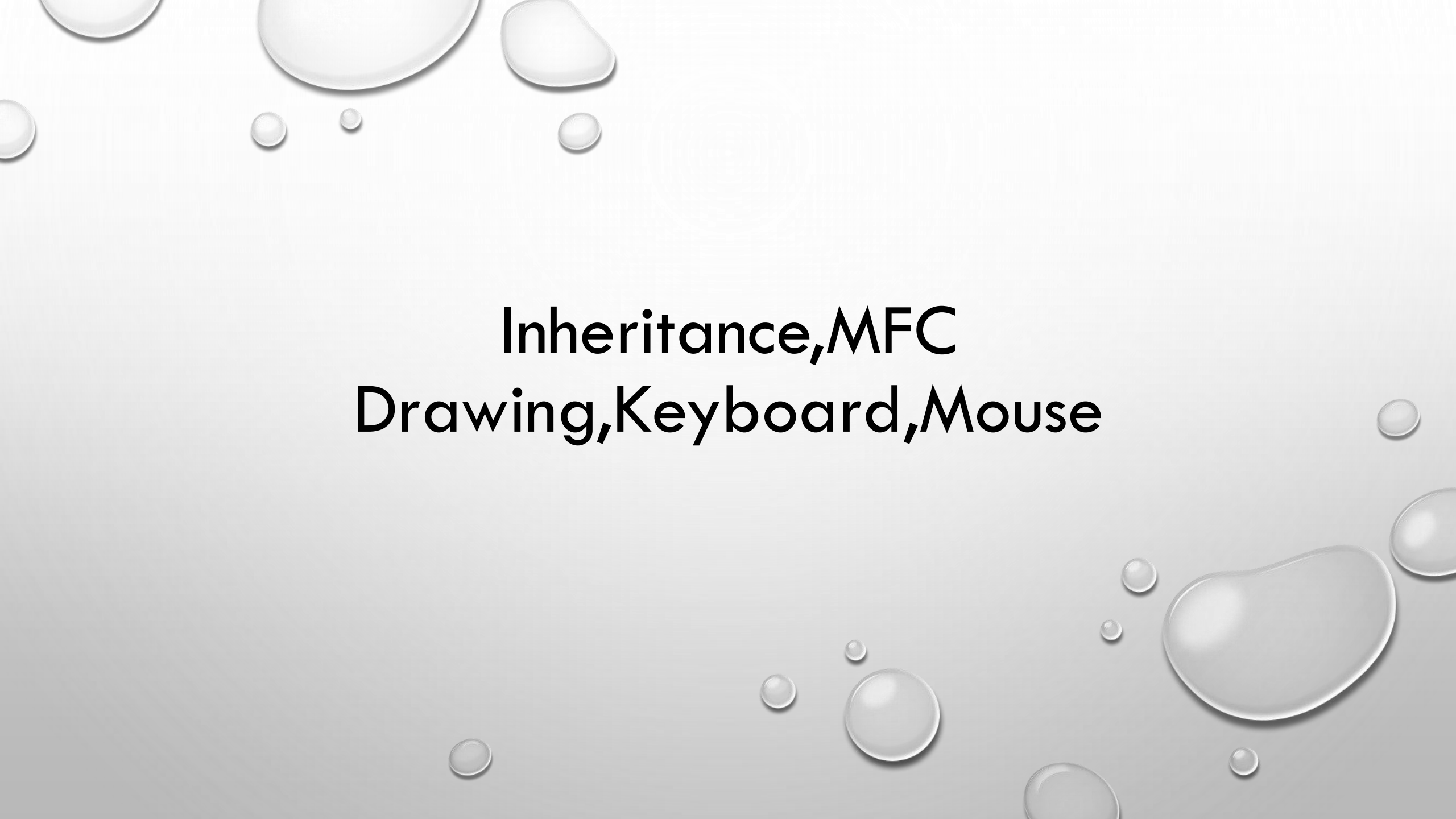


The background of the slide is a light gray gradient, decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and subtle. They are scattered across the slide, with a higher concentration in the top-left and bottom-right corners, and a few near the center.

Inheritance,MFC

Drawing,Keyboard,Mouse

INHERITANCE

You create a new class to inherit the properties of an existing class.

```
class Parent {  
public:  
    void show() { std::cout << "Parent show\n"; }  
};  
class Child : public Parent {  
public:  
    void show() { std::cout << "Child show\n"; } // 隱藏了父類別的 show()  
};  
int main() {  
    Child c;  
    c.show(); // 印出 "Child show"  
    c.Parent::show(); // 如果硬要呼叫父類別的，必須加上範疇解析運算子 `::`  
}
```

MFC Drawing

建立新專案

最近使用的專案範本(R)

-  空白專案
- C++
-  MFC 應用程式
- C++

搜尋範本 (Alt+S)(S)

所有語言(L) 所有平台(P) 所有專案類型(T)

 建置不依賴 .sln 或 .vcxproj 檔案的新式跨平台 C++ 應用程式。

C++ Windows Linux 主控台 macOS

 Windows 傳統式精靈

使用精靈建立您自己的 Windows 應用程式。

C++ Windows 桌面 主控台 程式庫

 Windows 傳統型應用程式

此專案適用於在 Windows 執行並有圖形化使用者介面的應用程式。

C++ Windows 桌面

 空白方案

建立不含專案的空白方案

其他

 MFC 應用程式

建置在 Windows 上執行且具有複雜使用者介面的應用程式。

C++ Windows 桌面

 動態連結程式庫 (DLL)

建置可以在多個執行中 Windows 應用程式之間共用的 .dll。

C++ Windows 程式庫

下一步(N)

設定新的專案

MFC 應用程式 C++ Windows 桌面

專案名稱(I)

MFCteach

位置(L)

C:\Users\User\source\repos

方案(S)

建立新方案

解決方案名稱(M) ⓘ

MFCteach

☒ 將解決方案與專案置於相同目錄中(D)

專案 將在「C:\Users\User\source\repos\MFCteach\」中建立

上一步(B) 建立(C)

MFC 應用程式

應用程式類型選項

應用程式類型

文件範本屬性

使用者介面功能

進階功能

產生的類別

應用程式類型(T)

單一文件

應用程式類型選項:

☐ 索引文件(B)

☒ 文件/檢視架構支援(V)

對話方塊選項(I)

<無>

複合文件支援

<無>

文件支援選項:

☐ 主動式文件伺服器(A)

☐ 主動式文件容器(D)

☐ 支援複合檔案(U)

專案樣式

MFC standard

視覺效果樣式與色彩(Y)

