

1번문제

1-1 o

1-2 x -드래그가 아니라 왼쪽마우스 클릭시 그려지게 만듦

1-3 o

1-4 o

1-5 x 십자무늬 못함. 나머지는 구현

view.cpp

```

1
2  // P2019305059_1View.cpp: CP20193050591View 클래스의 구현
3  //
4
5  #include "pch.h"
6  #include "framework.h"
7  // SHARED_HANDLERS는 미리 보기, 축소판 그림 및 검색 필터 처리기를 구현하는 ATL 프로젝트에서 정의할 수 있으며
8  // 해당 프로젝트와 문서 코드를 공유하도록 해 줍니다.
9  #ifndef SHARED_HANDLERS
10 #include "P2019305059_1.h"
11 #endif
12
13 #include "P2019305059_1Doc.h"
14 #include "P2019305059_1View.h"
15
16 #ifdef _DEBUG
17 #define new DEBUG_NEW
18 #endif
19
20
21 // CP20193050591View
22
23 IMPLEMENT_DYNCREATE(CP20193050591View, CView)
24
25 BEGIN_MESSAGE_MAP(CP20193050591View, CView)
26     // 표준 인쇄 명령입니다.
27     ON_COMMAND(ID_FILE_PRINT, &CView::OnFilePrint)
28     ON_COMMAND(ID_FILE_PRINT_DIRECT, &CView::OnFilePrint)
29     ON_COMMAND(ID_FILE_PRINT_PREVIEW, &CView::OnFilePrintPreview)
30     ON_WM_LBUTTONDOWN()
31     ON_COMMAND(ID_FIG_CIR, &CP20193050591View::OnFigCir)
32     ON_COMMAND(ID_FIG_RECT, &CP20193050591View::OnFigRect)
33     // ON_COMMAND(ID_COLOR_BLUE, &CP20193050591View::OnColorBlue)
34     // ON_COMMAND(ID_COLOR_RED, &CP20193050591View::OnColorRed)
35     // ON_COMMAND(ID_COLOR_YELLOW, &CP20193050591View::OnColorYellow)
36     //ON_WM_PAINT()
37     ON_COMMAND(ID_COLOR_PINK, &CP20193050591View::OnColorPink)
38     ON_COMMAND(ID_COLOR_PU, &CP20193050591View::OnColorPu)
39     ON_COMMAND(ID_COLOR_GREEN, &CP20193050591View::OnColorGreen)
40     END_MESSAGE_MAP()
41
42 // CP20193050591View 생성/소멸
43

```

```

43
44 CP20193050591View::CP20193050591View() noexcept
45 {
46     // TODO: 여기에 생성 코드를 추가합니다.
47 }
48
49
50 CP20193050591View::~CP20193050591View()
51 {
52 }
53
54 BOOL CP20193050591View::PreCreateWindow(CREATESTRUCT& cs)
55 {
56     // TODO: CREATESTRUCT cs를 수정하여 여기에서
57     // Window 클래스 또는 스타일을 수정합니다.
58
59     return CView::PreCreateWindow(cs);
60 }
61
62 // CP20193050591View 그리기
63
64 void CP20193050591View::OnDraw(CDC* pDC)
65 {
66     CP20193050591Doc* pDoc = GetDocument();
67     ASSERT_VALID(pDoc);
68     if (!pDoc)
69         return;
70
71     // TODO: 여기에 원시 데이터에 대한 그리기 코드를 추가합니다.
72     CRect rect;
73     GetClientRect(&rect);
74
75     CFont font, * pOldFont;
76     font.CreatePointFont(200, _T("Times New Roman"));
77     pOldFont = (CFont*)pDC->SelectObject(&font);
78
79     pDC->SetTextAlign(TA_CENTER);
80     pDC->SetTextColor(RED(0, 255, 0));
81     pDC->SetBkColor(RED(255, 255, 255));
82     pDC->TextOutW(rect.right / 2, 0, CString("2019305059 이현수"));
83
84     if (temp==TRUE) {
85         if (m_fig == 1) {

