```
#include <iostream>
using namespace std;

class SingletonClass
{
        int m_value;
        // the following declaration states that class Singleton has
        // a private class field of type pointer to SingletonClass
        // named theInstance.
        // However the definition of the variable is done outside class
        // (just as an external static variable in C) − see below
        static SingletonClass *theInstance;

        SingletonClass(int v = 0)
        {
                m_value = v;
        }
public:
        int get_value()
        {
                return m_value;
        }
        void set_value(int v)
        {
                m_value = v;
        }
        static SingletonClass *getInstance()
        {
                if (!theInstance)
                        theInstance = new SingletonClass(10);
                return theInstance;
        }

        static int c_value;
};

// Access specifiers will not give you error while defining the member.
SingletonClass *SingletonClass::theInstance;
int SingletonClass::c_value = 128;

// You can also declare regular external/external static variables
int regularExternal = 44;
static int regularExternalStatic;

void main() {
        SingletonClass *singleton1 = SingletonClass::getInstance();
        SingletonClass *singleton2 = SingletonClass::getInstance();

        // You cannot do the following since theInstance is privatre
        // SingletonClass *singleton3 = SingletonClass::theInstance;

        // the following three statements are ok
        SingletonClass::c_value = 24;
        regularExternal = 22;
        regularExternalStatic = −4;
}

```



```
; Listing generated by Microsoft (R) Optimizing Compiler Version 19.12.25834.0

        TITLE   c:\K−State\cis450\Programs\CoursePrograms\Chap3\Singleton\Singleton.cpp
        .686P
        .XMM
        include listing.inc
        .model  flat

INCLUDELIB LIBCMT
INCLUDELIB OLDNAMES

PUBLIC  ?theInstance@SingletonClass@@0PAV1@A            ; SingletonClass::theInstance
PUBLIC  ?c_value@SingletonClass@@2HA                    ; SingletonClass::c_value
PUBLIC  ?regularExternal@@3HA                           ; regularExternal
_BSS    SEGMENT
?theInstance@SingletonClass@@0PAV1@A DD 01H DUP (?)     ; SingletonClass::theInstance
_BSS    ENDS
_DATA   SEGMENT
?c_value@SingletonClass@@2HA DD 080H                    ; SingletonClass::c_value
?regularExternal@@3HA DD 02cH                           ; regularExternal
_DATA   ENDS
PUBLIC  ??0SingletonClass@@AAE@H@Z                      ; SingletonClass::SingletonClass
PUBLIC  ?getInstance@SingletonClass@@SAPAV1@XZ          ; SingletonClass::getInstance
PUBLIC  _main
EXTRN   ??2@YAPAXI@Z:PROC                               ; operator new
_BSS    SEGMENT
?regularExternalStatic@@3HA DD 01H DUP (?)              ; regularExternalStatic
_BSS    ENDS
; Function compile flags: /Odtp
; File c:\k−state\cis450\programs\courseprograms\chap3\singleton\singleton.cpp
_TEXT   SEGMENT
_singleton2$ = −8                                       ; size = 4
_singleton1$ = −4                                       ; size = 4
_main   PROC

; 45   : void main() {

  00000 55              push    ebp
  00001 8b ec           mov     ebp, esp
  00003 83 ec 08        sub     esp, 8

; 46   :        SingletonClass *singleton1 = SingletonClass::getInstance();

  00006 e8 00 00 00 00  call    ?getInstance@SingletonClass@@SAPAV1@XZ ; SingletonClass::getInstance
  0000b 89 45 fc        mov     DWORD PTR _singleton1$[ebp], eax

; 47   :        SingletonClass *singleton2 = SingletonClass::getInstance();

  0000e e8 00 00 00 00  call    ?getInstance@SingletonClass@@SAPAV1@XZ ; SingletonClass::getInstance
  00013 89 45 f8        mov     DWORD PTR _singleton2$[ebp], eax

; 48   :
; 49   :        // You cannot do the following since theInstance is privatre
; 50   :        // SingletonClass *singleton3 = SingletonClass::theInstance;
; 51   :
; 52   :        // the following three statements are ok
; 53   :        SingletonClass::c_value = 24;

