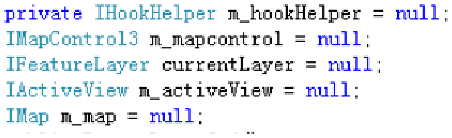
Chapter 15 Finish

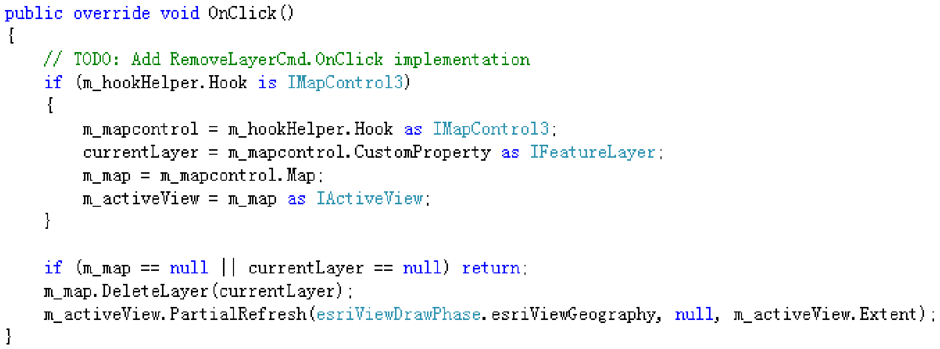
1. Add Base Command class: RemobeLayerCmd

- Class member:



```
private IHookHelper m_hookHelper = null;
IMapControl3 m_mapcontrol = null;
IFeatureLayer currentLayer = null;
IActiveView m_activeView = null;
IMap m_map = null;
```

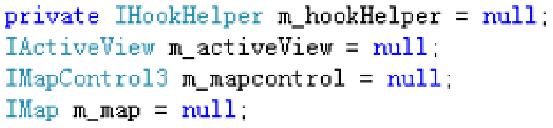
- OnClick:



```
public override void OnClick()
{
    // TODO: Add RemoveLayerCmd.OnClick implementation
    if (m_hookHelper.Hook is IMapControl3)
    {
        m_mapcontrol = m_hookHelper.Hook as IMapControl3;
        currentLayer = m_mapcontrol.CustomProperty as IFeatureLayer;
        m_map = m_mapcontrol.Map;
        m_activeView = m_map as IActiveView;
    }

    if (m_map == null || currentLayer == null) return;
    m_map.DeleteLayer(currentLayer);
    m_activeView.PartialRefresh(esriViewDrawPhase.esriViewGeography, null, m_activeView.Extent);
}
```

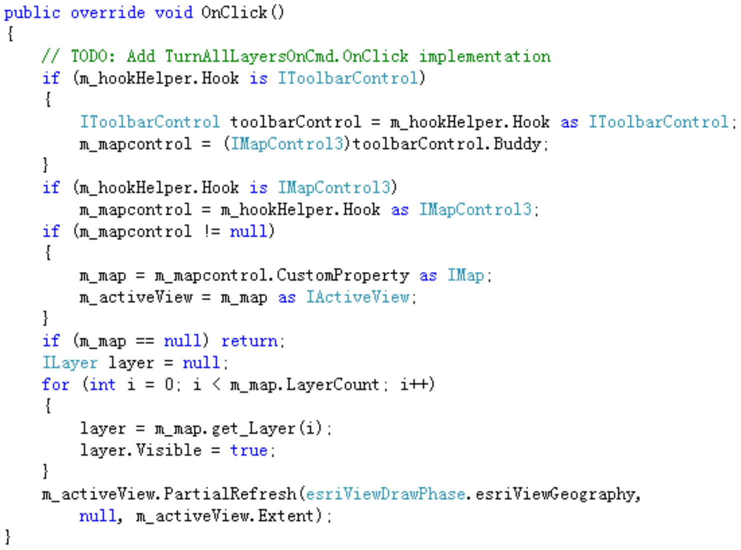
2. Add Base Command class: TurnAllLayersOnCmd

- Class member:



```
private IHookHelper m_hookHelper = null;
IActiveView m_activeView = null;
IMapControl3 m_mapcontrol = null;
IMap m_map = null;
```