```

```

P2019305059_1View.cpp P2019305059_1View.h P2019305059_1Doc.h P2019305059_1Doc.cpp P20193050591.rc...AINFRAME - Menu
P2019305059_1 CP20193050591View OnDraw(CDC * p
82 pDC->TextOutW(rect.right / 2, 0, CString("2019305059 이현수"));
83
84 if (temp==TRUE) {
85     if (m_fig == 1) {
86         CPen pen1(PA_SOLID, 20, pDoc->ColorGet());
87         pDC->SelectObject(&pen1);
88         pDC->Ellipse(100, 100, 200, 200);
89         if (m_inColor == 0) {
90             CBrush brush(RED(255, 255, 255));
91             CBrush* pOldBrush = pDC->SelectObject(&brush);
92             pDC->Ellipse(100, 100, 200, 200);
93         }
94         else if (m_inColor == 1) {
95             CBrush brush(RED(255, 0, 127));
96             CBrush* pOldBrush = pDC->SelectObject(&brush);
97             pDC->Ellipse(100, 100, 200, 200);
98         }
99         else if (m_inColor == 2) {
100             CBrush brush(RED(95,0, 255));
101             CBrush* pOldBrush = pDC->SelectObject(&brush);
102             pDC->Ellipse(100, 100, 200, 200);
103         }
104         else if (m_inColor == 3) {
105             CBrush brush(RED(0, 255, 0));
106             CBrush* pOldBrush = pDC->SelectObject(&brush);
107             pDC->Ellipse(100, 100, 200, 200);
108         }
109     }
110     else {
111         CPen blackpen(PA_SOLID, 20, pDoc->ColorGet());
112         pDC->SelectObject(&blackpen);
113         pDC->Rectangle(100, 100, 200, 200);
114         if (m_inColor == 0) {
115             CBrush brush(RED(255, 255, 255));
116             CBrush* pOldBrush = pDC->SelectObject(&brush);
117             pDC->Rectangle(100, 100, 200, 200);
118         }
119         else if (m_inColor == 1) {
120             CBrush brush(RED(255, 0, 127));
121             CBrush* pOldBrush = pDC->SelectObject(&brush);
122             pDC->Rectangle(100, 100, 200, 200);
123         }
124     }

```

```

P2019305059_1View.cpp  P2019305059_1View.h  P2019305059_1Doc.h  P2019305059_1Doc.cpp  P20193050591.rc...AINFRAME - Menu
P2019305059_1  → CP20193050591View  OnDraw(CDC * p
121         CBrush brush(RGB(255, 0, 127));
122         CBrush* pOldBrush = pDC->SelectObject(&brush);
123         pDC->Rectangle(100, 100, 200, 200);
124     }
125     else if (m_inColor == 2) {
126         CBrush brush(RGB(95, 0, 255));
127         CBrush* pOldBrush = pDC->SelectObject(&brush);
128         pDC->Rectangle(100, 100, 200, 200);
129     }
130     else if (m_inColor == 3) {
131         CBrush brush(RGB(0, 255, 0));
132         CBrush* pOldBrush = pDC->SelectObject(&brush);
133         pDC->Rectangle(100, 100, 200, 200);
134     }
135     }
136     temp = FALSE;
137 }
138
139 }
140
141
142 // CP20193050591View 인쇄
143
144 BOOL CP20193050591View::OnPreparePrinting(CPrintInfo* pInfo)
145 {
146     // 기본적인 준비
147     return DoPreparePrinting(pInfo);
148 }
149
150 void CP20193050591View::OnBeginPrinting(CDC* /*pDC*/, CPrintInfo* /*pInfo*/)
151 {
152     // TODO: 인쇄하기 전에 추가 초기화 작업을 추가합니다.
153 }
154
155 void CP20193050591View::OnEndPrinting(CDC* /*pDC*/, CPrintInfo* /*pInfo*/)
156 {
157     // TODO: 인쇄 후 정리 작업을 추가합니다.
158 }
159
160
161 // CP20193050591View 진단
162
163 #ifdef _DEBUG
164 void CP20193050591View::AssertValid() const
165 {
166     CView::AssertValid();
167 }
168
169 void CP20193050591View::Dump(CDumpContext& dc) const
170 {
171     CView::Dump(dc);
172 }
173
174 CP20193050591Doc* CP20193050591View::GetDocument() const // 디버그되지 않은 버전은 인라인으로 지정됩니다.
175 {
176     ASSERT(m_pDocument->IsKindOf(RUNTIME_CLASS(CP20193050591Doc)));
177     return (CP20193050591Doc*)m_pDocument;
178 }
179 #endif // _DEBUG
180
181
182 // CP20193050591View 메시지 처리기
183
184
185 void CP20193050591View::OnLButtonDown(UINT nFlags, CPoint point)
186 {
187     // TODO: 여기에 메시지 처리기 코드를 추가 및/또는 기본값을 호출합니다.
188     temp = TRUE;
189     Invalidate();
190     CView::OnLButtonDown(nFlags, point);
191 }

