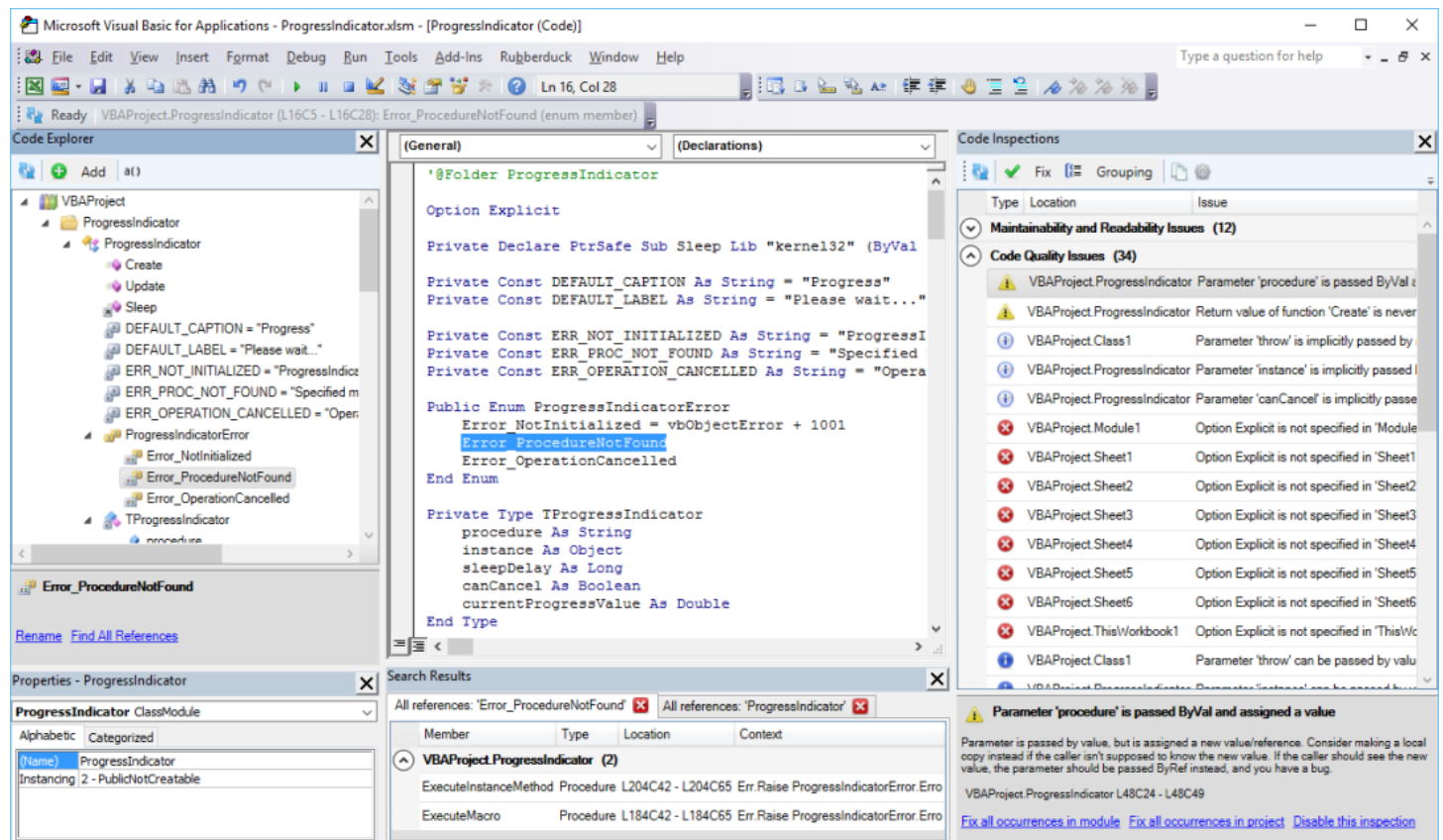


## Rubberduck News

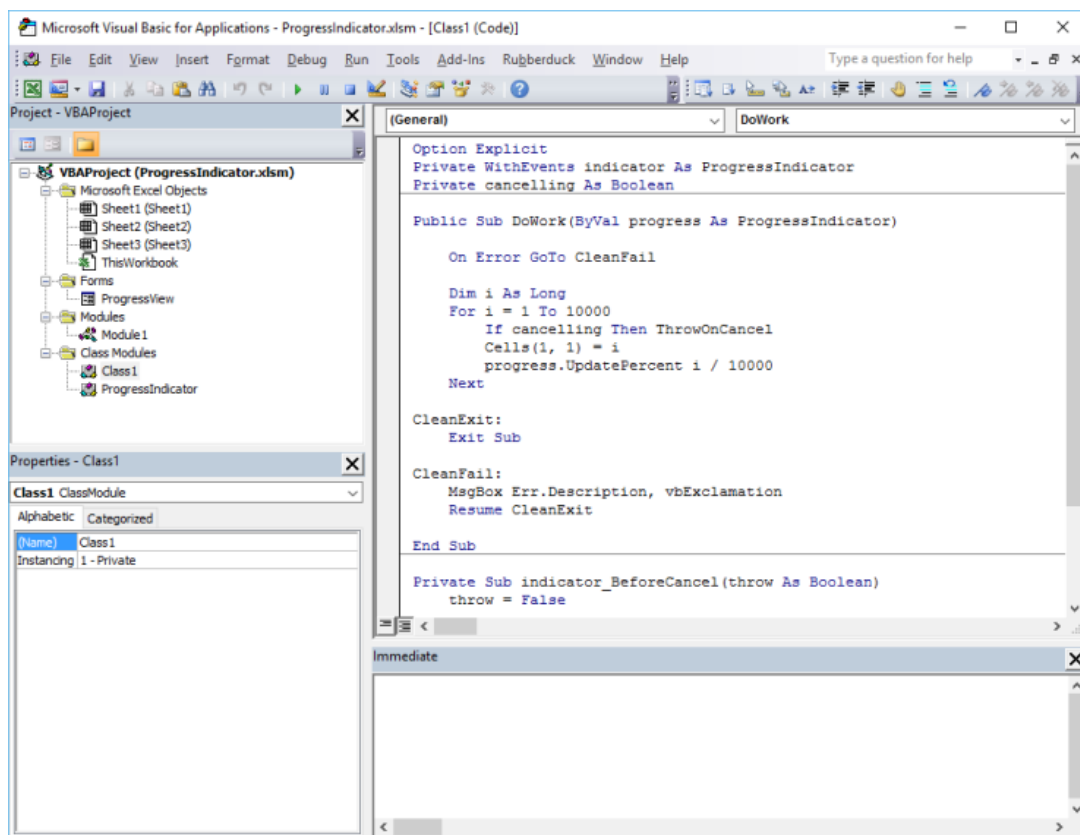
*Every programmer needs a Rubberduck*



