Subject: Re: 2024rc1
Posted by Lance on Mon, 14 Oct 2024 14:23:10 GMT
View Forum Message <> Reply to Message

mirek wrote on Mon, 14 October 2024 10:13Lance wrote on Sat, 12 October 2024 20:20Code Reformat Issue.

I have been having issues with it for a while. Today I spend a few hours to create a almost mininal example.

```
#include <CtrlLib/CtrlLib.h>
using namespace Upp;

class S{
 void Set ( Size& sz,
        int x, int y, int z,
        int x1, int y1, int z1
      );

 const S& f(Rect& r, Rect& s)const;

 struct D
 {
  int s(int rc)const
  {
   return rc*2;
  }

  int e(int rc)const
  {
   return rc*1;
  }

  int w(int rc)const
  {
   return rc*3;
  }

  int t()const
  {
   return 4;
  }


  int g(int k)const
  {
   return k;
  }
```

```
  void alloc();

  void alloc(int a);

  int v1;
  int v2;
 };
 D col,row;
};


void S::D::alloc ()
{
}

static void func ( int& x, int& y, int& x1, int& y1, int x2, int y2,
     int x3, int y3, int x4, int y4
      )
{
}


const S& S::f(Rect& r, Rect& s)const
{
 int t, l, row_section_bottom, n;
 r.top = row.g(r.top);
 s.top = t !=0 && r.top<row.v1 ? row.v1 : r.top;

 r.left = col.g(r.left);
 s.left = l != 0 && r.left< col.v1 ? col.v1 : r.left;

 r.bottom = row.g(r.bottom);
 s.bottom = row_section_bottom == 1  && r.bottom > row.v1 + row.v2 ?
  row.v1+row.v2 : r.bottom;

 r.right = col.g(r.right);
 s.right = n == 1  && r.right> col.v1 + col.v2 ?
  col.v1+col.v2 : r.right;
 return *this;
}
```

Add the code as a separate cpp file in a CtrlLib application, with it current, press Ctrl+I to reformat it. The file before reformat compiles fine, not the reformatted one.

Works for me in windows and works in Ubuntu. Unfortunately, this feature is now using clang-format that can be different per distro...

Would be nice to give me a hint which host platform is in use...

Also, if nothing helps, please post reformatted text as well.

Mirek

Operation System: Ubuntu 24.04.1 LTS
GNOME version: 46
Windowing System: Wayland
clang-format --version: Ubuntu clang-format version 18.1.3 (1ubuntu1)

Reformatted output

```
#include <CtrlLib/CtrlLib.h>
using namespace Upp;

class S {
 void Set(Size& sz, int x, int y, int z, int x1, int y1, int z1);

 const S& f(Rect& r, Rect& s) const;

 struct D {
  int s(int rc) const { return rc * 2; }

  int e(int rc) const { return rc * 1; }

  int w(int rc) const { return rc * 3; }

  int t() const { return 4; }

  int g(int k) const { return k; }

  void alloc();

  void alloc(int a);

  int v1;
  int v2;
 };
 D col, row;
};

void S::D::alloc() {}

static void func(int& x, int& y, int& x1, int& y1, int x2, int y2, int x3, int y3, int x4,
```

```
                int y4)
{
}

const S& S::f(Rect& r, Rect& s) const
{
 int t, l, row_section_bottom, n;
 r.top = row.g(r.top);
 s.top = t != 0 && r.top < row.v1 ? row.v1 : r.top;
 r.left = col.g(r.left);
 s.left = l != 0 && r.left< col.v1 ? col.v1 : r.left;
 s.left = l != 0 && r.left < col.v1 ? col.v1 : r.left;
 r.bottom = row.g(r.bottom);
 s.bottom = row_section_bottom == 1  && r.bottom > row.v1 + row.v2 ?
 s.bottom =
  row_section_bottom == 1 && r.bottom > row.v1 + row.v2 ? row.v1 + row.v2 : r.bottom;
 r.right = col.g(r.right);
 s.right = n == 1  && r.right> col.v1 + col.v2 ?
 s.right = n == 1 && r.right > col.v1 + col.v2 ? col.v1 + col.v2 : r.right;
}
```

Line 46 (?), Line 48 ending (;), LIne 50 (?), Line 51 ending (;) are highlighted by theide(libclang) to indicate grammer errors.