```

```
P2019305059_1View.cpp  P2019305059_1View.h  P2019305059_1Doc.h  P2019305059_1Doc.cpp  P20193050591.rc...AINFRAME - Menu
P2019305059_1  CP20193050591View  OnDraw(CDC *pDC)

190     CView::OnLButtonDown(nFlags, point);
191 }
192
193
194 void CP20193050591View::OnFigCir()
195 {
196     // TODO: 여기에 명령 처리기 코드를 추가합니다.
197     m_fig = 1;
198     Invalidate();
199 }
200
201
202 void CP20193050591View::OnFigRect()
203 {
204     // TODO: 여기에 명령 처리기 코드를 추가합니다.
205     m_fig = 2;
206     Invalidate();
207 }
208
209
210
211 void CP20193050591View::OnPaint()
212 {
213     // CPaintDC dc(this); // device context for painting
214     // TODO: 여기에 메시지 처리기 코드를 추가합니다.
215     // 그리기 메시지에 대해서는 CView::OnPaint()을(를) 호출하지 마십시오.
216 }
217
218
219
220 void CP20193050591View::OnColorPink()
221 {
222     // TODO: 여기에 명령 처리기 코드를 추가합니다.
223     m_inColor = 1;
224     Invalidate();
225 }
226
227
228 void CP20193050591View::OnColorPu()
229 {
230     // TODO: 여기에 명령 처리기 코드를 추가합니다.
231     m_inColor = 2;
232     Invalidate();
233 }
```

```
P2019305059_1View.cpp  P2019305059_1View.h  P2019305059_1Doc.h  P2019305059_1Doc.cpp  P20193050591.rc...AINFRAME - Menu
P2019305059_1  CP20193050591View  OnDraw(CDC *pDC)

208
209
210
211 void CP20193050591View::OnPaint()
212 {
213     // CPaintDC dc(this); // device context for painting
214     // TODO: 여기에 메시지 처리기 코드를 추가합니다.
215     // 그리기 메시지에 대해서는 CView::OnPaint()을(를) 호출하지 마십시오.
216 }
217
218
219
220 void CP20193050591View::OnColorPink()
221 {
222     // TODO: 여기에 명령 처리기 코드를 추가합니다.
223     m_inColor = 1;
224     Invalidate();
225 }
226
227
228 void CP20193050591View::OnColorPu()
229 {
230     // TODO: 여기에 명령 처리기 코드를 추가합니다.
231     m_inColor = 2;
232     Invalidate();
233 }
234
235
236 void CP20193050591View::OnColorGreen()
237 {
238     // TODO: 여기에 명령 처리기 코드를 추가합니다.
239     m_inColor = 3;
240     Invalidate();
241 }
242
```

```
P2019305059_1View.cpp P2019305059_1View.h P2019305059_1Doc.h P2019305059_1Doc.cpp P2019305059_1.rc...AINFRAME - Menu
P2019305059_1 → CP20193050591Doc InitializeSearchC

148     }
149     #endif // _DEBUG
150
151     // CP20193050591Doc 명령
152 void CP20193050591Doc::ColorSet(COLORREF color) {
153     m_curColor = color;
154     UpdateAllViews(NULL);
155 }
156
157 COLORREF CP20193050591Doc::ColorGet() {
158     return m_curColor;
159 }
160
161 void CP20193050591Doc::OnColorRed()
162 {
163     // TODO: 여기에 명령 처리기 코드를 추가합니다.
164     ColorSet(RED);
165 }
166
167 void CP20193050591Doc::OnColorBlue()
168 {
169     // TODO: 여기에 명령 처리기 코드를 추가합니다.
170     ColorSet(BLUE);
171 }
172
173 void CP20193050591Doc::OnColorYellow()
174 {
175     // TODO: 여기에 명령 처리기 코드를 추가합니다.
176     ColorSet(YELLOW);
177 }
178
179 void CP20193050591Doc::OnColorGreen()
180 {
181     // TODO: 여기에 명령 처리기 코드를 추가합니다.
182 }
183
184 void CP20193050591Doc::OnColorPink()
185 {
186     // TODO: 여기에 명령 처리기 코드를 추가합니다.
187 }
```