## VBA Rubberducking (Part 1)

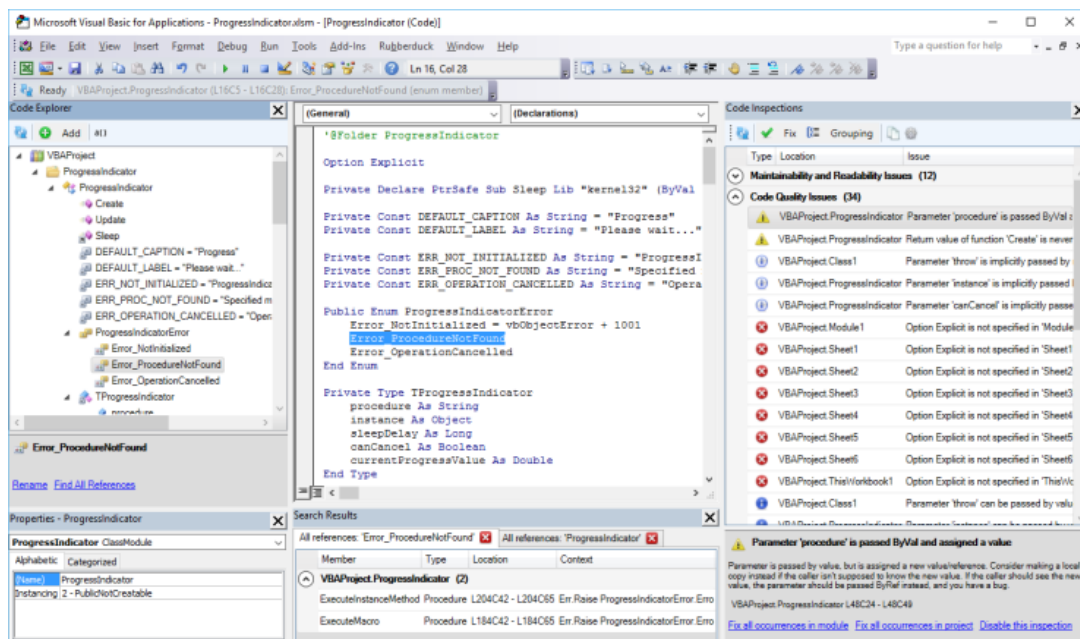
Posted on May 4, 2016 August 9, 2016 by Rubberduck VBA