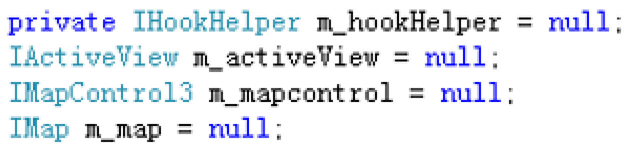
- OnClick:



```
public override void OnClick()
{
    // TODO: Add TurnAllLayersOnCmd.OnClick implementation
    if (m_hookHelper.Hook is IToolbarControl)
    {
        IToolbarControl toolbarControl = m_hookHelper.Hook as IToolbarControl;
        m_mapcontrol = (IMapControl3)toolbarControl.Buddy;
    }
    if (m_hookHelper.Hook is IMapControl3)
        m_mapcontrol = m_hookHelper.Hook as IMapControl3;
    if (m_mapcontrol != null)
    {
        m_map = m_mapcontrol.CustomProperty as IMap;
        m_activeView = m_map as IActiveView;
    }
    if (m_map == null) return;
    ILayer layer = null;
    for (int i = 0; i < m_map.LayerCount; i++)
    {
        layer = m_map.get_Layer(i);
        layer.Visible = true;
    }
    m_activeView.PartialRefresh(esriViewDrawPhase.esriViewGeography,
        null, m_activeView.Extent);
}
```

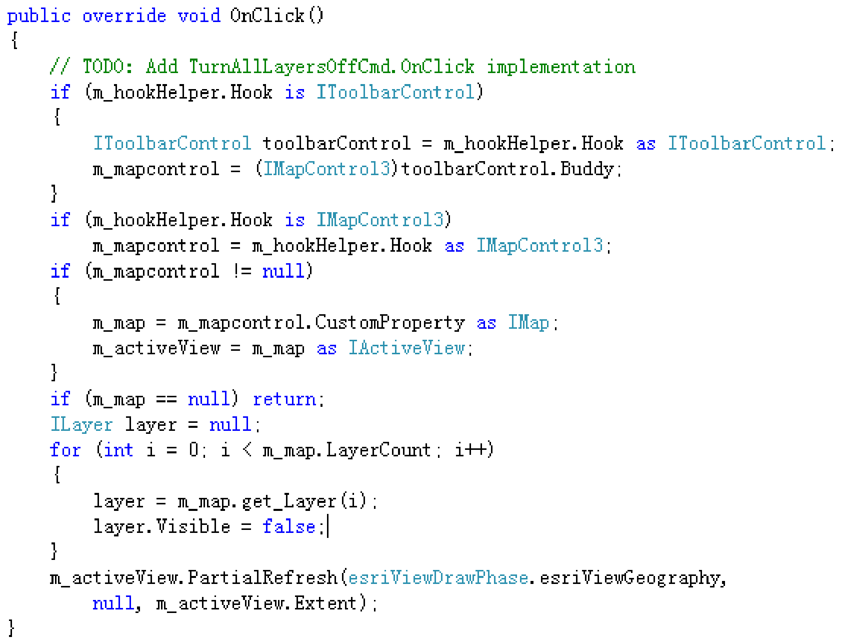
3. Add Base Command class: TurnAllLayersOffCmd

- Class member:



```
private IHookHelper m_hookHelper = null;
IActiveView m_activeView = null;
IMapControl3 m_mapcontrol = null;
IMap m_map = null;
```

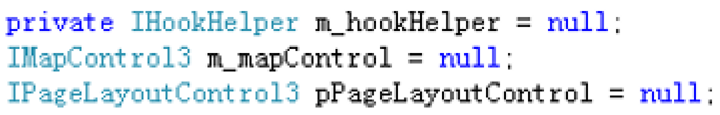
- OnClick:



```
public override void OnClick()
{
    // TODO: Add TurnAllLayersOffCmd.OnClick implementation
    if (m_hookHelper.Hook is IToolbarControl)
    {
        IToolbarControl toolbarControl = m_hookHelper.Hook as IToolbarControl;
        m_mapcontrol = (IMapControl3)toolbarControl.Buddy;
    }
    if (m_hookHelper.Hook is IMapControl3)
        m_mapcontrol = m_hookHelper.Hook as IMapControl3;
    if (m_mapcontrol != null)
    {
        m_map = m_mapcontrol.CustomProperty as IMap;
        m_activeView = m_map as IActiveView;
    }
    if (m_map == null) return;
    ILayer layer = null;
    for (int i = 0; i < m_map.LayerCount; i++)
    {
        layer = m_map.get_Layer(i);
        layer.Visible = false;
    }
    m_activeView.PartialRefresh(esriViewDrawPhase.esriViewGeography,
        null, m_activeView.Extent);
}
```

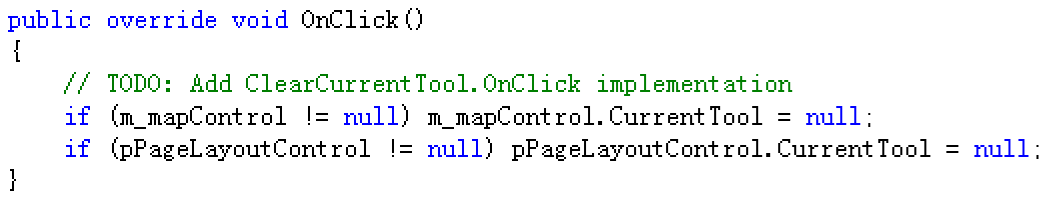
4. Add Base Command class: ClearCurrentTool