2번문제

2-1 ○

2-2 ○

2-3 ○

2-4 ○

2-5 ○

2-6 ○

2-7 ○

2-8 ○

CSlider.cpp

```
pch.cpp  CSlider.cpp  CSlider.h  P2019305059_2Dlg.cpp  P2019305059_2Dlg.h  bitmap1.bmp - Bi...[48x48, 4비트, BMP]  P2019305059_2.cpp
P2019305059_2  → CSlider  OnBnClickedOk()

1  // CSlider.cpp: 구현 파일
2  //
3
4  #include "pch.h"
5  #include "P2019305059_2.h"
6  #include "CSlider.h"
7  #include "afxdialogex.h"
8
9
10 // CSlider 대화 상자
11
12 IMPLEMENT_DYNAMIC(CSlider, CDialogEx)
13
14 CSlider::CSlider(CWnd* pParent /*=nullptr*/)
15     : CDialogEx(IDD_DIALOG1, pParent)
16     , m_nB(0)
17     , m_nG(0)
18     , m_nR(0)
19     {
20     }
21
22
23 CSlider::~CSlider()
24 {
25 }
26
27 void CSlider::DoDataExchange(CDataExchange* pDX)
28 {
29     CDialogEx::DoDataExchange(pDX);
30     // DDX_Control(pDX, IDC_SLIDER_G, m_slidR);
31     DDX_Control(pDX, IDC_SLIDER_B, m_slidB);
32     DDX_Control(pDX, IDC_SLIDER_G, m_slidG);
33     DDX_Control(pDX, IDC_SLIDER_R, m_slidR);
34     DDX_Text(pDX, IDC_EDIT_B, m_nB);
35     DDX_Text(pDX, IDC_EDIT_G, m_nG);
36     DDX_Text(pDX, IDC_EDIT_R, m_nR);
37 }
38
39
40 BEGIN_MESSAGE_MAP(CSlider, CDialogEx)
41     ON_WM_PAINT()
42     ON_WM_HSCROLL()
43     ON_BN_CLICKED(IDOK, &CSlider::OnBnClickedOk)
44 END_MESSAGE_MAP()
45
46
47 // CSlider 메시지 처리기
48
```

```
pch.cpp CSlder.cpp x CSlder.h P2019305059_2Dlg.cpp P2019305059_2Dlg.h bitmap1.bmp - Bi...[48x48, 4비트, BMP] P2019305059_2.cpp
P2019305059_2 CSlder OnBnClickedOk()
46
47 // CSlder 메시지 처리기
48
49
50 BOOL CSlder::OnInitDialog()
51 {
52     CDialogEx::OnInitDialog();
53
54     // TODO: 여기에 추가 초기화 작업을 추가합니다.
55     m_slidB.SetRange(0, 255);
56     m_slidG.SetRange(0, 255);
57     m_slidR.SetRange(0, 255);
58     return TRUE; // return TRUE unless you set the focus to a control
59     // 예외: OCX 속성 페이지는 FALSE를 반환해야 합니다.
60 }
61
62
63 void CSlder::OnPaint()
64 {
65     CPaintDC dc(this); // device context for painting
66     // TODO: 여기에 메시지 처리기 코드를 추가합니다.
67     // 그리기 메시지에 대해서는 CDialogEx::OnPaint()을(를) 호출하지 마십시오.
68     CRect rect;
69     CClientDC rgbdc(GetDlgItem(IDC_STATIC));
70     CStatic* pSRGB = (CStatic*)GetDlgItem(IDC_STATIC);
71     pSRGB->GetClientRect(rect);
72     rgbdc.FillSolidRect(rect, m_cRGB);
73     pSRGB->ValidateRect(rect);
74 }
75
76
77 void CSlder::OnHScroll(UINT nSBCode, UINT nPos, CScrollBar* pScrollBar)
78 {
79     // TODO: 여기에 메시지 처리기 코드를 추가 및/또는 기본값을 호출합니다.
80     CRect rect;
81     GetDlgItem(IDC_STATIC)->GetWindowRect(&rect);
82     ScreenToClient(&rect);
83     int nR = m_slidR.GetPos();
84     int nG = m_slidG.GetPos();
85     int nB = m_slidB.GetPos();
86
87     if ((pScrollBar == (CScrollBar*)&m_slidR) || (pScrollBar == (CScrollBar*)&m_slidG) || (pScrollBar == (CScrollBar*)&m_slidB)) {
88         int nPosR = m_slidR.GetPos();
89         int nPosG = m_slidG.GetPos();
90         int nPosB = m_slidB.GetPos();
91
92         m_nR = nPosR;
93         m_nG = nPosG;
```

```
pch.cpp CSlder.cpp x CSlder.h P2019305059_2Dlg.cpp P2019305059_2Dlg.h bitmap1.bmp - Bi...[48x48, 4비트, BMP] P2019305059_2.cpp
P2019305059_2 CSlder OnBnClickedOk()
70     CStatic* pSRGB = (CStatic*)GetDlgItem(IDC_STATIC);
71     pSRGB->GetClientRect(rect);
72     rgbdc.FillSolidRect(rect, m_cRGB);
73     pSRGB->ValidateRect(rect);
74 }
75
76
77 void CSlder::OnHScroll(UINT nSBCode, UINT nPos, CScrollBar* pScrollBar)
78 {
79     // TODO: 여기에 메시지 처리기 코드를 추가 및/또는 기본값을 호출합니다.
80     CRect rect;
81     GetDlgItem(IDC_STATIC)->GetWindowRect(&rect);
82     ScreenToClient(&rect);
83     int nR = m_slidR.GetPos();
84     int nG = m_slidG.GetPos();
85     int nB = m_slidB.GetPos();
86
87     if ((pScrollBar == (CScrollBar*)&m_slidR) || (pScrollBar == (CScrollBar*)&m_slidG) || (pScrollBar == (CScrollBar*)&m_slidB)) {
88         int nPosR = m_slidR.GetPos();
89         int nPosG = m_slidG.GetPos();
90         int nPosB = m_slidB.GetPos();
91
92         m_nR = nPosR;
93         m_nG = nPosG;
94         m_nB = nPosB;
95
96         m_cRGB = RGB(m_nR, m_nG, m_nB);
97         UpdateData(FALSE);
98         InvalidateRect(&rect);
99     }
100     else {
101         CDialogEx::OnHScroll(nSBCode, nPos, pScrollBar);
102     }
103 }
104
105
106 void CSlder::OnBnClickedOk()
107 {
108     // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
109     Invalidate();
110     CDialogEx::OnOK();
111 }
112
```