Windows Native/Default

☐ 啟用視覺樣式切換(C)

資源語言(L)

zh-TW

使用 MFC

在共用 DLL 中使用 MFC

上一步

下一步

完成

取消

類別檢視

<搜尋>

MFCteach

對應

巨集和常數

全域函式和變數

CAboutDlg

CMainFrame

CMFCTestApp

CMFCTestDoc

CMFCTestView

~CMFCTestView()

AssertValid() const

CMFCTestView()

Dump(CDumpContext & dc) const

GetDocument() const

OnBeginPrinting(CDC* pDC, CPrintInfo* pInfo)

OnDraw(CDC* pDC)

OnEndPrinting(CDC* pDC, CPrintInfo* pInfo)

OnPreparePrinting(CPrintInfo* pInfo)

PreCreateWindow(CREATESTRUCT& cs)

MFCteach

CMFCTestView

```
1
2 // CMFCTestView.h: CMFCTestView 類別的介面
3 //
4
5 #pragma once
6
7
8 class CMFCTestView : public CView
9 {
10 protected: // 僅從序列化建立
11     CMFCTestView() noexcept;
12     DECLARE_DYNCREATE(CMFCTestView)
13
14 // 屬性
15 public:
16     CMFCTestDoc* GetDocument() const;
17
18 // 作業
19 public:
20
21 // 覆寫
22 public:
23     virtual void OnDraw(CDC* pDC); // 覆寫以描繪此檢視
24     virtual BOOL PreCreateWindow(CREATESTRUCT& cs);
25 protected:
26     virtual BOOL OnPreparePrinting(CPrintInfo* pInfo);
27     virtual void OnBeginPrinting(CDC* pDC, CPrintInfo* pInfo);
28     virtual void OnEndPrinting(CDC* pDC, CPrintInfo* pInfo);
29
30 // 程式碼實作
31 public:
32     virtual ~CMFCTestView();
33 #ifdef _DEBUG
34     virtual void AssertValid() const;
35     virtual void Dump(CDumpContext& dc) const;
36 #endif
37
38 protected:
39
40 // 產生的訊息對應函式
41 protected:
```

89 %

找不到任何問題

輸出

顯示輸出來源(S):

輸出

錯誤清單

Draw a rectangle&circle

```
void CmfcView::OnDraw(CDC* pDC)
{
    CmfcDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if (!pDoc)
        return;

    // TODO: add draw code for native data here

    pDC->Rectangle(100, 100, 300, 300); → Draw a square
    pDC->Ellipse(100,100,300,300);      → Draw a circle
}
```

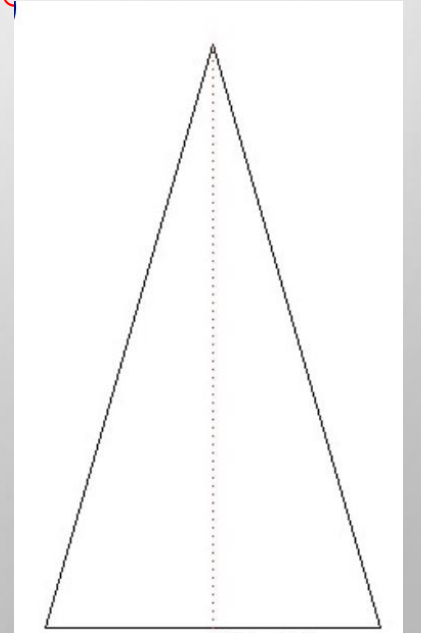
MoveTo&LineTo

```
VOID CMFCVIEW::ONDRAW(CDC* PDC)
{
    CMFCDOC* PDOC = GETDOCUMENT();
    ASSERT_VALID(PDOC);
    IF (!PDOC)
        RETURN;

    // TODO: ADD DRAW CODE FOR NATIVE DATA
    HERE

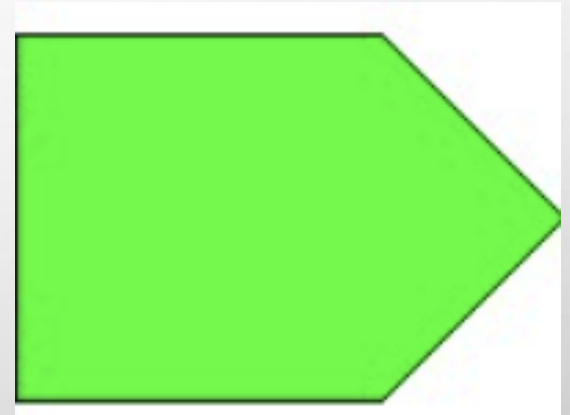
    PDC->MOVETO(200, 50);
    PDC->LINETO(100, 500);
    PDC->LINETO(300, 500);
    PDC->LINETO(200, 50);
```

```
for (int y = 50; y <= 500; y
    += 5)
    pDC->SetPixel(200, y, RGB(0,
    0, 255)); → draw a dashed line
}
```



Polygon()

```
void CmfcView::OnDraw(CDC* pDC)
{
    CBrush aBrush( RGB(0,255,0) );
    CBrush* pBrush= pDC->SelectObject(&aBrush);
    CPoint p[5] = { CPoint(100, 100), CPoint(100, 200), CPoint(200, 200),
    CPoint(250, 150), CPoint(200, 100) };
    pDC->Polygon(p, 5);
    pDC->SelectObject(pBrush);
}
```



TextOut()&SetTextColor()

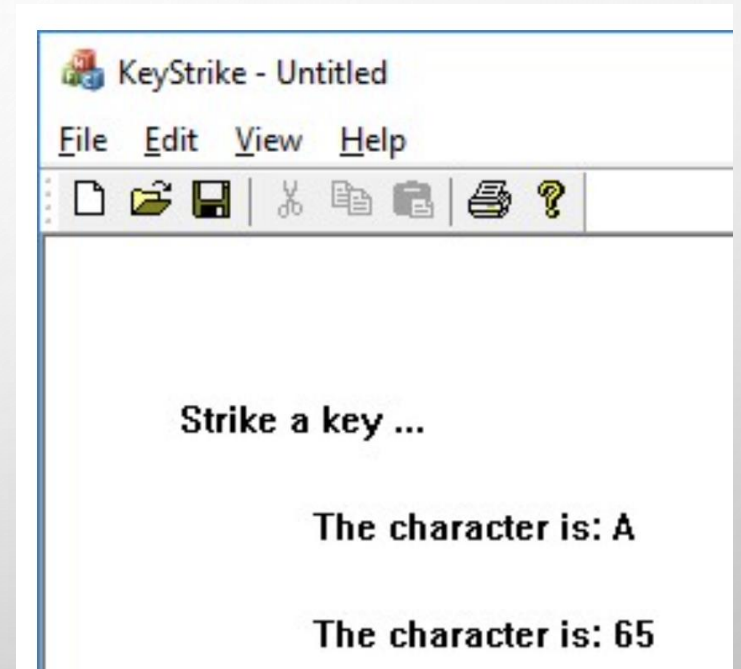
```
void CEX01View::OnDraw(CDC* pDC)
{
    CEX01Doc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if (!pDoc)
        return;

    pDC->TextOut(50, 50, L"Good evening");
    pDC->SetTextColor(RGB(0, 0, 255)); →Change the text color to blue.
    pDC->TextOut(50, 100, L"Good evening");
}
```

Exercise

Please draw a 火柴人 inside a room using shapes such as rectangles, circles, solid lines, and triangles.