- Class member:



```
private IHookHelper m_hookHelper = null;
IMapControl3 m_mapControl = null;
IPageLayoutControl3 pPageLayoutControl = null;
```

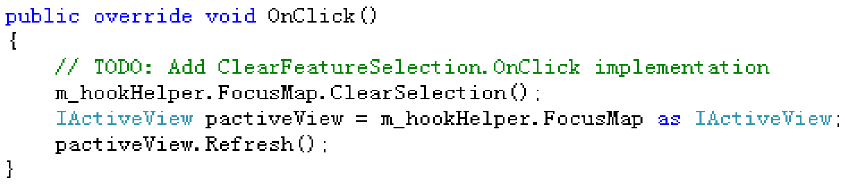
- OnClick:



```
public override void OnClick()
{
    // TODO: Add ClearCurrentTool.OnClick implementation
    if (m_mapControl != null) m_mapControl.CurrentTool = null;
    if (pPageLayoutControl != null) pPageLayoutControl.CurrentTool = null;
}
```

5. Add Base Command class: ClearFeatureSelection

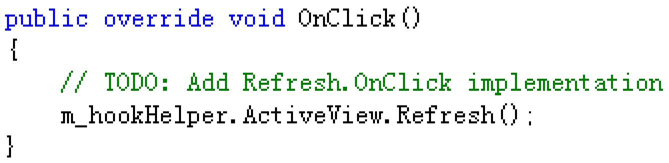
- OnClick:



```
public override void OnClick()
{
    // TODO: Add ClearFeatureSelection.OnClick implementation
    m_hookHelper.FocusMap.ClearSelection();
    IActiveView pactiveView = m_hookHelper.FocusMap as IActiveView;
    pactiveView.Refresh();
}
```

6. Add Base Command class: Refresh

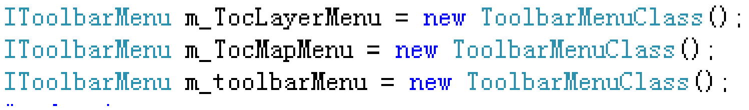
- OnClick:



```
public override void OnClick()
{
    // TODO: Add Refresh.OnClick implementation
    m_hookHelper.ActiveView.Refresh();
}
```

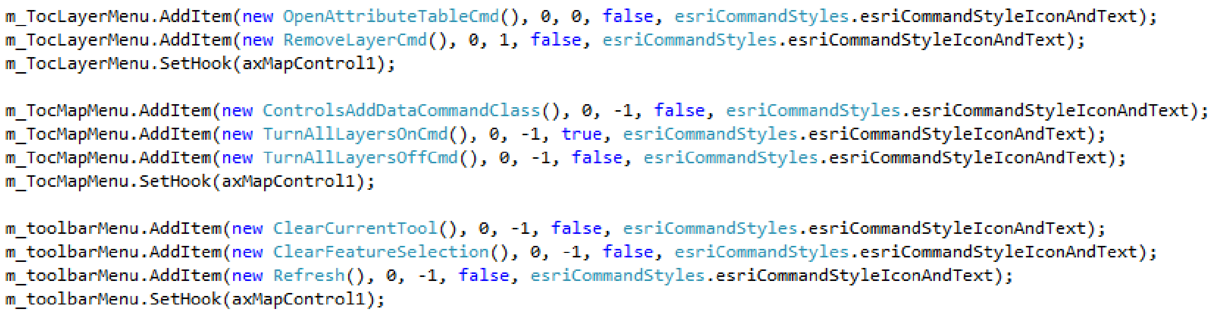
7. Back to MainForm:

- Add Class Member:



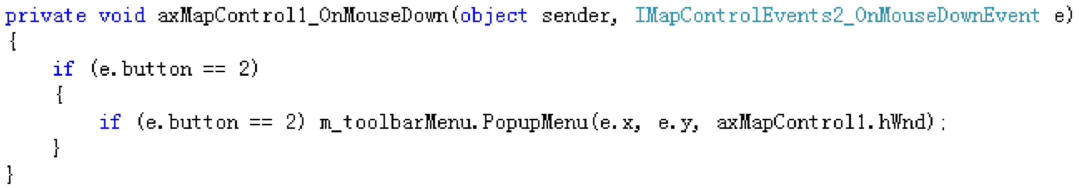
```
IToolbarMenu m_TocLayerMenu = new ToolbarMenuClass();
IToolbarMenu m_TocMapMenu = new ToolbarMenuClass();
IToolbarMenu m_toolbarMenu = new ToolbarMenuClass();
```