dlg.cpp

```
pch.cpp  CSlider.cpp  CSlider.h  P2019305059_2Dlg.cpp  P2019305059_2Dlg.h  bitmap1.bmp - BI...[48x48, 4비트, BMP]  P2019305059_2.cpp
P2019305059_2  → CP20193050592Dlg  OnClickedButtonSld()

1
2  // P2019305059_2Dlg.cpp: 구현 파일
3  //
4
5  #include "pch.h"
6  #include "framework.h"
7  #include "P2019305059_2.h"
8  #include "P2019305059_2Dlg.h"
9  #include "afxdialogex.h"
10
11 #ifdef _DEBUG
12 #define new DEBUG_NEW
13 #endif
14
15 // 응용 프로그램 정보에 사용되는 CAboutDlg 대화 상자입니다.
16
17 class CAboutDlg : public CDialogEx
18 {
19 public:
20     CAboutDlg();
21
22     // 대화 상자 데이터입니다.
23 #ifdef AFX_DESIGN_TIME
24     enum { IDD = IDD_ABOUTBOX };
25 #endif
26
27     protected:
28     virtual void DoDataExchange(CDataExchange* pDX);    // DDX/DDV 지원입니다.
29
30     // 구현입니다.
31     protected:
32     DECLARE_MESSAGE_MAP()
33 };
34
35 CAboutDlg::CAboutDlg() : CDialogEx(IDD_ABOUTBOX)
36 {
37 }
38
39 void CAboutDlg::DoDataExchange(CDataExchange* pDX)
40 {
41     CDialogEx::DoDataExchange(pDX);
42 }
43
44 BEGIN_MESSAGE_MAP(CAboutDlg, CDialogEx)
45 END_MESSAGE_MAP()
46
47
48
49
50
51
52
53 CP20193050592Dlg::CP20193050592Dlg(CWnd* pParent /*=nullptr*/)
54 : CDialogEx(IDD_P2019305059_2_DIALOG, pParent)
55 {
56     m_hIcon = AfxGetApp()->LoadIcon(IDR_MAINFRAME);
57 }
58
59 void CP20193050592Dlg::DoDataExchange(CDataExchange* pDX)
60 {
61     CDialogEx::DoDataExchange(pDX);
62 }
63
64 BEGIN_MESSAGE_MAP(CP20193050592Dlg, CDialogEx)
65     ON_WM_SYSCOMMAND()
66     ON_WM_PAINT()
67     ON_WM_QUERYDRAGICON()
68     ON_WM_TIMER()
69     ON_WM_SIZE()
70     ON_BN_CLICKED(IDC_BUTTON_EXIT, &CP20193050592Dlg::OnClickedButtonExit)
71     ON_BN_CLICKED(IDC_BUTTON_NOTE, &CP20193050592Dlg::OnClickedButtonNote)
72     ON_BN_CLICKED(IDC_BUTTON_SLD, &CP20193050592Dlg::OnClickedButtonSld)
73 END_MESSAGE_MAP()
74
75 // CP20193050592Dlg 메시지 처리기
76
77 BOOL CP20193050592Dlg::OnInitDialog()
78 {
79     CDialogEx::OnInitDialog();
80
81     // 시스템 메뉴에 "정보..." 메뉴 항목을 추가합니다.
82
83     // IDM_ABOUTBOX는 시스템 명령 범위에 있어야 합니다.
84     ASSERT((IDM_ABOUTBOX & 0xFFF0) == IDM_ABOUTBOX);
85     ASSERT(IDM_ABOUTBOX < 0xF000);
86
87     htimer = SetTimer(1, 1000, NULL);
88     CMenu* pSysMenu = GetSystemMenu(FALSE);
89     if (pSysMenu != nullptr)
90     {
91         BOOL bNameValid;
92         CString strAboutMenu;
```



```

91     {
92         BOOL bNameValid;
93         CString strAboutMenu;
94         bNameValid = strAboutMenu.LoadString(IDS_ABOUTBOX);