KeyStrike

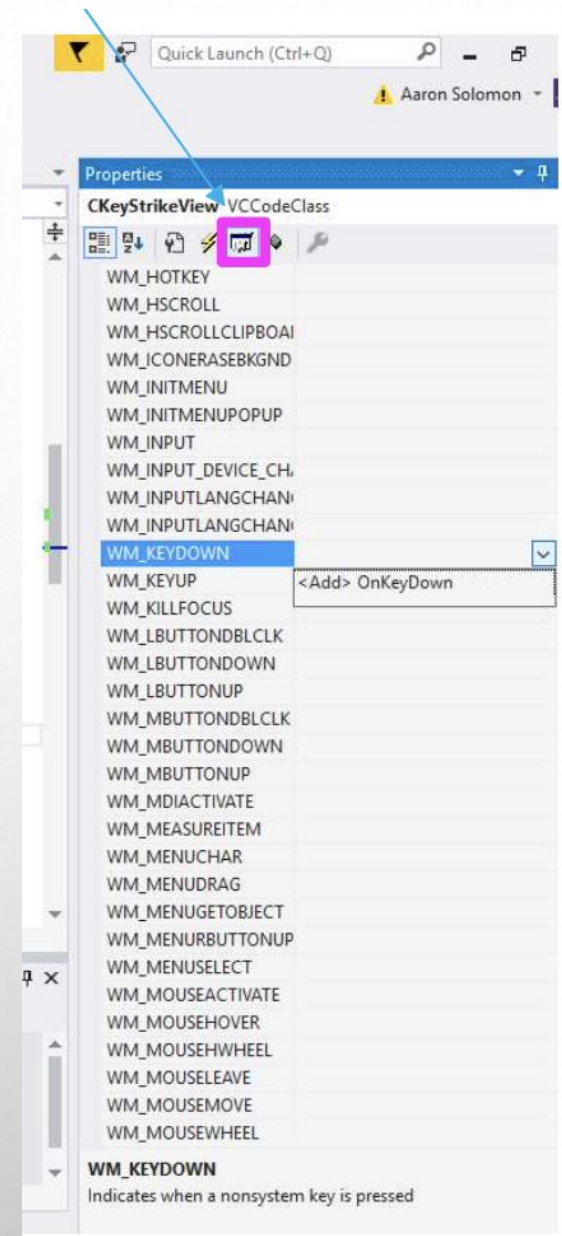
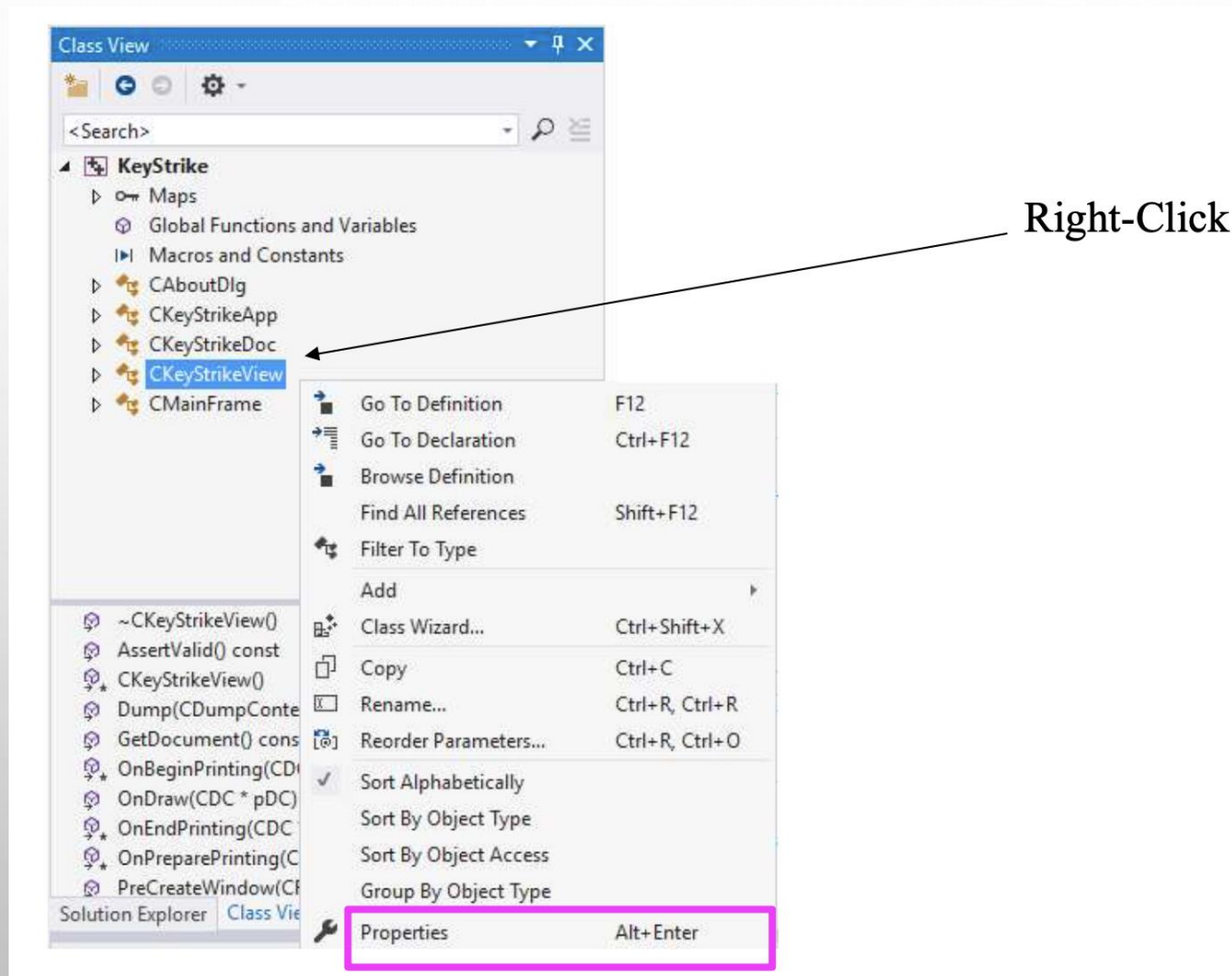


KEYDOWN VS BUTTONDOWN

- WM_KEYDOWN → PRESS A KEY ON THE KEYBOARD.
- WM_LBUTTONDOWN → CLICK THE LEFT MOUSE BUTTON.

