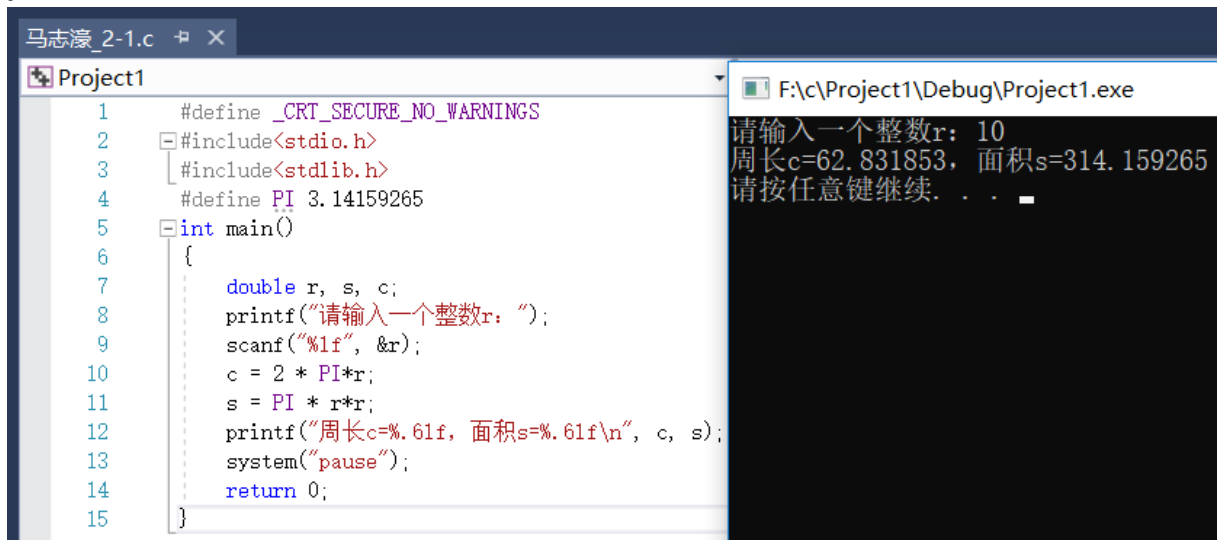**2-1**

```c
#define _CRT_SECURE_NO_WARNINGS
#include<stdio.h>
#include<stdlib.h>
#define PI 3.14159265
int main()
{
    double r, s, c;
    printf("请输入一个整数r：");
    scanf("%lf", &r);
    c = 2 * PI*r;
    s = PI * r*r;
    printf("周长c=%.6lf，面积s=%.6lf\n", c, s);
    system("pause");
    return 0;
}
```

2-2

```c
#define _CRT_SECURE_NO_WARNINGS
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int a;
    printf("Enter x:");
    scanf("%d", &a);
    if (a % 3 == 0 && a % 5 == 0 && a % 7 == 0)
        printf("能同时被 3、5、7 整除\n");
    if ((a % 3 != 0 && a % 5 == 0 && a % 7 == 0) || (a % 3 == 0 && a % 5 != 0 && a % 7 == 0) || (a % 3 == 0 && a % 5 == 0 && a % 7 != 0))
        printf("能被其中两数整除\n");
    if ((a % 3 != 0 && a % 5 != 0 && a % 7 == 0) || (a % 3 == 0 && a % 5 != 0 && a % 7 != 0) || (a % 3 != 0 && a % 5 == 0 && a % 7 != 0))
        printf("能被其中一个数整除\n");
    if (a % 3 != 0 && a % 5 != 0 && a % 7 != 0)
        printf("不能被 3、5、7 任一个数整除\n");
    system("pause");
    return 0;
}
```
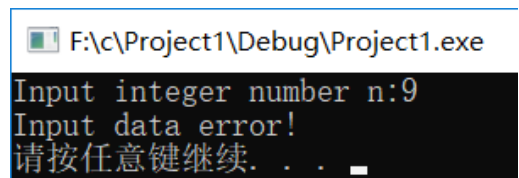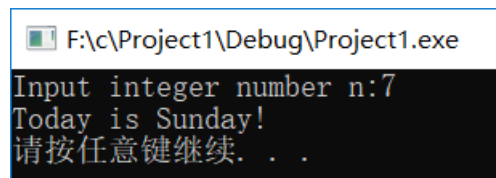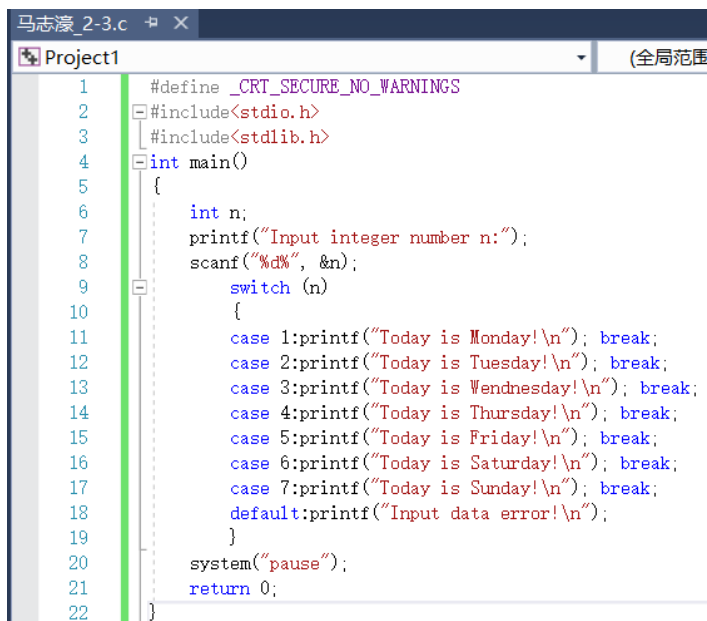