The VBE editor was last updated in 1998 – back when VB6 was all the rage, and the .NET framework was probably just a little more than a nice idea.



The VBE was just slightly less full-featured than its standalone counterpart, Visual Studio 6.0; however years went by, and the latest Visual Studio versions make the VBE look like an odd beast from another century.

Enter Rubberduck.



There *are* other VBE add-ins out there. For many years, VBA (and VB6) devs have loved using *MZ-Tools* and *Smart Indenter* – perhaps the two most popular add-ins ever written for the VBE. One has a lightning-fast analyzer that is capable of finding unused declarations, and even locates references in commented-out code; the other offers a highly configurable indenter that lets you instantly apply an indenting style to an entire module, or more surgically to a single procedure. What does Rubberduck bring to the table?

Lots, lots, *lots* of things.

This post is the first in a series of post that walk you through the various features of the Rubberduck open-source VBE add-in.

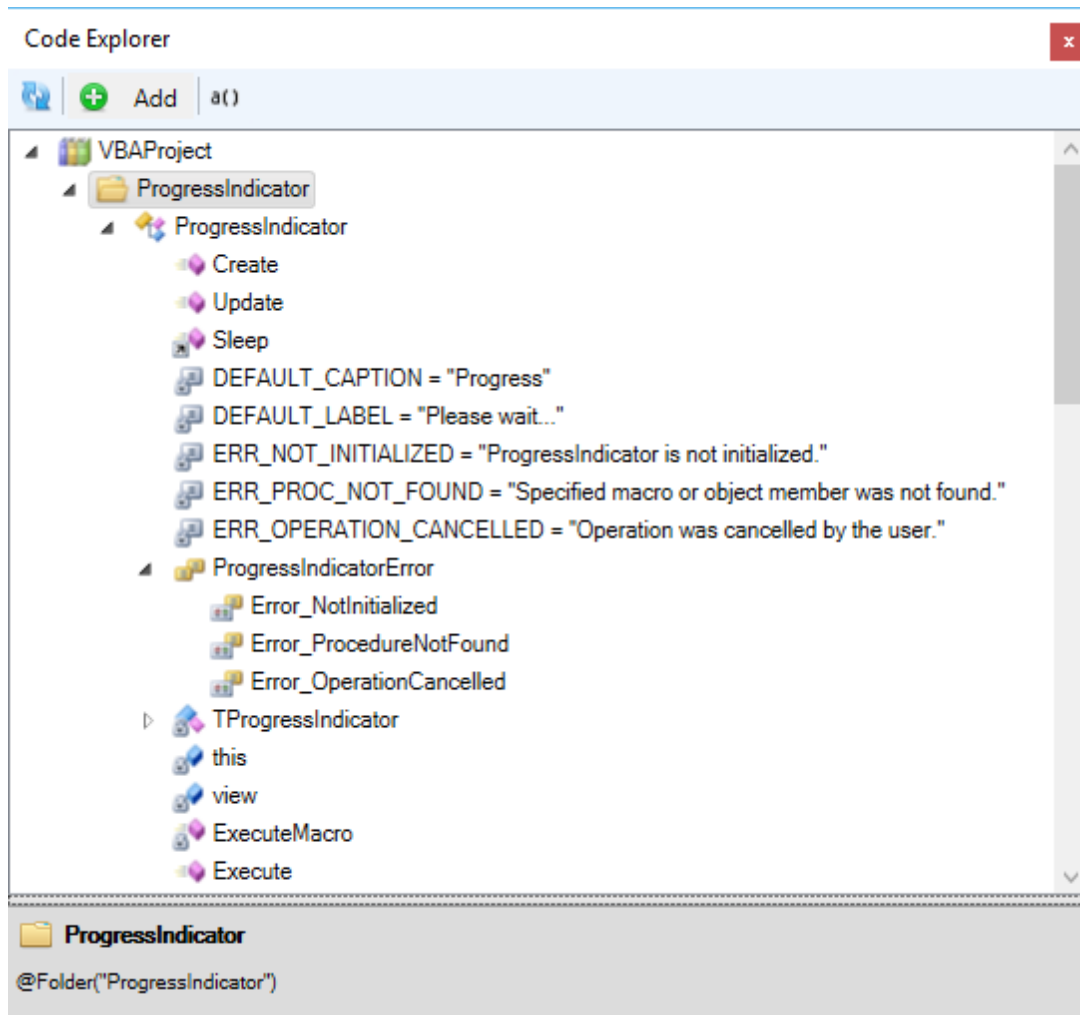
## Navigation Tools

One of the most frustrating aspects of the VBE, is its limited set navigation tools. Let's recap what the *vanilla VBE* gives us:

- **Ctrl+F / "Find"** is a little more than a basic text search, that lets you search and replace text in the current procedure, module, project, or selection. Although VBA isn't case-sensitive, you can match case, and use pattern matching (<https://msdn.microsoft.com/en-us/library/te33kb6t.aspx>), which isn't exactly a regex search, but better than nothing.