OnDraw()

```
void CKeyStrikeView::OnDraw(CDC* pDC)
{
    CKeyStrikeDoc* pDoc = GetDocument();
    ASSERT_VALID(pDoc);
    if (!pDoc)
        return;
    pDC->TextOut(50, 60, L"Strike a key ...");
}
```



OnKeyDown()

```
void CKeyStrikeView::OnKeyDown(UINT nChar, UINT nRepCnt, UINT nFlags)
```

```
{
```

```
// TODO: Add your message handler code here and/or call default
```

```
CClientDC aDC(this);
```

```
CString Buffer;
```

```
Buffer.Format(L"The character is: %c", nChar);
```

```
aDC.TextOut(100, 100, Buffer);
```

```
Buffer.Format(L"The character is: %d", nChar);
```

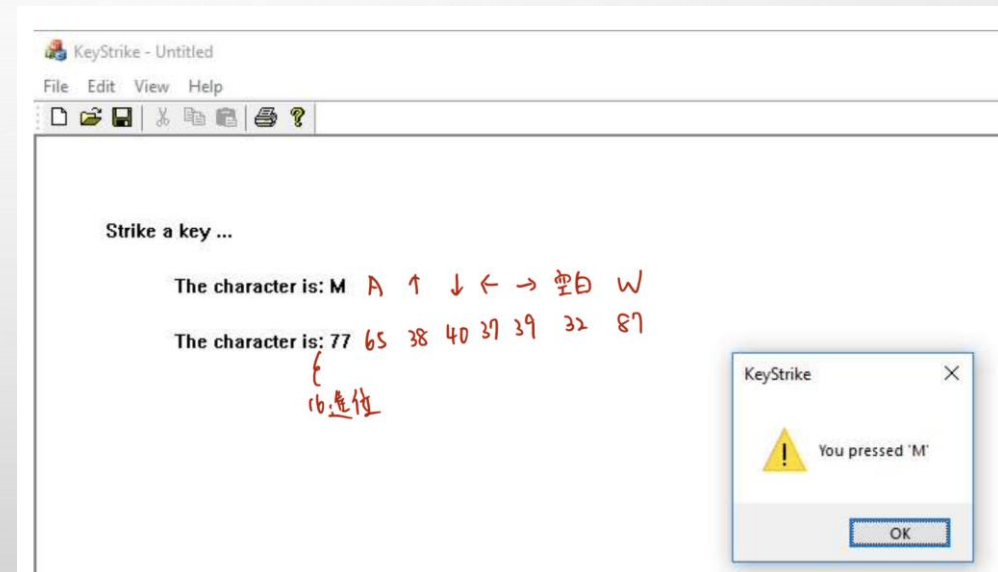
```
aDC.TextOut(100, 140, Buffer);
```

```
if (nChar== 'M')
```

```
AfxMessageBox(L"You pressed 'M'", MB_OK);
```

```
}
```

```
CView::OnKeyDown(nChar, nRepCnt, nFlags);
```



Exercise:Control Snake by WASD Keys

Please use 'A' ,'S','D' to control snake .

```
void CSnakeView::OnKeyDown(UINT nChar, UINT nRepCnt, UINT
nFlags)
{
    int x, y;
    switch (nChar) {
        case 'W':
            x = m_snake[0].x;
            y = m_snake[0].y - 1;
            m_snake.insert(m_snake.begin(), CPoint(x, y));
            m_snake.pop_back();
            break;
    }
    RedrawWindow();
    CView::OnKeyDown(nChar, nRepCnt, nFlags);
}
```

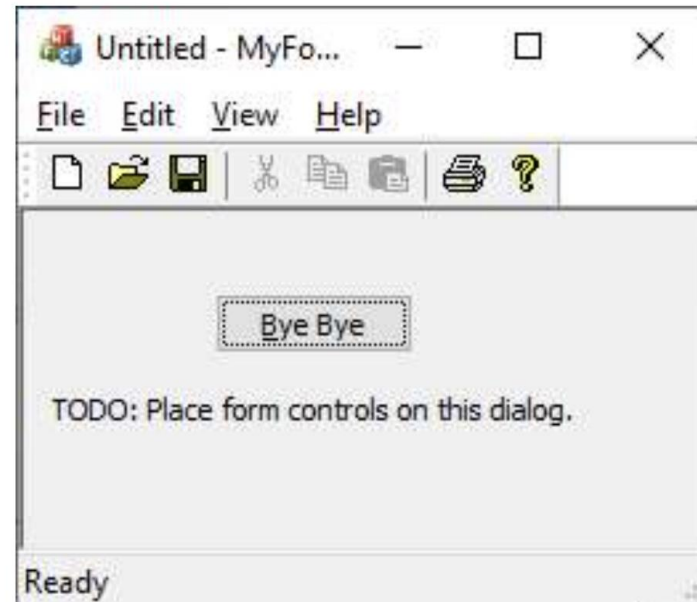
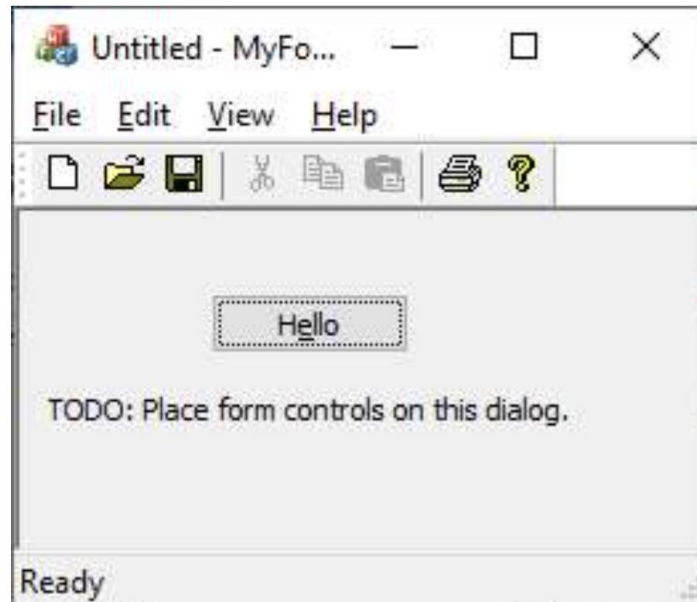