95         ASSERT(bNameValid);
96         if (!strAboutMenu.IsEmpty())
97         {
98             pSysMenu->AppendMenu(MF_SEPARATOR);
99             pSysMenu->AppendMenu(MF_STRING, IDM_ABOUTBOX, strAboutMenu);
100         }
101     }
102
103     // 이 대화 상자의 아이콘을 설정합니다. 응용 프로그램의 주 창이 대화 상자가 아닐 경우에는
104     // 프레임워크가 이 작업을 자동으로 수행합니다.
105     SetIcon(m_hIcon, TRUE); // 큰 아이콘을 설정합니다.
106     SetIcon(m_hIcon, FALSE); // 작은 아이콘을 설정합니다.
107
108     // TODO: 여기에 추가 초기화 작업을 추가합니다.
109     nR = 255;
110     nG = 228;
111     nB = 0;
112     return TRUE; // 포커스를 컨트롤에 설정하지 않으면 TRUE를 반환합니다.
113 }
114
115 void CP20193050592Dlg::OnSysCommand(UINT nID, LPARAM lParam)
116 {
117     if ((nID & 0xFFF0) == IDM_ABOUTBOX)
118     {
119         CAboutDlg dlgAbout;
120         dlgAbout.DoModal();
121     }
122     else
123     {
124         CDialogEx::OnSysCommand(nID, lParam);
125     }
126 }
127
128 // 대화 상자에 최소화 단추를 추가할 경우 아이콘을 그리려면
129 // 아래 코드가 필요합니다. 문서/뷰 모델을 사용하는 MFC 애플리케이션의 경우에는
130 // 프레임워크에서 이 작업을 자동으로 수행합니다.
131
132 void CP20193050592Dlg::OnPaint()
133 {
134     if (!IsIconic())
135     {
136         CPaintDC dc(this); // 그리기를 위한 디바이스 컨텍스트입니다.
137
138         SendMessage(WM_ICONERASEBKGND, reinterpret_cast<LPARAM>(dc.GetSafeHdc()), 0);
139     }
140 }
141
142 // 클라이언트 사각형에서 아이콘을 가운데에 맞춥니다.
143 int cxIcon = GetSystemMetrics(SM_CXICON);
144 int cyIcon = GetSystemMetrics(SM_CYICON);
145 CRect rect;
146 GetClientRect(&rect);
147 int x = (rect.Width() - cxIcon + 1) / 2;
148 int y = (rect.Height() - cyIcon + 1) / 2;
149
150 // 아이콘을 그립니다.
151 dc.DrawIcon(x, y, m_hIcon);
152 }
153 else
154 {
155     CPaintDC dc(this);
156     CPen pen(PS_SOLID, 10, RGB(nR,nG,nB));
157     dc.SelectObject(&pen);
158     dc.Rectangle(0, 0, m_nWidth / 2, 3 * m_nHeight / 4);
159
160     HBITMAP hBitmap = ::LoadBitmap(m_hInstance, MAKEINTRESOURCE(IDB_BITMAP1));
161
162     HDC hMemDC = ::CreateCompatibleDC(NULL);
163     SelectObject(hMemDC, hBitmap);
164     ::StretchBlt(dc.m_hDC, 0+10, 0+10, m_nWidth/2-10, 3*m_nHeight/4-10, hMemDC, 0, 0, 48, 48, SRCCOPY);
165
166     CDialogEx::OnPaint();
167 }
168
169 // 사용자가 최소화된 창을 끄는 동안에 커서가 표시되도록 시스템에서
170 // 이 함수를 호출합니다.
171 void CP20193050592Dlg::OnQueryDragIcon()
172 {
173     return static_cast<HOURSOP>(m_hIcon);
174 }
175
176
177
178 void CP20193050592Dlg::OnTimer(UINT_PTR nIDEvent)
179 {
180     // TODO: 여기에 메시지 처리기 코드를 추가 및/또는 기본값을 호출합니다.
181     CTime now = CTime::GetCurrentTime();
182     CString strTime;
183
184 }
  