- **Shift+F2 / "Go to Definition"**, is actually fantastic: you can right-click any identifier and jump to its declaration – and if it's an identifier defined in a referenced library, it takes you to its entry in the *Object Browser*.
- **Ctrl+R / "Project Explorer"** is a dockable toolwindow that lists all opened projects and the modules under them, in a convenient TreeView where you can double-click on a node and navigate there.
- **Ctrl+Shift+F2 / "Last Position"** is also fantastic: the VBE keeps a little stack of recent places you've been, and works like a "back" browser button that takes you back to where you were before. Quite possibly my personal favorite of all.
- **Bookmarks** is under-used... for a reason. You can toggle any line as a bookmark, and cycle through them, but there's no place to see them all at once.

And... that's about it. Let's see what Rubberduck has to offer.

## Code Explorer



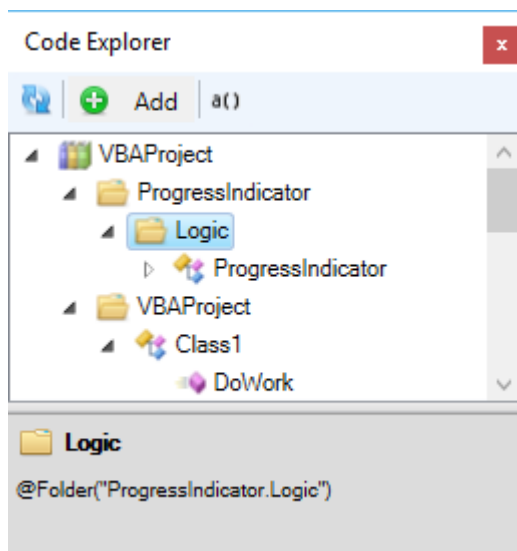
This isn't the *final* version (we haven't released it in 2.0 yet). When it grows up, it wants to be a full-fledged replacement for the *Project Explorer*. Its default hotkey even hijacks the Ctrl+R shortcut. Here's what it does that the *Project Explorer* doesn't do:

- **Drill down** to module members, and then further down to list enum and user-defined type members.
- **See** constant values as they appear in code.
- **Navigate** not only to any module, but any field, enum member, constant, procedure, property get/let/set accessor, function, imported library functions and procedures.
- **Rename** anything.. without breaking the code that references what you're renaming.
- **Find all references** to anything.
- **Indent** an entire project, or a selected module.

