

VTK/Examples/Cxx/PolyData/ColoredPoints

From KitwarePublic

< VTK | Examples | Cxx

ColoredPoints.cxx

```
#include <vtkVersion.h>
#include <vtkSmartPointer.h>
#include <vtkPoints.h>
#include <vtkPolyData.h>
#include <vtkPointData.h>
#include <vtkCellArray.h>
#include <vtkUnsignedCharArray.h>
#include <vtkPolyDataMapper.h>
#include <vtkActor.h>
#include <vtkRenderWindow.h>
#include <vtkRenderer.h>
#include <vtkRenderWindowInteractor.h>
#include <vtkVertexGlyphFilter.h>
#include <vtkProperty.h>

int main(int, char *[])
{
    vtkSmartPointer<vtkPoints> points =
        vtkSmartPointer<vtkPoints>::New();
    points->InsertNextPoint (0.0, 0.0, 0.0);
    points->InsertNextPoint (1.0, 0.0, 0.0);
    points->InsertNextPoint (0.0, 1.0, 0.0);

    vtkSmartPointer<vtkPolyData> pointsPolydata =
        vtkSmartPointer<vtkPolyData>::New();

    pointsPolydata->SetPoints(points);

    vtkSmartPointer<vtkVertexGlyphFilter> vertexFilter =
        vtkSmartPointer<vtkVertexGlyphFilter>::New();
    #if VTK_MAJOR_VERSION <= 5
        vertexFilter->SetInputConnection(pointsPolydata->GetProducerP
    #else
        vertexFilter->SetInputData(pointsPolydata);
    #endif
    vertexFilter->Update();

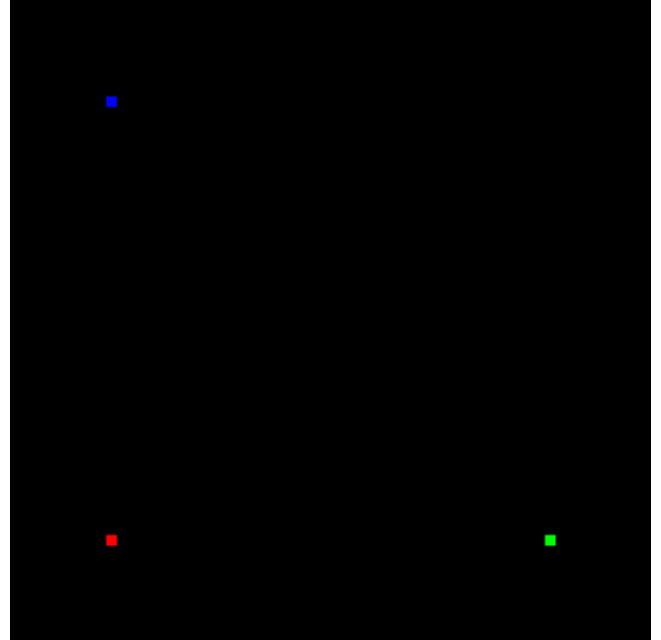
    vtkSmartPointer<vtkPolyData> polydata =
        vtkSmartPointer<vtkPolyData>::New();
    polydata->ShallowCopy(vertexFilter->GetOutput());

    // Setup colors
    unsigned char red[3] = {255, 0, 0};
    unsigned char green[3] = {0, 255, 0};
    unsigned char blue[3] = {0, 0, 255};

    vtkSmartPointer<vtkUnsignedCharArray> colors =
        vtkSmartPointer<vtkUnsignedCharArray>::New();
    colors->SetNumberOfComponents(3);
    colors->SetName ("Colors");
    colors->InsertNextTupleValue(red);
    colors->InsertNextTupleValue(green);
    colors->InsertNextTupleValue(blue);

    polydata->GetPointData()->SetScalars(colors);

    // Visualization
    vtkSmartPointer<vtkPolyDataMapper> mapper =
        vtkSmartPointer<vtkPolyDataMapper>::New();
    #if VTK_MAJOR_VERSION <= 5
```



```

mapper->SetInputConnection(polydata->GetProducerPort());
#else
mapper->SetInputData(polydata);
#endif

vtkSmartPointer<vtkActor> actor =
    vtkSmartPointer<vtkActor>::New();
actor->SetMapper(mapper);
actor->GetProperty()->SetPointSize(5);

vtkSmartPointer<vtkRenderer> renderer =
    vtkSmartPointer<vtkRenderer>::New();
vtkSmartPointer<vtkRenderWindow> renderWindow =
    vtkSmartPointer<vtkRenderWindow>::New();
renderWindow->AddRenderer(renderer);
vtkSmartPointer<vtkRenderWindowInteractor> renderWindowInteractor =
    vtkSmartPointer<vtkRenderWindowInteractor>::New();
renderWindowInteractor->SetRenderWindow(renderWindow);

renderer->AddActor(actor);

renderWindow->Render();
renderWindowInteractor->Start();

return EXIT_SUCCESS;
}

```

CMakeLists.txt

```

cmake_minimum_required(VERSION 2.8)

PROJECT(ColoredPoints)

find_package(VTK REQUIRED)
include(${VTK_USE_FILE})

add_executable(ColoredPoints MACOSX_BUNDLE ColoredPoints)

if(VTK_LIBRARIES)
    target_link_libraries(ColoredPoints ${VTK_LIBRARIES})
else()
    target_link_libraries(ColoredPoints vtkHybrid vtkWidgets)
endif()

```

Download and Build ColoredPoints

Click here to download ColoredPoints

(<http://git.Kitware.com/vtkwiki/examplestarballs/vtkwikiexamplestarballs/raw/master/ColoredPoints.tar>). and its *CMakeLists.txt* file.

Once the tarball *ColoredPoints.tar* has been downloaded and extracted,

```
cd ColoredPoints/build
```

- If VTK is installed:

```
cmake ..
```

- If VTK is not installed but compiled on your system, you will need to specify the path to your VTK build:

```
cmake -DVTK_DIR:PATH=/home/me/vtk_build ..
```

Build the project:

```
make
```

and run it:

```
./ColoredPoints
```

WINDOWS USERS PLEASE NOTE: Be sure to add the VTK bin directory to your path. This will resolve the VTK dll's at run time.

Retrieved from "<http://www.vtk.org/Wiki/index.php?title=VTK/Examples/Cxx/PolyData/ColoredPoints&oldid=51171>"

-
- This page was last modified on 21 January 2013, at 10:23.
 - Content is available under Attribution2.5 unless otherwise noted.