```

```

136     CPaintDC dc(this); // 그리기를 위한 디바이스 컨텍스트입니다.
137
138     SendMessage(WM_ICONERASEBKGND, reinterpret_cast<LPARAM>(dc.GetSafeHdc()), 0);
139
140     // 클라이언트 사각형에서 아이콘을 가운데에 맞춥니다.
141     int cxIcon = GetSystemMetrics(SM_CXICON);
142     int cyIcon = GetSystemMetrics(SM_CYICON);
143     CRect rect;
144     GetClientRect(&rect);
145     int x = (rect.Width() - cxIcon + 1) / 2;
146     int y = (rect.Height() - cyIcon + 1) / 2;
147
148     // 아이콘을 그립니다.
149     dc.DrawIcon(x, y, m_hIcon);
150 }
151 else
152 {
153     CPaintDC dc(this);
154     CPen pen(PS_SOLID, 10, RGB(nR,nG,nB));
155     dc.SelectObject(&pen);
156     dc.Rectangle(0, 0, m_nWidth / 2, 3 * m_nHeight / 4);
157
158     HBITMAP hBitmap = ::LoadBitmap(m_hInstance, MAKEINTRESOURCE(IDB_BITMAP1));
159
160     HDC hMemDC = ::CreateCompatibleDC(NULL);
161     SelectObject(hMemDC, hBitmap);
162     ::StretchBlt(dc.m_hDC, 0+10, 0+10, m_nWidth/2-10, 3*m_nHeight/4-10, hMemDC, 0, 0, 48, 48, SRCCOPY);
163
164     CDialogEx::OnPaint();
165 }
166
167
168 // 사용자가 최소화된 창을 끄는 동안에 커서가 표시되도록 시스템에서
169 // 이 함수를 호출합니다.
170 void CP20193050592Dlg::OnQueryDragIcon()
171 {
172     return static_cast<HOURSOP>(m_hIcon);
173 }
174
175
176
177
178 void CP20193050592Dlg::OnTimer(UINT_PTR nIDEvent)
179 {
180     // TODO: 여기에 메시지 처리기 코드를 추가 및/또는 기본값을 호출합니다.
181     CTime now = CTime::GetCurrentTime();
182     CString strTime;
183
184 }
  
