

```
1 // CSmartColourOps.cpp : implementation file
2 //
3
4 #include "pch.h"
5 #include "DesignArk.h"
6 #include "afxdialogex.h"
7 #include "CSmartColourStatements.h"
8
9
10 // CSmartColourOps dialog
11
12 IMPLEMENT_DYNAMIC(CSmartColourStatements, CMFCPropertyPage)
13
14 CSmartColourStatements::CSmartColourStatements(std::vector<CString>
    syntax_string, COLORREF syntax_colour, std::vector<CString>
    statement_string, COLORREF statement_colour, CWnd* pParent/* =
    nullptr*/)
15     : CMFCPropertyPage(IDD_SMARTCOLOUR_OPS /*, pParent*/)
16     , m_Listbox_Statements_syntax_control()
17     , m_Listbox_Statements_statments_control()
18 {
19     this->m_Strings_Statements_syntax_var = syntax_string;
20     this->m_Colour_Statements_syntax_var = syntax_colour;
21
22     this->m_Strings_Statements_statements_var = statement_string;
23     this->m_Colour_Statements_statements_var = statement_colour;
24 }
25
26 CSmartColourStatements::~CSmartColourStatements()
27 {
28 }
29
30 void CSmartColourStatements::DoDataExchange(CDataExchange* pDX)
31 {
32     CMFCPropertyPage::DoDataExchange(pDX);
33     DDX_Control(pDX, IDC_MFCVSLISTBOX_OPS,
34         m_Listbox_Statements_statments_control);
35     DDX_Control(pDX, IDC_COMBO_COLOUR_OPS,
36         m_Combo_Statements_statments_control);
37     DDX_Control(pDX, IDC_MFCVSLISTBOX_OPS2,
38         m_Listbox_Statements_syntax_control);
39     DDX_Control(pDX, IDC_COMBO_COLOUR_OPS2,
40         m_Combo_Statements_syntax_control);
41 }
42
43
44 BOOL CSmartColourStatements::OnInitDialog()
45 {
46     CDialog::OnInitDialog();
47
48     for (auto& it : this->m_Strings_Statements_syntax_var) {
49         this->m_Listbox_Statements_syntax_control.AddItem(it);
50     }
51
52     this->m_Combo_Statements_syntax_control.SetColor(this->m_Colour_Statements_syntax_var);
53 }
```

```
49
50     for (auto& it : this->m_Strings_Statements_statements_var) {
51         this->m_Listbox_Statements_statments_control.AddItem(it);
52     }
53
54     this->m_Combo_Statements_statments_control.SetColor(this->m_Colour_Statements_statements_var);
55
56     return 0;
57 }
58
59 BOOL CSmartColourStatements::OnKillActive()
60 {
61     // Syntax
62     this->m_Strings_Statements_syntax_var.resize(this->m_Listbox_Statements_syntax_control.GetCount());
63
64     for (int i = 0; i < this->m_Listbox_Statements_syntax_control.GetCount(); i++) {
65         this->m_Strings_Statements_syntax_var[i] = this->m_Listbox_Statements_syntax_control.GetItemText(i);
66     }
67
68     // Insertion Sort
69     for (int i = 1; i < this->m_Strings_Statements_syntax_var.size(); i++) {
70         CString currentValue = this->m_Strings_Statements_syntax_var[i];
71         int pos = i;
72
73         while (pos > 0 && this->m_Strings_Statements_syntax_var[pos - 1] > currentValue) {
74             this->m_Strings_Statements_syntax_var[pos] = this->m_Strings_Statements_syntax_var[pos - 1];
75             pos--;
76         }
77         this->m_Strings_Statements_syntax_var[pos] = currentValue;
78     }
79
80     this->m_Colour_Statements_syntax_var = this->m_Combo_Statements_syntax_control.GetColor();
81
82     // Statements
83     this->m_Strings_Statements_statements_var.resize(this->m_Listbox_Statements_statments_control.GetCount());
84
85     for (int i = 0; i < this->m_Listbox_Statements_statments_control.GetCount(); i++) {
86         this->m_Strings_Statements_statements_var[i] = this->m_Listbox_Statements_statments_control.GetItemText(i);
87     }
88
89     // Insertion Sort
90     for (int i = 1; i < this->m_Strings_Statements_statements_var.size(); i++) {
91         CString currentValue = this-
```

```
        >m_Strings_Statements_statements_var[i];
92     int pos = i;
93
94     while (pos > 0 && this->m_Strings_Statements_statements_var
95            [pos - 1] > currentValue) {
96         this->m_Strings_Statements_statements_var[pos] = this-
97             >m_Strings_Statements_statements_var[pos - 1];
98         pos--;
99     }
100
101     this->m_Colour_Statements_statements_var = this-
102         >m_Combo_Statements_statements_control.GetColor();
103
104     return CMFCPropertyPage::OnKillActive();
105 }
106
107
108 BEGIN_MESSAGE_MAP(CSmartColourStatements, CMFCPropertyPage)
109 END_MESSAGE_MAP()
110
111
112 // CSmartColourOps message handlers
113
```