야 한다고 말해주는 명령어

stdio.h의 .h는 헤더 파일로, 사용자가 만들었거나 이미 만들어져 있는 함수들을 정리한 목차

<>은 이미 만들어져 있는 헤더파일을 의미

“”은 사용자가 만든 헤더 파일을 의미

int main() :

C언어의 가장 기본적인 함수

실행할 때 가장 먼저 실행되는 함수

return 0; :

함수가 종료될 때 함수 내에서 도출된 값을 돌려주는 역할

void 와 같은 경우 사용하지 않는다.

Tip) ; 은 문장이 끝날 때 사용하는 마침표의 역할을 한다. Python에서는 사용하지 않지만 대부분의 언어에서는 끝에 세미콜론을 찍어야 명령을 끝낸다.

<자주 쓰는 용어>

Call by Value / Call by reference :

값에 의한 호출 / 참조에 의한 호출

void func(int \*n){

\*n=20;

}

void main(){

int n=10;

func(&n);

printf(“%d”,n);

}

void func(int n){

n=20;

}

void main(){

int n=10;

func(n);

printf(“%d”,n);

}

Call by value 는 데이터의 값을 복사해서 함수로 전달 / 원본 데이터 값이 변경되지 않는다.

Call by reference는 데이터의 값이 아닌 주소를 함수로 전달 / 원본 데이터 값을 변경 가능하다.

<자주 쓰는 용어>

Parameter / Argument :

매개변수 / 전달인자

매개변수는 변수(variable)로, 전달인자는 값(value)으로 보는 것이 일반적이다.

//parameter (매개변수)

//함수의 정의 부분에 나열되어 있는 변수, 여기서는 plus 함수 정의시에 사용되는 a, b를 parameter(매개변수) 라고 한다.

int plus(int a, int b){

return a + b

}

//argument (전달인자)

//함수를 호출할때 전달 되는 실제 값, 여기서는 plus 라는 함수에 넣어주는 값 1, 2를 argument(전달인자)라고 한다.

result = plus(1, 2);

<자주 쓰는 용어>

Prototype :

#include <stdio.h>

void showDan(int num); //이것을 Prototype 이라고 한다.

int main(void){

int dan = 1;

int num = 1;

for (dan = 1; dan <= 9; dan++){

showDan(dan);

}

}

void showDan(int num){

printf("=== %d단 ===\n", dan);

for (num = 1; num <= 9; num++){

printf("%d \* %d = %d\n", dan, num, dan \* num);

}

}

함수의 선언을 맨 위에 적어주고 정의를 아래에 하는 경우,

맨 위의 함수의 선언을 프로토타입이라고 한다