- In the Load event, add:

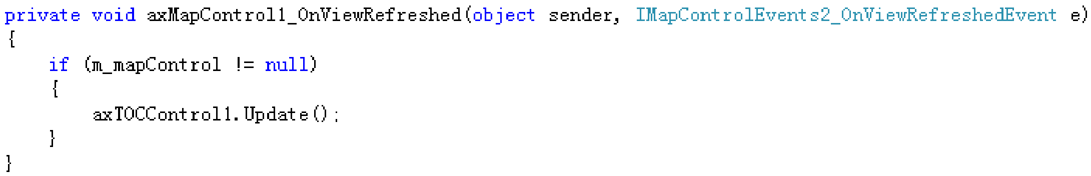


```
m_TocLayerMenu.AddItem(new OpenAttributeTableCmd(), 0, 0, false, esriCommandStyles.esriCommandStyleIconAndText);
m_TocLayerMenu.AddItem(new RemoveLayerCmd(), 0, 1, false, esriCommandStyles.esriCommandStyleIconAndText);
m_TocLayerMenu.SetHook(axMapControl1);

m_TocMapMenu.AddItem(new ControlsAddDataCommandClass(), 0, -1, false, esriCommandStyles.esriCommandStyleIconAndText);
m_TocMapMenu.AddItem(new TurnAllLayersOnCmd(), 0, -1, true, esriCommandStyles.esriCommandStyleIconAndText);
m_TocMapMenu.AddItem(new TurnAllLayersOffCmd(), 0, -1, false, esriCommandStyles.esriCommandStyleIconAndText);
m_TocMapMenu.SetHook(axMapControl1);

m_toolbarMenu.AddItem(new ClearCurrentTool(), 0, -1, false, esriCommandStyles.esriCommandStyleIconAndText);
m_toolbarMenu.AddItem(new ClearFeatureSelection(), 0, -1, false, esriCommandStyles.esriCommandStyleIconAndText);
m_toolbarMenu.AddItem(new Refresh(), 0, -1, false, esriCommandStyles.esriCommandStyleIconAndText);
m_toolbarMenu.SetHook(axMapControl1);
```

- OnMouseDown Event and OnViewRefreshed Event of MapControl:

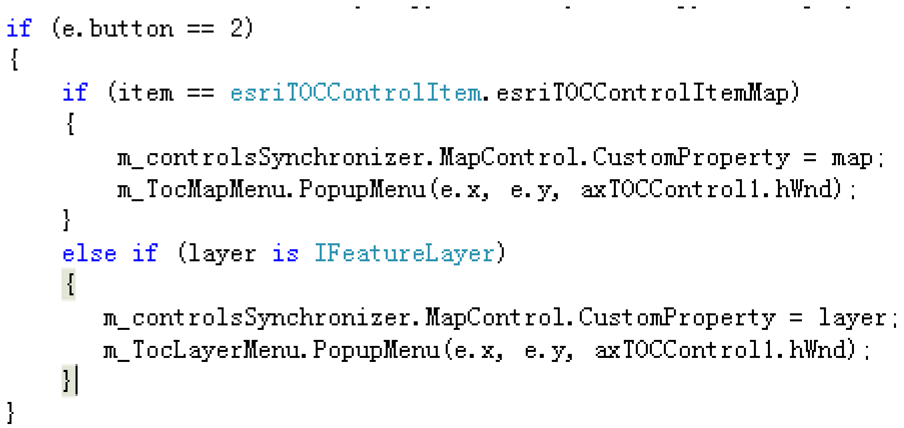


```
private void axMapControl1_OnMouseDown(object sender, IMapControlEvents2_OnMouseDownEvent e)
{
    if (e.button == 2)
    {
        if (e.button == 2) m_toolbarMenu.PopupMenu(e.x, e.y, axMapControl1.hWnd);
    }
}
```



```
private void axMapControl1_OnViewRefreshed(object sender, IMapControlEvents2_OnViewRefreshedEvent e)
{
    if (m_mapControl != null)
    {
        axTOCControl1.Update();
    }
}
```

- OnMouseDown Event of TocControl:



```
if (e.button == 2)
{
    if (item == esriTOCControlItem.esriTOCControlItemMap)
    {
        m_controlsSynchronizer.MapControl.CustomProperty = map;
        m_TocMapMenu.PopupMenu(e.x, e.y, axTOCControl1.hWnd);
    }
    else if (layer is IFeatureLayer)
    {
       m_controlsSynchronizer.MapControl.CustomProperty = layer;
       m_TocLayerMenu.PopupMenu(e.x, e.y, axTOCControl1.hWnd);
    }
}
```