MFC Controls

Common Controls

- Static Controls
- Button Controls
- Edit Controls

Button Controls



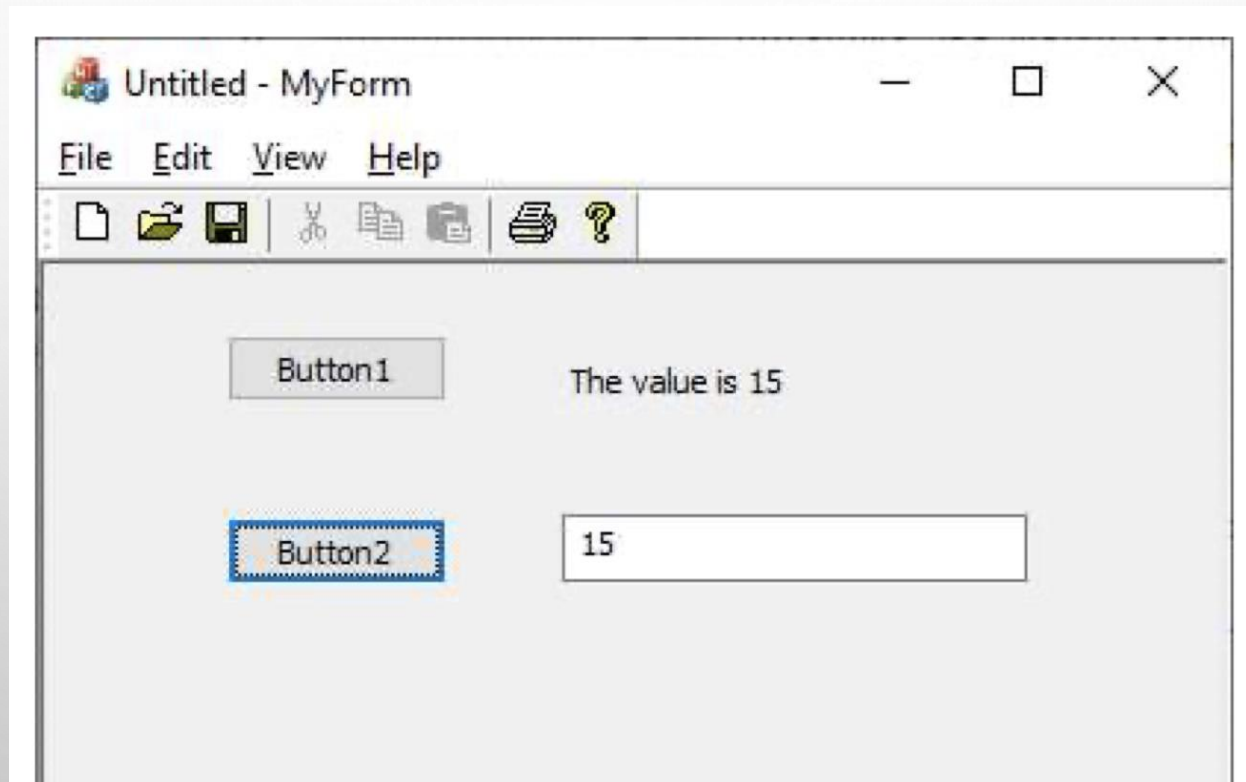
OnBnClickCutton()

```
void CMyFormView::OnBnClickedButton1()
{
    static bool shown = false; // static
    variable

    CWnd* pBtn = GetDlgItem(IDC_BUTTON1);
    if (shown)
        pBtn->SetWindowText(L"&Bye Bye");
    else
        pBtn->SetWindowText(L"H&ello");
    shown = !shown;
}
```

```
void CMyFormView::OnBnClickedButton2()
{
    CClientDC aDC(this);
    aDC.SetROP2(R2_NOTXORPEN);
    static bool shown = false;
    aDC.Ellipse(50,50, 150, 150);
    CWnd* pBtn = GetDlgItem(IDC_BUTTON2);
    if (shown)
        pBtn->SetWindowText(L"&Show");
    else
        pBtn->SetWindowText(L"H&ide");
    shown = !shown;
}
```

Edit Box Control



```
void CMyFormView::OnBnClickedButton1()
{
    GetDlgItem(IDC_EDIT1)->SetWindowText(L"Hello");
}
```

```
void CMyFormView::OnBnClickedButton2()
{
    CString str, msg;
    GetDlgItem(IDC_EDIT1)->GetWindowText(str);
    int i = _ttoi(str);
    msg.Format( L"The value is %d", i);
    GetDlgItem(IDC_STATIC)-
    >SetWindowText(msg);
}
```