  00016 c7 05 00 00 00
        00 18 00 00 00  mov     DWORD PTR ?c_value@SingletonClass@@2HA, 24 ; SingletonClass::c_value, 00000018H

; 54   :        regularExternal = 22;

```

```
  00020 c7 05 00 00 00
        00 16 00 00 00   mov     DWORD PTR ?regularExternal@@3HA, 22 ; regularEx
ternal, 00000016H

; 55   :        regularExternalStatic = -4;

  0002a c7 05 00 00 00
        00 fc ff ff ff   mov     DWORD PTR ?regularExternalStatic@@3HA, -4 ; fff
ffffcH

; 56   : }

  00034 33 c0            xor     eax, eax
  00036 8b e5            mov     esp, ebp
  00038 5d               pop     ebp
  00039 c3               ret     0
_main   ENDP
_TEXT   ENDS
; Function compile flags: /Odtp
; File c:\k-state\cis450\programs\courseprograms\chap3\singleton\singleton.cpp
;       COMDAT ?getInstance@SingletonClass@@SAPAV1@XZ
_TEXT   SEGMENT
tv74 = -8                                               ; size = 4
$T1 = -4                                                ; size = 4
?getInstance@SingletonClass@@SAPAV1@XZ PROC             ; SingletonClass::getIns
tance, COMDAT

; 28   :        {

  00000 55               push    ebp
  00001 8b ec            mov     ebp, esp
  00003 83 ec 08         sub     esp, 8

; 29   :                if (!theInstance)

  00006 83 3d 00 00 00
        00 00            cmp     DWORD PTR ?theInstance@SingletonClass@@0PAV1@A,
 0 ; SingletonClass::theInstance
  0000d 75 31            jne     SHORT $LN2@getInstanc

; 30   :                        theInstance = new SingletonClass(10);

  0000f 6a 04            push    4
  00011 e8 00 00 00 00   call    ??2@YAPAXI@Z            ; operator new
  00016 83 c4 04         add     esp, 4
  00019 89 45 fc         mov     DWORD PTR $T1[ebp], eax
  0001c 83 7d fc 00      cmp     DWORD PTR $T1[ebp], 0
  00020 74 0f            je      SHORT $LN4@getInstanc
  00022 6a 0a            push    10                      ; 0000000aH
  00024 8b 4d fc         mov     ecx, DWORD PTR $T1[ebp]
  00027 e8 00 00 00 00   call    ??0SingletonClass@@AAE@H@Z ; SingletonClass::Si
ngletonClass
  0002c 89 45 f8         mov     DWORD PTR tv74[ebp], eax
  0002f eb 07            jmp     SHORT $LN5@getInstanc
$LN4@getInstanc:
  00031 c7 45 f8 00 00
        00 00            mov     DWORD PTR tv74[ebp], 0
$LN5@getInstanc:
  00038 8b 45 f8         mov     eax, DWORD PTR tv74[ebp]
  0003b a3 00 00 00 00   mov     DWORD PTR ?theInstance@SingletonClass@@0PAV1@A,
 eax ; SingletonClass::theInstance
$LN2@getInstanc:

; 31   :                return theInstance;

  00040 a1 00 00 00 00   mov     eax, DWORD PTR ?theInstance@SingletonClass@@0PA
V1@A ; SingletonClass::theInstance

; 32   :        }

  00045 8b e5            mov     esp, ebp
  00047 5d               pop     ebp
  00048 c3               ret     0
```



```
?getInstance@SingletonClass@@SAPAV1@XZ ENDP             ; SingletonClass::getIns
tance
_TEXT   ENDS
; Function compile flags: /Odtp
; File c:\k-state\cis450\programs\courseprograms\chap3\singleton\singleton.cpp
;       COMDAT ??0SingletonClass@@AAE@H@Z
_TEXT   SEGMENT
_this$ = -4                                             ; size = 4
_v$ = 8                                                 ; size = 4
??0SingletonClass@@AAE@H@Z PROC                         ; SingletonClass::Single
tonClass, COMDAT
; _this$ = ecx

; 14   :        SingletonClass(int v = 0)

  00000 55               push    ebp
  00001 8b ec            mov     ebp, esp
  00003 51               push    ecx
  00004 89 4d fc         mov     DWORD PTR _this$[ebp], ecx

; 15   :        {
; 16   :                m_value = v;

  00007 8b 45 fc         mov     eax, DWORD PTR _this$[ebp]
  0000a 8b 4d 08         mov     ecx, DWORD PTR _v$[ebp]
  0000d 89 08            mov     DWORD PTR [eax], ecx

; 17   :        }

  0000f 8b 45 fc         mov     eax, DWORD PTR _this$[ebp]
  00012 8b e5            mov     esp, ebp
  00014 5d               pop     ebp
  00015 c2 04 00         ret     4
??0SingletonClass@@AAE@H@Z ENDP                         ; SingletonClass::Single
tonClass
_TEXT   ENDS
END
```