But the coolest thing is that Rubberduck's *Code Explorer* takes special comments like this:

```
'@Folder("ProgressIndicator.Logic")
```

And then renders the module like this:

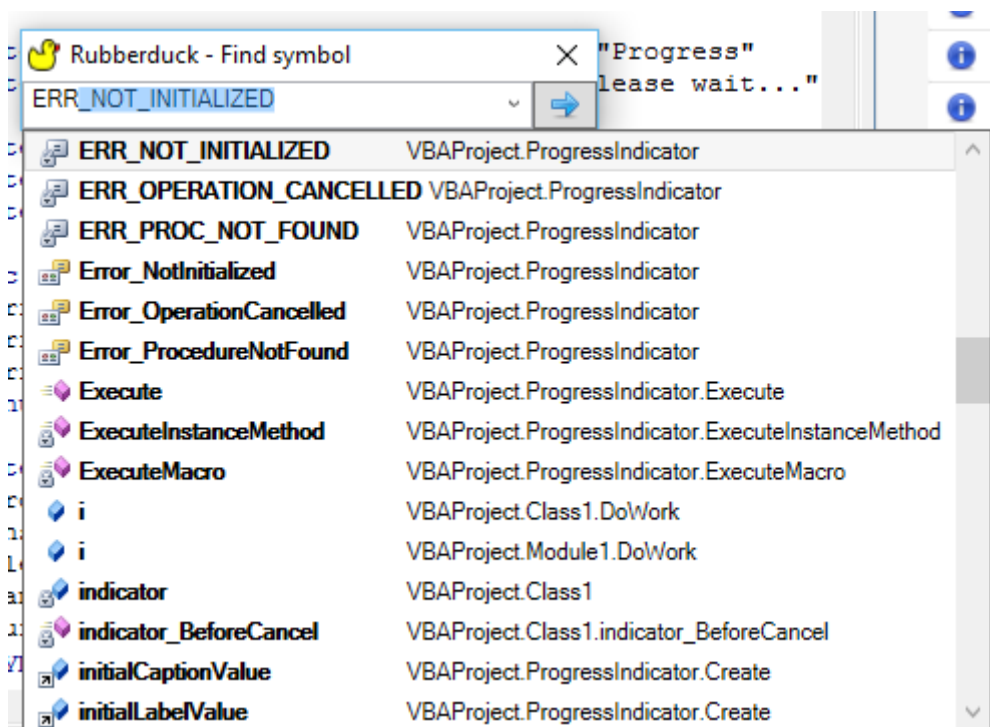


That's right. **Folders**. In VBA. Sure, okay, they're not *real* folders – it's a trick, an illusion... but that *trick* now means that with a simple annotation in every module, you can organize your VBA project *the way you want to*; you're no longer forced to search for a class module among 80 others in a large project, you're free to regroup forms together with their related classes!

This feature alone is a game changer: with it, class modules can become first-class citizen; you don't have to fear drowning in a sea of modules, and you don't have to give them funky prefixes to have them sorted in a way that makes it anywhere near *decent* to navigate.

## Find Symbol

One of my favorite ReSharper features, is Ctrl+T / "go to anything". When I realized we could have this feature in the VBE, I went ahead and did it. This simple feature lets you type the name of **any** identifier, and locate a specific instance of it:

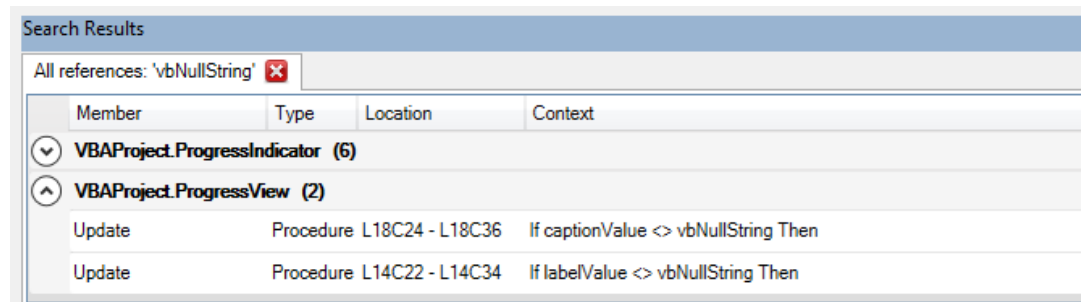


This includes any variable, constant, type, enum, procedure, function, property, library function/procedure, parameter, ...even line labels can be navigated to.

