Posted by alex100 on Sat, 08 Mar 2008 09:44:22 GMT

View Forum Message <> Reply to Message

Hello all,

I was trying to print the contents of a RichEdit Control (in a form) without showing the printer dlg and for that I created the method:

```
bool PrinterJob::ExecuteNoDlg()
{
 return Execute0(false);
}
```

And nothing was printed.

There is a method called

```
bool PrinterJob::Execute()
{
 return Execute0(true);
}
```

that prints showing the printer dlg. In my opinion this method must have optional argument to be possible to choose if one wants to show the printer dialog or not:
bool Execute(bool bShow=true);

But lets forget this method and concentrate focus on the

bool PrinterJob::Execute0(bool dodlg) method:

As dlg.nCopies is not initialized when the pronted dlg is not showed:

```
if(dodlg)
{
dlg.Flags =  PD_DISABLEPRINTTOFILE|PD_NOSELECTION|PD_HIDEPRINTTOFILE|PD_R
ETURNDC;
  if(!PrintDlg(&dlg)) return false;
  hdc = dlg.hDC;
}
else
{
```


The dlg.nCopies remains with the value zero and in the code:

```
for(int c = 0; c < ((dlg.Flags & PD_COLLATE) ? dlg.nCopies : 1); c++)
```

```
for(int i = dlg.nFromPage - 1; i <= dlg.nToPage - 1; i++)
 for(int c = 0; c < ((dlg.Flags & PD_COLLATE) ? 1 : dlg.nCopies); c++)
  page.Add(i);
```

the page.Add(i); instruction is never executed. Thus, in function:

void Print(Draw& w, const RichText& text, const Rect& page, const Vector<int>& pg)

the for(int pi = 0; pi < pg.GetCount(); pi++) cicle is never executed even rich edit control has lots of pages.

To fix this I initialized the variable

dlg.nCopies=1;

On the beginning of the bool PrinterJob::Execute0(bool dodlg) method.

What do you think?

Alex

Subject: Re: Bug in PrinterJob class?!
Posted by mirek on Sat, 08 Mar 2008 16:03:15 GMT
View Forum Message <> Reply to Message

alex100 wrote on Sat, 08 March 2008 04:44Hello all,

I was trying to print the contents of a RichEdit Control (in a form) without showing the printer dlg and for that I created the method:

bool PrinterJob::ExecuteNoDlg()
{
 return Execute0(false);
}

And nothing was printed.

There is a method called

bool PrinterJob::Execute()
{

```
 return Execute0(true);
}
```

that prints showing the printer dlg. In my opinion this method must have optional argument to be possible to choose if one wants to show the printer dialog or not:
bool Execute(bool bShow=true);

But lets forget this method and concentrate focus on the

bool PrinterJob::Execute0(bool dodlg) method:

As dlg.nCopies is not initialized when the pronted dlg is not showed:

```
if(dodlg)
{
dlg.Flags =   PD_DISABLEPRINTTOFILE|PD_NOSELECTION|PD_HIDEPRINTTOFILE|PD_RETURNDC;
  if(!PrintDlg(&dlg)) return false;
  hdc = dlg.hDC;
}
else
{
```

The dlg.nCopies remains with the value zero and in the code:

```
for(int c = 0; c < ((dlg.Flags & PD_COLLATE) ? dlg.nCopies : 1); c++)
 for(int i = dlg.nFromPage - 1; i <= dlg.nToPage - 1; i++)
  for(int c = 0; c < ((dlg.Flags & PD_COLLATE) ? 1 : dlg.nCopies); c++)
   page.Add(i);
```

the page.Add(i); instruction is never executed. Thus, in function:

void Print(Draw& w, const RichText& text, const Rect& page, const Vector<int>& pg)

the for(int pi = 0; pi < pg.GetCount(); pi++) cicle is never executed even rich edit control has lots of pages.

To fix this I initialized the variable

dlg.nCopies=1;

On the beginning of the bool PrinterJob::Execute0(bool dodlg) method.

What do you think?

Alex

I think we should release more often; this bug was already solved... (exactly the way you suggest). Sorry for the troubles...

Mirek