

U++ - Bug #2004

ide: Problem with "note:" being red because there is "error" in the identifier

11/28/2019 01:10 PM - Miroslav Fidler

Status:	Approved	Start date:	11/28/2019
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description			
https://slack-imgs.com/?c=1&#38;o1=ro&#38;url=https%3A%2F%2Fi.postimg.cc%2FQXWN72pg%2Fscreen%2F20191122-143531.png			

History

#1 - 11/30/2019 01:55 AM - Sender Ghost

Some testcase for [Ide::FindLineError](#) method:

```
ClearErrorsPane();
//String line = "/usr/local/include/gtk-3.0/gdk/gdkmain.h:70:32: note: 'gdk_error_trap_push' declared here";
String line = Format("%s:1:1: note: 'gdk_error_trap_push' declared here", GetFirstFile());
ConsoleLine(line);
SetBottom(BERRORS);
```

which possible to place at the end of [Ide::SetMain](#) method.

Current f.kind equals to 1 (as for [error](#) case), while its value maybe as 3 (by [default](#)).

#2 - 11/30/2019 02:25 AM - Sender Ghost

Another testcase, based on current implementation of [Ide::FindLineError](#) method:

```
#include <Core/Core.h>

int CheckLine(const String& ln)
{
    int kind = 3;

    if(ln.Find("warning") >= 0)
        kind = 2;
    else {
        int q = ln.Find("error");
        if(q >= 0 && (q == 0 || !IsAlpha(ln[q - 1])) && (q + 5 >= ln.GetCount() || !IsAlpha(ln[q - 1])))
            kind = 1;
    }

    return kind;
}

CONSOLE_APP_MAIN
```

```
{
#define CHECKLINE(line, kind) ASSERT(CheckLine(line) == kind);
    CHECKLINE("/path/to/file:1:1: error: expected expression", 1);
    CHECKLINE("/usr/local/include/X11/X.h:139:9: warning: 'CurrentTime' macro redefined", 2);
    CHECKLINE("/usr/local/include/gtk-3.0/gdk/gdkmain.h:70:32: note: 'gdk_error_trap_push' declared here", 3);
}
```

#3 - 11/30/2019 02:27 AM - Sender Ghost

Sender Ghost wrote:

```
| Another testcase, based on current implementation of Ide::FindLineError method:
| [...]
|
```

Need to add:

```
using namespace Upp;
```

#4 - 11/30/2019 08:53 AM - Sender Ghost

Also possible to add following cases to check:

```
CHECKLINE("/path/to/warning:1:1: error: expected expression", 1);
CHECKLINE("/path/to/error:1:1: warning: implicit conversion turns floating-point number into integer: 'double' to 'bool' [-Wfloat-conversion]", 2);
CHECKLINE("c:\\path\\to\\file.cpp(10): warning C4710: 'std::exception_ptr std::exception_ptr::_Current_exception(void) throw()': function not
inlined", 2);
CHECKLINE(" c:\\program files (x86)\\microsoft visual studio 14.0\\vc\\include\\exception(299): note: see declaration of
'std::exception_ptr::_Current_exception'", 3);
```

#5 - 11/30/2019 08:57 AM - Sender Ghost

or with ".cpp" extension:

```
CHECKLINE("/path/to/warning.cpp:1:1: error: expected expression", 1);
CHECKLINE("/path/to/error.cpp:1:1: warning: implicit conversion turns floating-point number into integer: 'double' to 'bool' [-Wfloat-conversion]",
2);
```

#6 - 11/30/2019 09:33 AM - Sender Ghost

Possible algorithm for C/C++ errors/warnings (Clang, GCC, MSVC):

```
if(ln.Find(": warning") >= 0)
    f.kind = 2;
else if(ln.Find(": error") >= 0)
    f.kind = 1;
```

```
else
  f.kind = 3;
```

But for other compilers this maybe different, I guess, based on current more generic implementation.

#7 - 12/04/2019 03:31 AM - Sender Ghost

- File *uppsrc_ide.diff* added

Attached patch for mentioned checks, just in case.

#8 - 12/04/2019 04:20 AM - Sender Ghost

- File *TheIDE_screenshot_20191204.png* added

Attached screenshot for TheIDE (with applied patch) after build of examples/AddressBook package for gtk3 branch.

#9 - 01/22/2020 02:09 PM - Miroslav Fidler

- Status changed from *New* to *Approved*

Files

uppsrc_ide.diff	710 Bytes	12/04/2019	Sender Ghost
TheIDE_screenshot_20191204.png	261 KB	12/04/2019	Sender Ghost