Just Ctrl+T, type something, hit ENTER, and you're there. Or browse the dropdown list and click that "go" button.

## Find all references

Whether you're looking for all call sites of a procedure in your code, or you're just curious about how many times you're using the **vbNullString** built-in constant, you can right-click any identifier (at the declaration, or any of its references) and *Find all references* will give it to you, in a convenient tabbed *search results* toolwindow:



Double-click any result to navigate there.

## Find all implementations

Similar to *find all references* (its results use the same toolwindow), this one is also one of my favorite ReSharper features, that Rubberduck simply *had* to implement. It's only useful when you're coding against abstractions and implementing interfaces (if you didn't know... yes, VBA code can do that!) – but then, it's the best way of navigating to implementations of an interface class or member.

For example, here I added two new class modules, added this line in each, and then implemented the members:

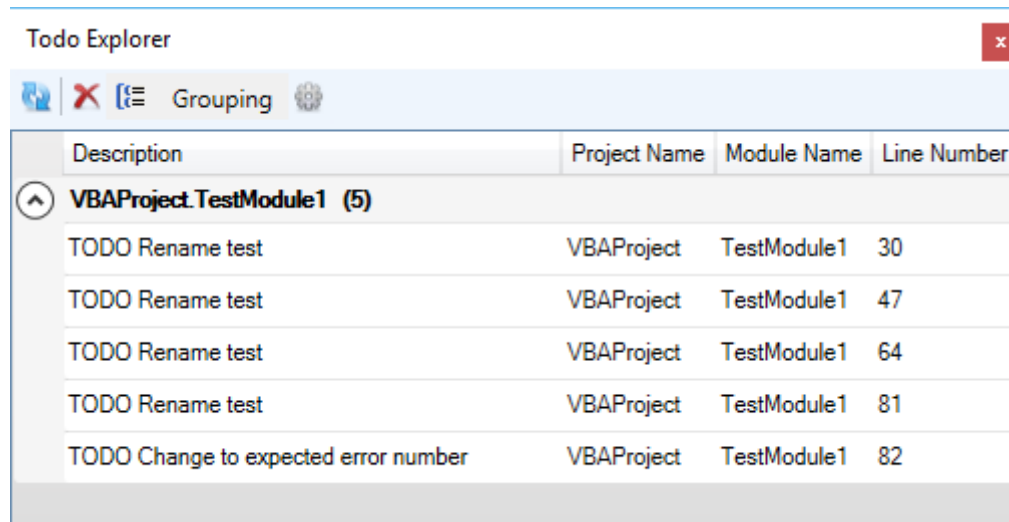
Implements ProgressIndicator

After refreshing the parser state, I can right-click the **Execute** method in my *ProgressIndicator* class, select "Find all implementations", and get this:



# TODO Explorer

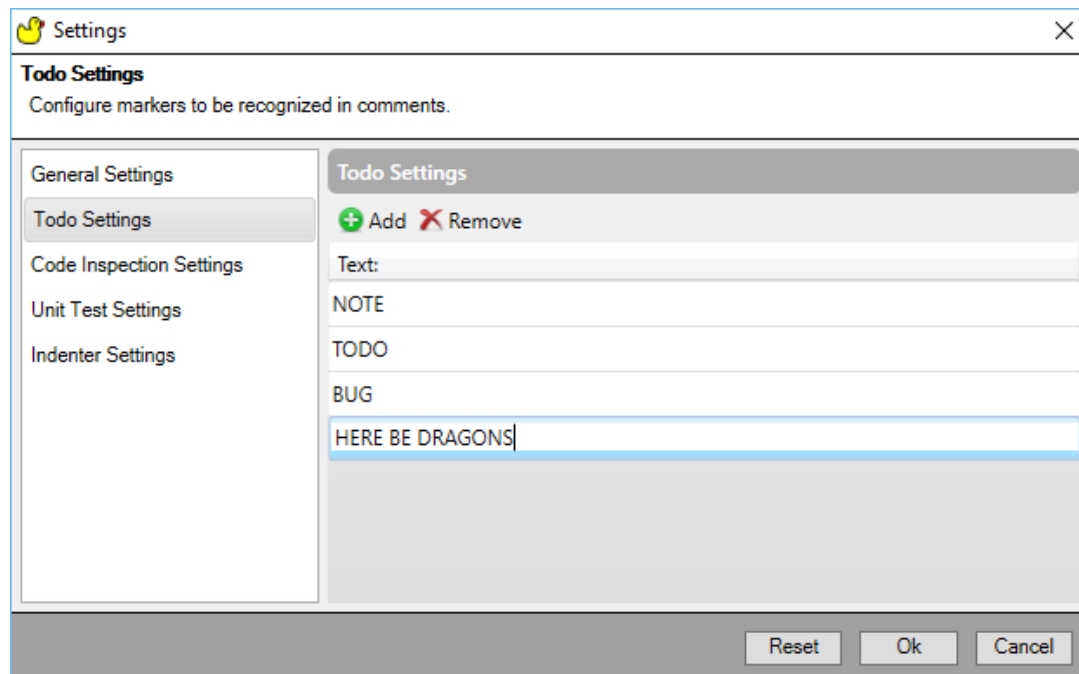
Rubberduck can (well, *does* actually) spot special markers in comments, and lets you navigate them in a dockable toolwindow – again, double-click navigates there:



