

11_4_1 리뷰 및 개선

목차

- 최초 코드 소개
- 최초 코드 실행
- 개선 코드 소개
- 개선 코드 실행

최초 코드

```
응용11_4_1.cpp > main(void)
1  #include <stdio.h>
2  char *ms_char[10]={"센티미터", "미터", "킬로미터", "인치", "피트",
3  | | |, "야드", "마일", "자(척)", "간", "리"};
4  void print_distance_unit(void);
5  void measure_calculate(int basic, int transe, double measure);
6
7  int main(void)
8  {
9      double measure;
10     int basic, transe;
11
12     printf("길이에 대한 도량형 환산 프로그램\n\n");
13     print_distance_unit();
14     printf("\n\n기본단위와 변환단위\n입력 후 Enter(예:1 5)>");
15     scanf("%d %d",&basic,&transe);
16
17     printf("값을 입력하고 Enter>");
18     scanf("%lf", &measure);
19     printf("\n");
20     measure_calculate(basic, transe, measure);
21     return 0;
22 }
23
24 void print_distance_unit(void)
25 {
26     int i;
27     for(i=0;i<=9;i++)
28     {
29         printf("%d:%-8s  ", i, ms_char[i]);
30         if (i%3==2)
31             printf("\n");
32     }
33 }
34
35 void measure_calculate(int basic, int transe, double measure)
36 {
```

```
void measure_calculate(int basic, int transe, double measure)
{
    //[함수 11.4.1]의 정의 부분 참고
    double msre[10], unit, result;

    msre[0]=392772; //centimeter
    msre[1]=msre[0]/100.; //meter
    msre[2]=msre[1]/1000.; //kilometer
    msre[3]=msre[0]/2.54; //inch
    msre[4]=msre[3]/12.; //feet
    msre[5]=msre[4]/3.; //yard
    msre[6]=msre[5]/1760.; //mile
    msre[7]=12960.; //자
    msre[8]=msre[7]/6.; //간
    msre[9]=msre[8]/(6.*36); //리

    unit=msre[basic];
    result=measure*(msre[transe]/unit);
    printf("%.2f %8s는 ", measure, ms_char[basic]);
    printf("%15.5f %8s\n", result, ms_char[transe]);
}
```

최초 코드 설명

최초 코드 분석

1. 전체 구조:

- 사용자는 먼저 환산하고 싶은 길이 단위를 선택합니다. 선택 가능한 단위에는 센티미터, 미터, 킬로미터, 인치, 피트, 야드, 마일, 자(척), 간, 리가 있습니다.
- 이후 사용자가 선택한 기본 단위와 변환할 단위, 그리고 해당 길이 값을 입력하면 변환된 결과가 출력됩니다.

2. 함수 설명:

- `print_distance_unit`: 모든 사용 가능한 길이 단위를 화면에 출력하는 함수입니다.
- `measure_calculate`: 기본 단위에서 변환할 단위로의 환산을 수행하고 결과를 출력하는 함수입니다. 이 함수 내에서 각 단위 간의 변환 비율이 미리 정의되어 있습니다.

최초 코드 실행

```
0:센티미터   1:미터       2:킬로미터  
3:인치       4:피트       5:야드  
6:마일       7:자 (척)    8:간  
9:리
```

```
기본단위와 변환단위  
입력 후 Enter(예 :1 5)>
```

```
기본단위와 변환단위  
입력 후 Enter(예 :1 5)>0 1  
값을 입력하고 Enter>100
```

```
100.00 센티미터는          1.00000   미터
```

