

```

//26 global and local
#include <stdio.h>

void prove(int local);
//สร้าง function ชื่อ prove มาเพื่อพิสูจน์ local และ global โดยส่ง parameter เป็น int โดยชื่อ local
int glo;
//สร้างตัวแปร global โดยมี datatype เป็น int และตั้งชื่อว่า glo

int main()
{
    int local;
    //สร้างตัวแปร local โดยมี datatype เป็น int และตั้งชื่อว่า local

    printf("global variable before initialize : %d\n", glo);
    //แสดงผลว่า "ตัวแปร global ก่อน initialize " และให้นำค่าของ glo มาแสดง

    printf("local variable before initialize : %d\n\n", local);
    //แสดงผลว่า "ตัวแปร local ก่อน initialize " และให้นำค่าของ local มาแสดง

    prove(local);
    //เรียกใช้ function prove และส่ง parameter ชื่อ local

    printf("\nglobal variable after initialize in another function : %d\n", glo);
    //แสดงผลว่า "ตัวแปร global หลัง initialize ใน function อื่น" และให้นำค่าของ glo มาแสดง

    printf("local variable after initialize in another function : %d\n\n", local);
    //แสดงผลว่า "ตัวแปร local หลัง initialize ใน function อื่น" และให้นำค่าของ local มาแสดง
}

void prove(int local)
{
    printf("input a value in global : ");
    scanf("%d", &glo);
    //แสดงผลว่า "ให้ใส่ค่าใน global" โดยเก็บค่าไว้ในตัวแปร glo

    printf("input a value in local : ");
    scanf("%d", &local);
    //แสดงผลว่า "ให้ใส่ค่าใน local" โดยเก็บค่าไว้ในตัวแปร local

    printf("\nglobal variable after initialize in prove function : %d\n", glo);
    //แสดงผลว่า "ตัวแปร global หลัง initialize ใน function prove" และให้นำค่าของ glo มาแสดง

    printf("local variable after initialize in prove function : %d\n\n", local);
    //แสดงผลว่า "ตัวแปร local หลัง initialize ใน function prove" และให้นำค่าของ local มาแสดง
}

/*สรุป

```

1. ถ้าไม่ initialize ค่าในตัวแปร global จะมีค่า default เป็น 0 แต่ของ local จะเป็นค่าสุ่มๆ

2.การเรียกใช้งานใน **function** อื่น ตัวแปร **global** ไม่จำเป็นต้องส่งเป็น **parameter** แต่ตัวแปร **local** ต้องส่ง **parameter** ไป **function** อื่นเพื่อใช้งาน

3.การ **initialize** ใน **function** อื่น ค่าที่ **initialize** ในตัวแปร **global** จะถูกเก็บค่าไว้ในตัวแปรนั้นๆทั้งโปรแกรม แต่ของ **local** จะถูกเก็บค่าไว้ในตัวแปรนั้นๆแค่นั้นของ **function** ตัวเอง

*/

Result

```
global variable before initialize : 0
local variable before initialize : 3829760

input a value in global : 5
input a value in local : 5

global variable after initialize in prove function : 5
local variable after initialize in prove function : 5

global variable after initialize in another function : 5
local variable after initialize in another function : 3829760

Press any key to continue . . . |
```