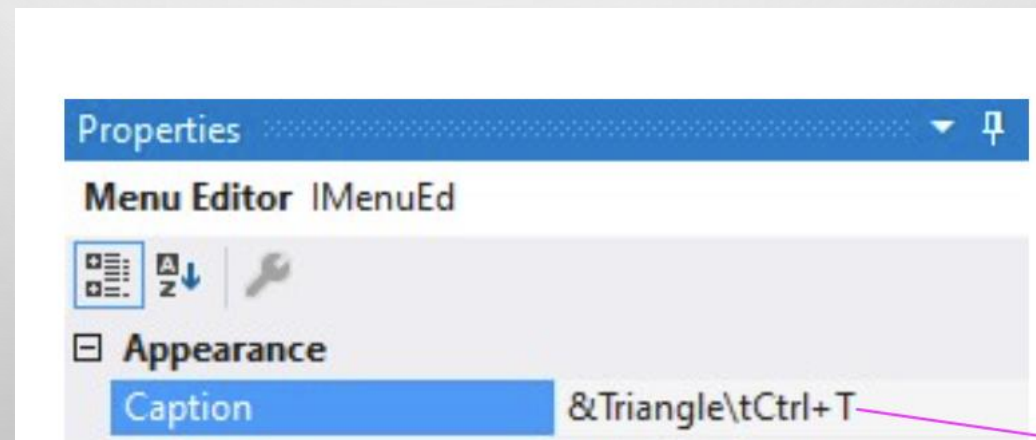
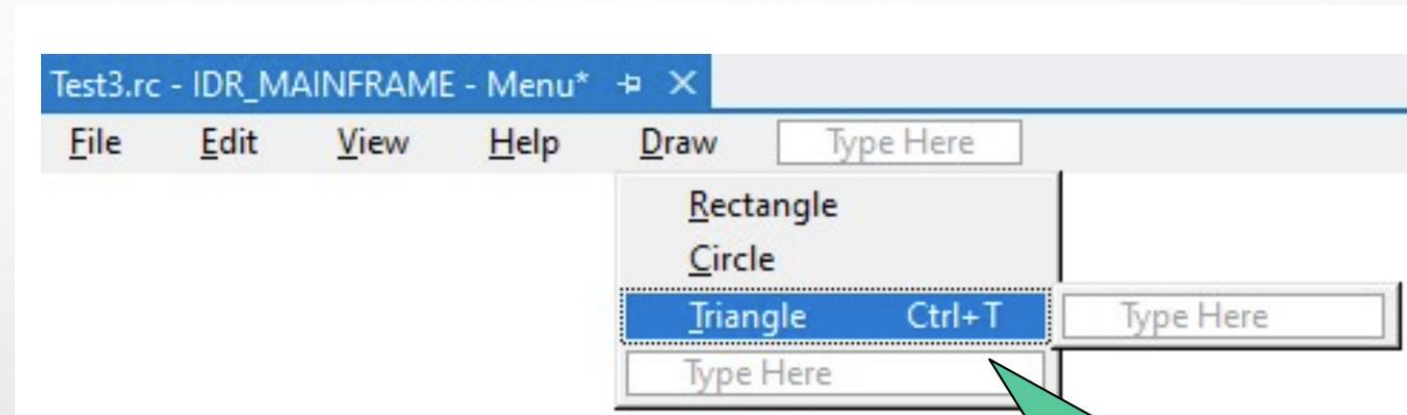
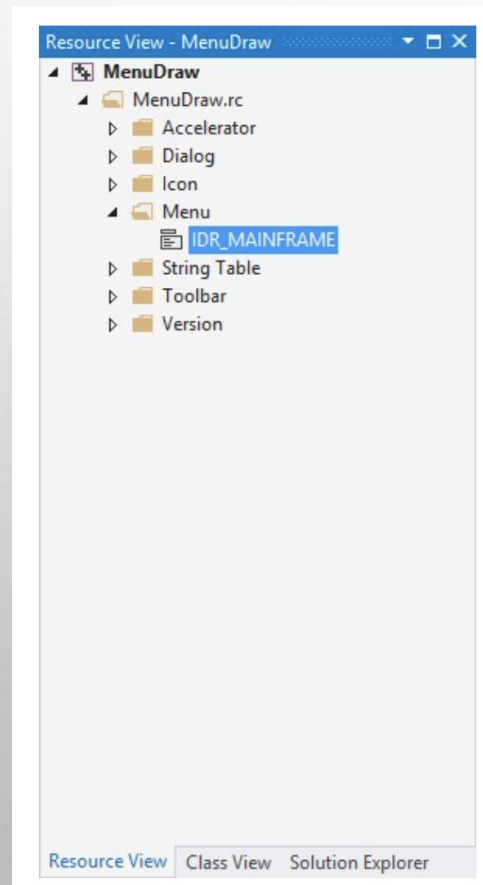
EXERCISE