在写 scanf 函数时容易遗漏地址符&。

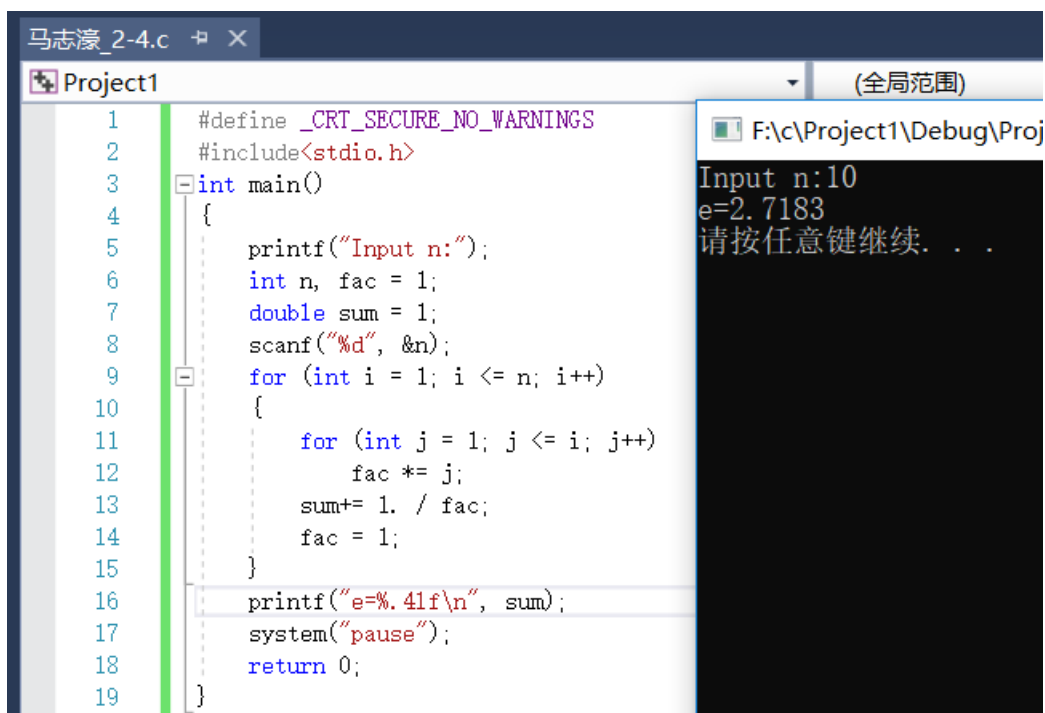2-3

```c
#define _CRT_SECURE_NO_WARNINGS
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int n;
    printf("Input integer number n:");
    scanf("%d%", &n);
        switch (n)
        {
        case 1:printf("Today is Monday!\n"); break;
        case 2:printf("Today is Tuesday!\n"); break;
        case 3:printf("Today is Wendnesday!\n"); break;
        case 4:printf("Today is Thursday!\n"); break;
        case 5:printf("Today is Friday!\n"); break;
        case 6:printf("Today is Saturday!\n"); break;
        case 7:printf("Today is Sunday!\n"); break;
        default:printf("Input data error!\n");
        }
    system("pause");
    return 0;
}
```

switch 函数的运用仍然不是很熟练，需多加训练。
scanf 函数中的后引号丢失。

2-4

```c
#define _CRT_SECURE_NO_WARNINGS
#include<stdio.h>
#include<stdlib.h>
int main()
{
    printf("Input n:");
    int n, fac = 1;
    double sum = 1;
    scanf("%d", &n);
    for (int i = 1; i <= n; i++)
    {
        for (int j = 1; j <= i; j++)
            fac *= j;
        sum+= 1. / fac;
        fac = 1;
    }
    printf("e=%.4lf\n", sum);
    system("pause");
    return 0;
}
```


需要在 13 行的 1 后加 "." 使其变为浮点型。