```
...D v21 Study\SDK\Samples\C++\vc3\Step12\MenuBase.cpp
```

```
1
```

```
#include "StdAfx.h"
 2 #include "MenuBase.h"
   extern int LibMessage(LPCTSTR str, int flags = MB_OK);
 5
  void MenuBase::initCoordinates(size_t id, size_t id_x, size_t id_y,
 6
     size_t id_z) {
 7
       BSTR name = _T("Координаты");
       this->coordinates = this->collection->Add
 8
          (ControlTypeEnum::ksControlPoint3D);
 9
       this->coordinates->Name = name;
10
       this->coordinates->Hint = name;
       this->coordinates->Tips = name;
11
12
       this->coordinates->Id = id;
13
       IPropertyEditPtr X = this->coordinates->GetCoordinate(0);
       IPropertyEditPtr Y = this->coordinates->GetCoordinate(1);
14
15
       IPropertyEditPtr Z = this->coordinates->GetCoordinate(2);
16
       X->Id = id_x;
17
       X->Value = this->getModel()->coordinates.X;
18
       X->CheckState = CheckStateEnum::ksCheckFixed;
19
20
       Y->Id = id_y;
21
22
       Y->Value = this->getModel()->coordinates.Y;
       Y->CheckState = CheckStateEnum::ksCheckFixed;
23
24
       Z->Id = id_z;
25
26
       Z->Value = this->getModel()->coordinates.Z;
       Z->CheckState = CheckStateEnum::ksCheckFixed;
27
28 }
29
   void MenuBase::initSurfaceType(size_t id) {
30
       BSTR name = _T("Тип основания");
31
32
       this->surfaceType = this->collection->Add
          (ControlTypeEnum::ksControlListStr);
33
       this->surfaceType->Name = name;
34
       this->surfaceType->Hint = name;
35
       this->surfaceType->Tips = name;
36
       this->surfaceType->Id = id;
37
       this->surfaceType->ReadOnly = true;
38
       this->surfaceType->Add(_T("Kpyr"));
39
       this->surfaceType->Add(_T("Треугольник"));
40
       this->surfaceType->Add(_T("Квадрат"));
       this->surfaceType->Add(_T("Пятиугольник"));
41
42
       this->surfaceType->Add(_T("Шестиугольник"));
       this->surfaceType->Add(_T("Тригон"));
43
       this->surfaceType->Add(_T("Pom6 35"));
44
       this->surfaceType->Add(_T("Pom6 55"));
45
       this->surfaceType->Add(_T("Pom6 80"));
46
       this->surfaceType->SetCurrentByIndex(this->getModel()-
47
         >surfaceType);
48 }
49
   void MenuBase::initSizeType(size_t id) {
50
       BSTR name = _T("Тип размера");
51
```

```
...D v21 Study\SDK\Samples\C++\vc3\Step12\MenuBase.cpp
52
        this->sizeType = this->collection->Add
          (ControlTypeEnum::ksControlListStr);
53
        this->sizeType->Name = name;
54
        this->sizeType->Hint = name;
55
        this->sizeType->Tips = name;
        this->sizeType->Id = id;
56
57
        this->sizeType->ReadOnly = true;
        this->sizeType->Add(_T("По радиусу описанной окружности"));
58
        this->sizeType->Add(_T("По радиусу вписанной окружности"));
59
60
        this->sizeType->Add(_T("По длине стороны"));
        this->sizeType->SetCurrentByIndex(this->getModel()->sizeType);
61
62
    }
63
    void MenuBase::initSize(size_t id) {
64
65
        BSTR name = _T("Величина размера");
        this->size = this->collection->Add
66
                                                                               P
          (ControlTypeEnum::ksControlEditLength);
67
        this->size->Name = name;
68
        this->size->Hint = name;
69
        this->size->Tips = name;
        this->size->Id = id;
70
71
        this->size->Value = this->getModel()->size;
    }
72
73
    void MenuBase::initHeight(size_t id) {
74
75
        BSTR name = _T("Высота");
76
        this->height = this->collection->Add
                                                                               P
          (ControlTypeEnum::ksControlEditLength);
77
        this->height->Name = name;
78
        this->height->Hint = name;
79
        this->height->Tips = name;
80
        this->height->Id = id;
81
        this->height->Value = this->getModel()->height;
    }
82
83
    void MenuBase::initHasHole(size_t id) {
84
        BSTR name = _T("Есть отверстие");
85
86
        this->hasHole = this->collection->Add
                                                                               P
          (ControlTypeEnum::ksControlTwinSwitcher);
87
        this->hasHole->Name = name;
88
        this->hasHole->Hint = name;
89
        this->hasHole->Tips = name;
        this->hasHole->Label2 = _T("Нет отверстия");
90
91
        this->hasHole->Id = id;
        this->hasHole->Value = (this->getModel()->hasHole ? 1 : 2);
92
    }
93
94
    void MenuBase::initHoleRadius(size_t id) {
95
96
        BSTR name = _T("Радиус отверстия");
97
        this->holeRadius = this->collection->Add
          (ControlTypeEnum::ksControlEditLength);
98
        this->holeRadius->Name = name;
99
        this->holeRadius->Hint = name;
```