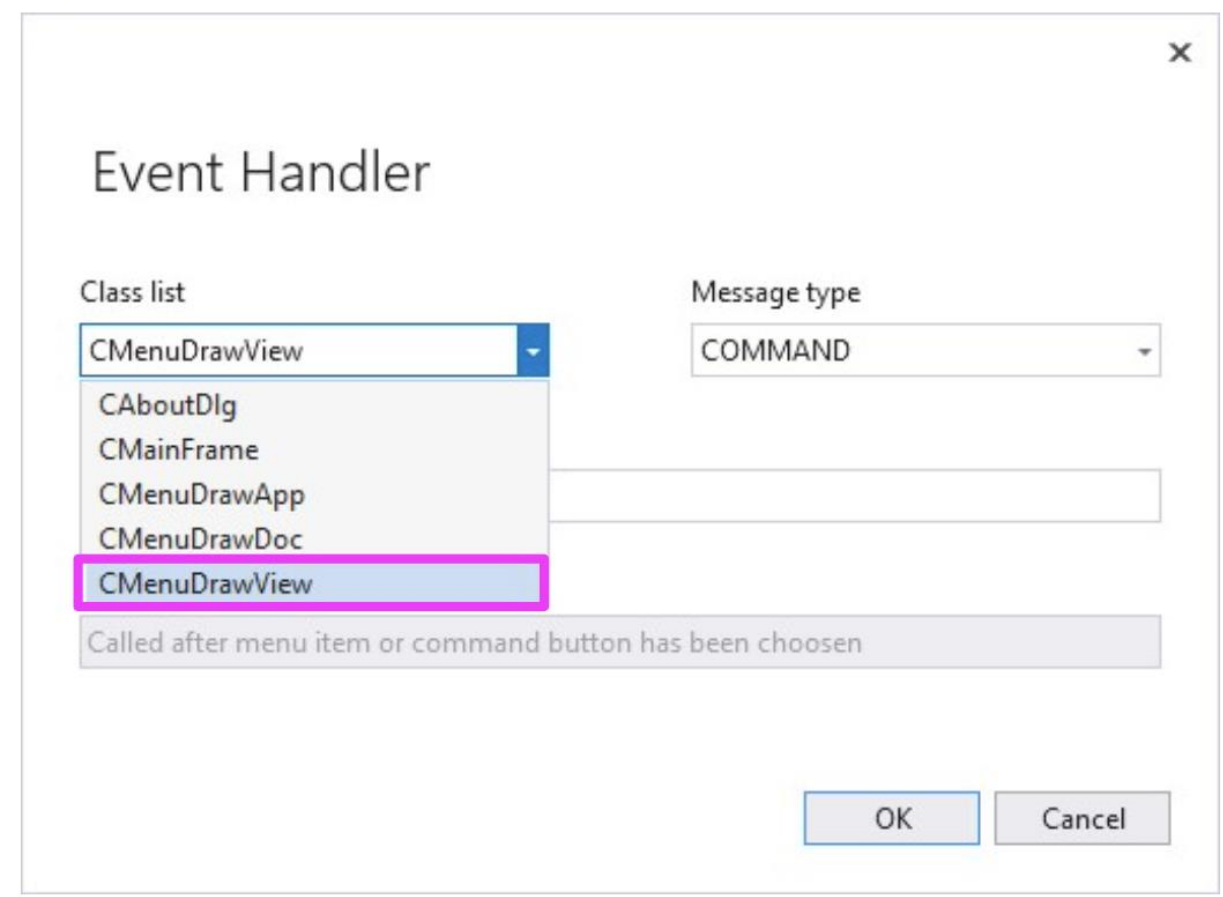
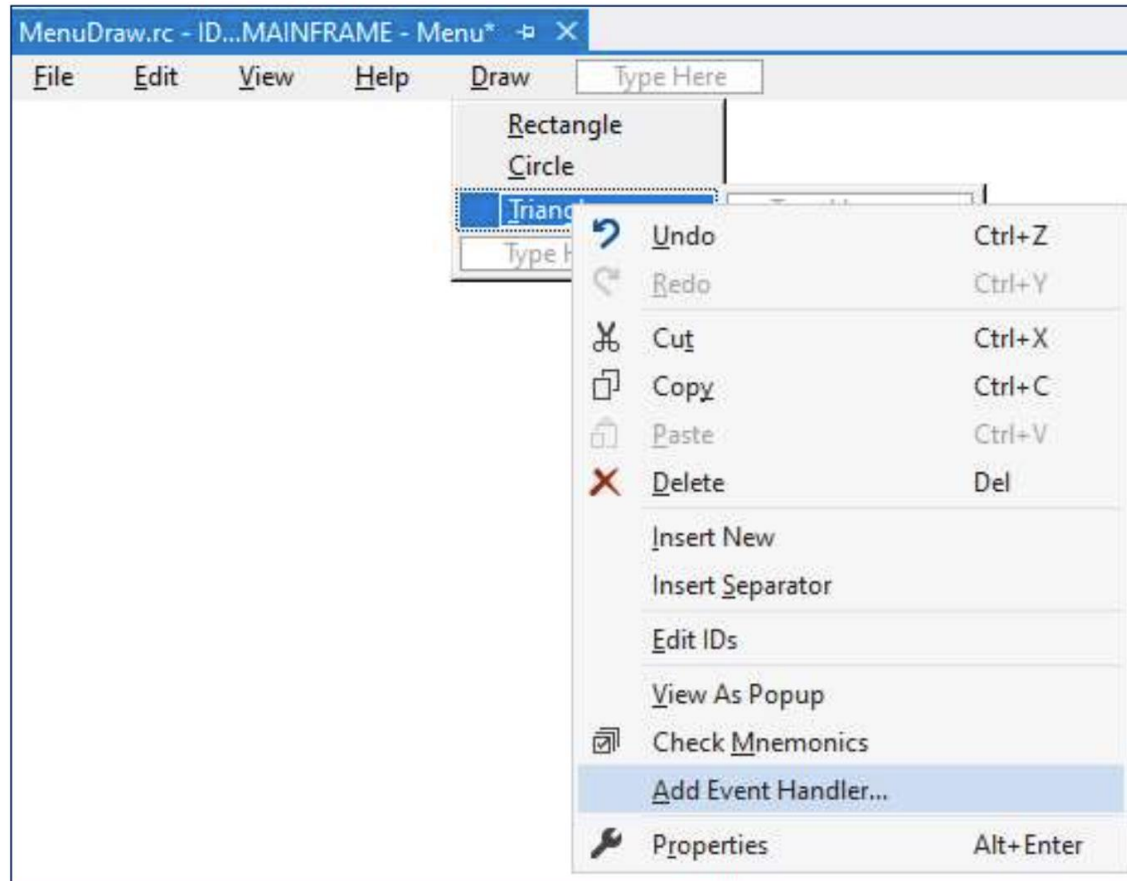
Please use 'Button control' and 'Edit control' to calculate the area of a rectangle.