Description	Project Name	Module Name	Line Number
VBAPProject.TestModule1 (5)			
TODO Rename test	VBAPProject	TestModule1	30
TODO Rename test	VBAPProject	TestModule1	47
TODO Rename test	VBAPProject	TestModule1	64
TODO Rename test	VBAPProject	TestModule1	81
TODO Change to expected error number	VBAPProject	TestModule1	82

Take that, *bookmarks*! You can group them by marker or by location.

By default, Rubberduck will mark **NOTE**, **TODO** and **BUG** as interesting, but you can always configure it to whatever suits your needs in the *Todo Settings* tab of the settings dialog:



Settings

**Todo Settings**  
Configure markers to be recognized in comments.

General Settings  
Todo Settings  
Code Inspection Settings  
Unit Test Settings  
Indenter Settings

Todo Settings

+ Add - Remove

Text:

NOTE

TODO

BUG

HERE BE DRAGONS

Reset Ok Cancel

## Regex Search & Replace

Okay, that one's not really there yet. But it's totally on the roadmap, and definitely coming in a future version of Rubberduck. Take that, *search with pattern*!



Whew! That covers Rubberduck's navigation features. What do you think?

**To be continued...**



REPORT THIS AD



REPORT THIS AD

Posted in [open-source](#), [rubberduck](#), [tutorials](#), [vba](#) Tagged [add-in](#), [code analysis](#), [identifier resolution](#), [navigation](#), [parsing](#), [rubberduck](#), [v2.0](#), [vba](#), [vba tools](#), [vbe](#)



## Published by Rubberduck VBA

---

I'm Mathieu Guindon (Microsoft MVP Office Apps & Services, 2018), you may have known me as "Mat's Mug" on Stack Overflow and Code Review Stack Exchange. I manage the Rubberduck open-source project, whose goal is to bring the Visual Basic Editor (VBE) - VBA's IDE - into the 21st century, by providing features modern IDE's provide. [View all posts by Rubberduck VBA](#)



## 6 thoughts on “VBA Rubberducking (Part 1)”

1. **Felipe Costa Gualberto**    May 4, 2016    Reply.

---

Hello,

Is the last version already with all these features and stable?

**rubberduckvba**    May 4, 2016    Reply.

---

Hi Felipe, except for the code explorer, all these features are in the 2.0.1a preview release – we’re still working on stabilizing the parser state and COM interactions though, so no, not exactly stable yet, but “good enough” for a preview release to get an idea of what’s coming to 2.0.

2. **VBA Rubberducking (Part 2) – Rubberduck News** May 14, 2016 Reply.

[...] of post that walk you through the various features of the Rubberduck open-source VBE add-in. The first post was about the navigation [...]

3. **VBA Rubberducking (Part 3) – Rubberduck News** May 18, 2016 Reply.

[...] Part 1 introduced the navigation features. [...]

4. **FreeMan**    May 18, 2016    Reply.

---

Shift-F2 ? I never knew!!!

5. **VBA Rubberducking (Part 4) – Rubberduck News** May 28, 2016 Reply.

[...] Part 1 introduced the navigation features. [...]