개선 코드(입력 검증 추가)

```
#include <stdio.h>

// 단위들을 전역 변수로 선언하여 다양한 함수에서 사용 가능하도록 합니다.
char *ms_char[10] = {"센티미터", "미터", "킬로미터", "인치", "피트", "야드", "마일", "자(척)", "간", "리"};

// 단위 목록 출력 함수
void print_distance_unit(void);
// 변환 계산 함수
void measure_calculate(int basic, int transe, double measure);

int main(void)
{
    double measure;
    int basic, transe;
    char again = 'y'; // 사용자에게 계속 변환할 것인지 물어볼 때 사용하는 변수

    while (again == 'y' || again == 'Y')
    {
        printf("길이에 대한 도량형 환산 프로그램\n\n");
        print_distance_unit();
        printf("\n\n기본단위와 변환단위\n입력 후 Enter(예:1 5)>");

        if (scanf("%d %d", &basic, &transe) != 2 || basic < 0 || basic > 9 || transe < 0 || transe > 9)
        {
            printf("잘못된 입력입니다. 다시 입력해주세요.\n");
            continue;
        }

        printf("값을 입력하고 Enter>");
        scanf("%lf", &measure);
        printf("\n");
        measure_calculate(basic, transe, measure);

        printf("\n계속 변환하실꺼면 'y' 그만하실꺼면 'n': ");
        scanf(" %c", &again);
    }
}
```

```
// 단위 목록을 출력하는 함수
void print_distance_unit(void)
{
    int i;
    for(i=0; i<=9; i++)
    {
        printf("%d:%-8s ", i, ms_char[i]);
        if (i%3 == 2)
            printf("\n");
    }
}

// 입력된 기본 단위에서 대상 단위로 길이를 변환하는 함수
void measure_calculate(int basic, int transe, double measure)
{
    double msre[10], unit, result;

    // 각 단위별로 센티미터를 기준으로 한 변환 계수를 정의
    msre[0]=392772; //centimeter
    msre[1]=msre[0]/100.; //meter
    msre[2]=msre[1]/1000.; //kilometer
    msre[3]=msre[0]/2.54; //inch
    msre[4]=msre[3]/12.; //feet
    msre[5]=msre[4]/3.; //yard
    msre[6]=msre[5]/1760.; //mile
    msre[7]=12960.; //자
    msre[8]=msre[7]/6.; //간
    msre[9]=msre[8]/(6.*36); //리

    unit = msre[basic];
    result = measure * (msre[transe] / unit);
    printf("%.2f %8s는 ", measure, ms_char[basic]);
    printf("%15.5f %8s\n", result, ms_char[transe]);
}
```

개선 코드 설명

1. 전역 변수:

- `ms_char` 는 사용 가능한 모든 길이 단위를 저장하는 문자열 배열입니다.

2. main 함수:

- 사용자에게 어떤 단위를 변환하고 싶은지 (`basic`), 어떤 단위로 변환하고 싶은지 (`transe`) 및 변환할 수치 (`measure`)를 묻습니다.
- 이러한 입력은 `again` 이 'y' 또는 'Y'인 동안 계속됩니다.

3. print_distance_unit 함수:

- 사용 가능한 모든 길이 단위를 화면에 출력합니다.

4. measure_calculate 함수:

- `basic` 단위에서 `transe` 단위로 `measure` 를 변환합니다.
- 변환은 `msre` 배열의 변환 계수를 사용하여 수행됩니다. 이 배열에는 각 단위 별로 센티미터를 기준으로 한 변환 계수가 포함되어 있습니다.

이 코드는 사용자에게 두 단위와 길이 값을 입력하게 하고, 그 값을 다른 단위로 변환하여 출력하는 간단한 길이 변환 프로그램입니다. 추가된 `again` 변수를 통한 루프로 인해 사용자는 원하는 만큼 여러 번 변환을 수행할 수 있습니다.

개선 코드 실행

```
기본단위와 변환단위  
입력 후 Enter(예:1 5)>0 1  
값을 입력하고 Enter>100  
  
100.00 센티미터는          1.00000   미터  
  
계속 변환하실꺼면 'y' 그만하실꺼면 'n':
```

```
계속 변환하실꺼면 'y' 그만하실꺼면 'n': y  
길이에 대한 도량형 환산 프로그램
```

```
0:센티미터   1:미터       2:킬로미터  
3:인치       4:피트       5:야드  
6:마일       7:자 (척)    8:간  
9:리
```

```
기본단위와 변환단위  
입력 후 Enter(예:1 5)>1 2  
값을 입력하고 Enter>1000
```

```
1000.00   미터는          1.00000   킬로미터
```

```
계속 변환하실꺼면 'y' 그만하실꺼면 'n': n
```

```
...Program finished with exit code 0  
Press ENTER to exit console.
```


출처

- terms.co.kr