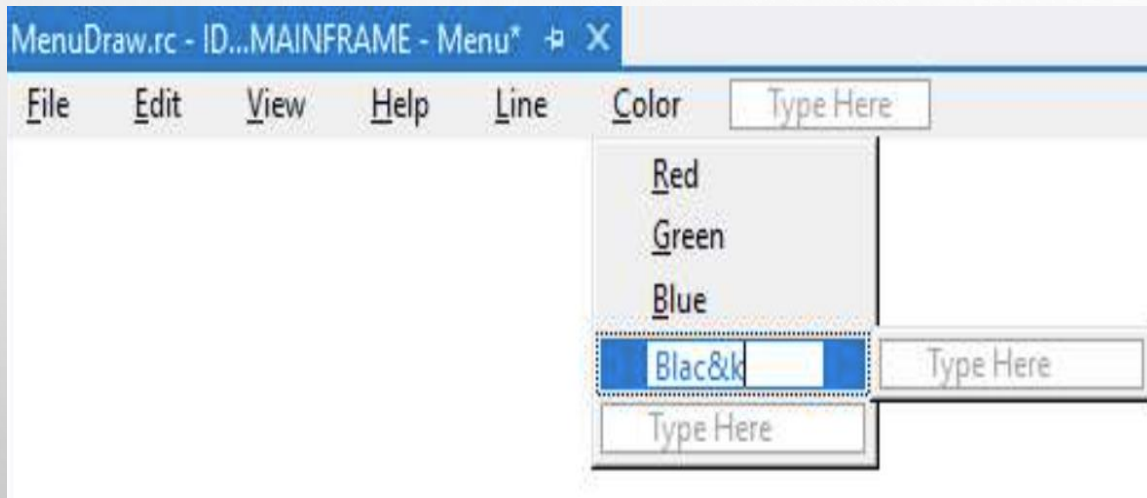
DIALOG

Adding an Item to the Menu Bar





Add Menu Items for Colors



Add data members in CMenuDrawView: → write in the .h

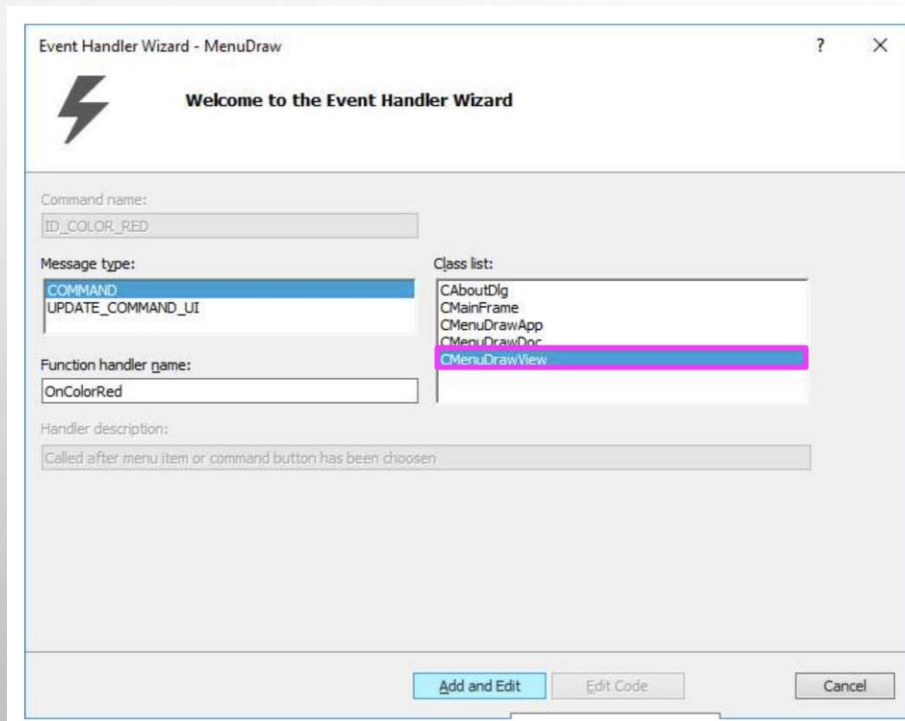
- COLORREF m_Color;
- CPen m_SolidPen;

Define 4 macros in CMenuDrawView.h: → write it at the beginning of the .h

- #define RED RGB(255, 0, 0)
- #define GREEN RGB(0, 255, 0)
- #define BLUE RGB(0, 0, 255)
- #define BLACK RGB(0, 0, 0)

Initialize in the constructor:

- m_Color = BLACK;
- m_SolidPen.CreatePen(PS_SOLID, 1, BLACK);

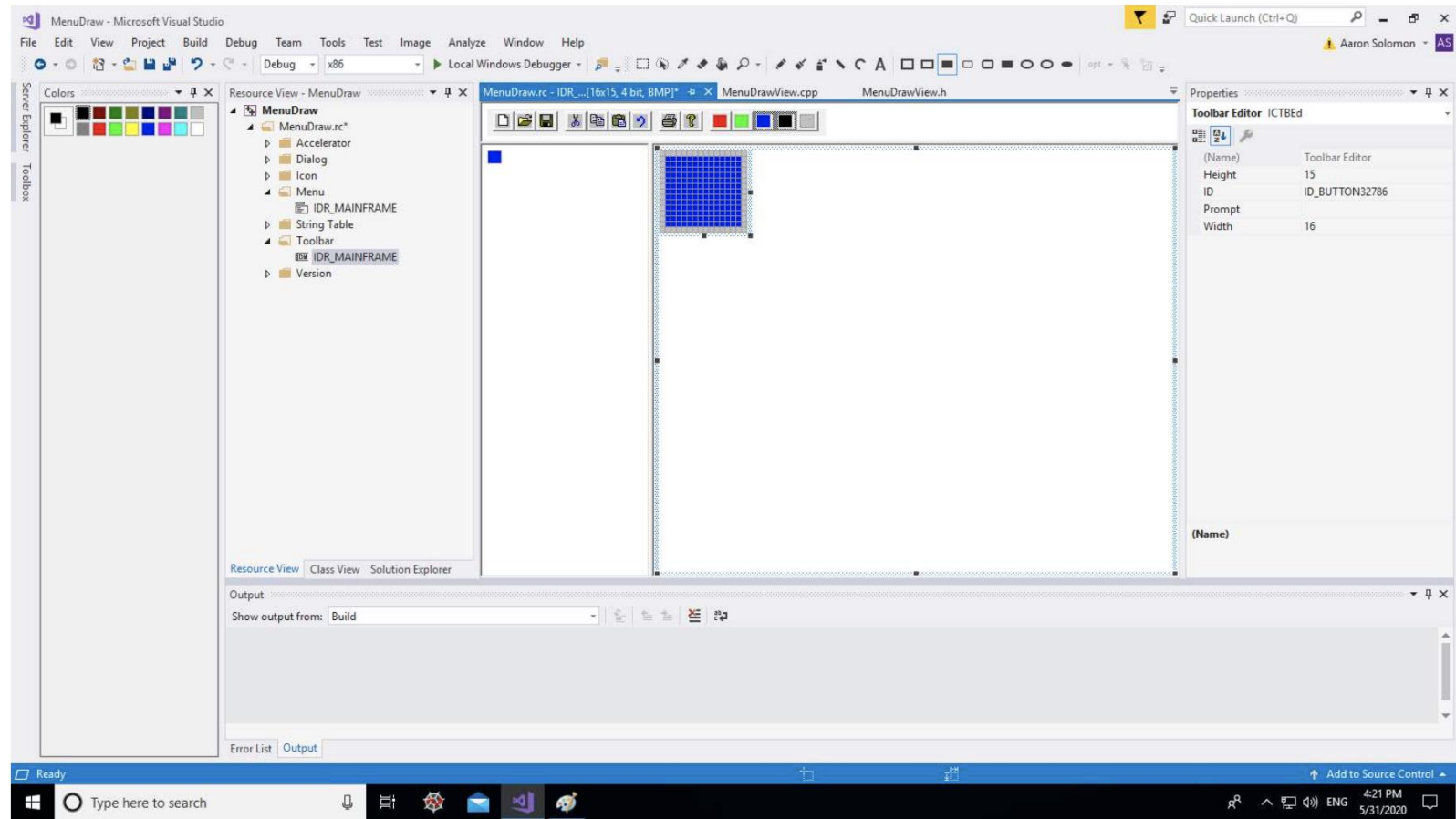


```
void CMenuDrawView::OnColorRed()
{
    m_Color = RED;
    ResetPens();
}
```

```
void CMenuDrawView::ResetPens()
{
    m_SolidPen.DeleteObject();
    m_SolidPen.CreatePen(PS_SOLID, 1, m_Color);
}
```

```
void CMouseView::OnDrawTriangle()
{
    CClientDC aDC(this);
    CPoint p[3] = { CPoint(500, 100), CPoint(500, 200), CPoint(600, 150) };
    aDC.SelectObject(&m_SolidPen);
    aDC.Polygon(p, 3);
}
```

Toolbar Buttons



Change the Default ID