```

pch.cpp	CSlider.cpp	CSlider.h	P2019305059_2Dlg.cpp	P2019305059_2Dlg.h	bitmap1.bmp - Bi...[48x48, 4비트, BMP]	P2019305059_2.cpp
---------	-------------	-----------	----------------------	--------------------	--------------------------------------	-------------------

P2019305059_2 → CP20193050592Dlg OnClickedButtonSld0

```
181 CTime gct = CTime::GetCurrentTime();
182 CString strTime;
183
184 strTime.Format(_T("%d 시 %d 분 %d 초"), gct.GetHour(), gct.GetMinute(), gct.GetSecond());
185 GetDlgItem(IDC_STATIC_TIME)->SetWindowText((LPCTSTR)strTime);
186
187 CDialogEx::OnTimer(nIDEvent);
188 }
189
190
191 void CP20193050592Dlg::OnSize(UINT nType, int cx, int cy)
192 {
193     CDialogEx::OnSize(nType, cx, cy);
194
195     // TODO: 여기에 메시지 처리기 코드를 추가합니다.
196     m_nWidth = cx;
197     m_nHeight=cy;
198     Invalidate();
199 }
200
201
202 void CP20193050592Dlg::OnClickedButtonExit()
203 {
204     // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
205     OnOK();
206 }
207
208
209 void CP20193050592Dlg::OnClickedButtonNote()
210 {
211     // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
212     system("notepad.exe");
213 }
214
215
216 void CP20193050592Dlg::OnClickedButtonSld()
217 {
218     // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
219     m_slide.DoModal();
220     nR = m_slide.m_nR;
221     nG = m_slide.m_nG;
222     nB = m_slide.m_nB;
223     Invalidate();
224 }
225
```