100

101

this->holeRadius->Tips = name;

this->holeRadius->Id = id;

```
...D v21 Study\SDK\Samples\C++\vc3\Step12\MenuBase.cpp
                                                                               3
102
        this->holeRadius->Value = this->getModel()->holeRadius;
103 }
104
    void MenuBase::initAngleAlpha(size_t id) {
105
        BSTR name = _T("Угол наклона α");
106
107
        this->angleAlpha = this->collection->Add
                                                                               P
          (ControlTypeEnum::ksControlEditAngle);
        this->angleAlpha->Name = name;
108
109
        this->angleAlpha->Hint = name;
110
        this->angleAlpha->Tips = name;
111
        this->angleAlpha->Id = id;
112
        this->angleAlpha->Value = this->getModel()->angleAlpha;
    }
113
114
    void MenuBase::initRoundingRadius(size_t id) {
115
        BSTR name = _T("Paguyc скругления");
116
117
        this->roundingRadius = this->collection->Add
                                                                               P
          (ControlTypeEnum::ksControlEditLength);
118
        this->roundingRadius->Name = name;
        this->roundingRadius->Hint = name;
119
120
        this->roundingRadius->Tips = name;
121
        this->roundingRadius->Id = id;
122
        this->roundingRadius->Value = this->getModel()->roundingRadius;
123
124
125
    void MenuBase::initButton(size_t id) {
126
        BSTR name = _T("Применить");
127
        this->button = this->collection->Add
                                                                               P
          (ControlTypeEnum::ksControlTextButton);
128
        this->button->Name = name;
129
        this->button->Hint = name;
130
        this->button->Tips = name;
131
        this->button->Id = id;
132
        this->button->Value = this->getModel()->button;
133 }
134
    void MenuBase::setFieldName(IPropertyEditPtr field, BSTR name) {
135
136
        field->Name = name;
137
        field->Hint = name;
138
        field->Tips = name;
    }
139
140
    void MenuBase::setFieldActive(IPropertyEditPtr field) {
141
142
        field->ReadOnly = false;
143
        field->Enable = true;
144
        field->Visible = true;
    }
145
146
147
    void MenuBase::setFieldInactive(IPropertyEditPtr field) {
148
        field->ReadOnly = true;
149
        field->Enable = false;
150
        field->Visible = false;
151 }
```

152

153

154

void MenuBase::checkHasHole() {

if (this->hasHole->Value.intVal == 1) {

```
...D v21 Study\SDK\Samples\C++\vc3\Step12\MenuBase.cpp

this->setFieldActive(field_this->bologodian)
                this->setFieldActive(field: this->holeRadius);
           }
156
157
           else {
                this->setFieldInactive(field: this->holeRadius);
158
159
160 }
161
```