



ZTPFGI
Suite

- Code Sense

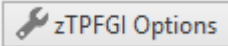
Introduction

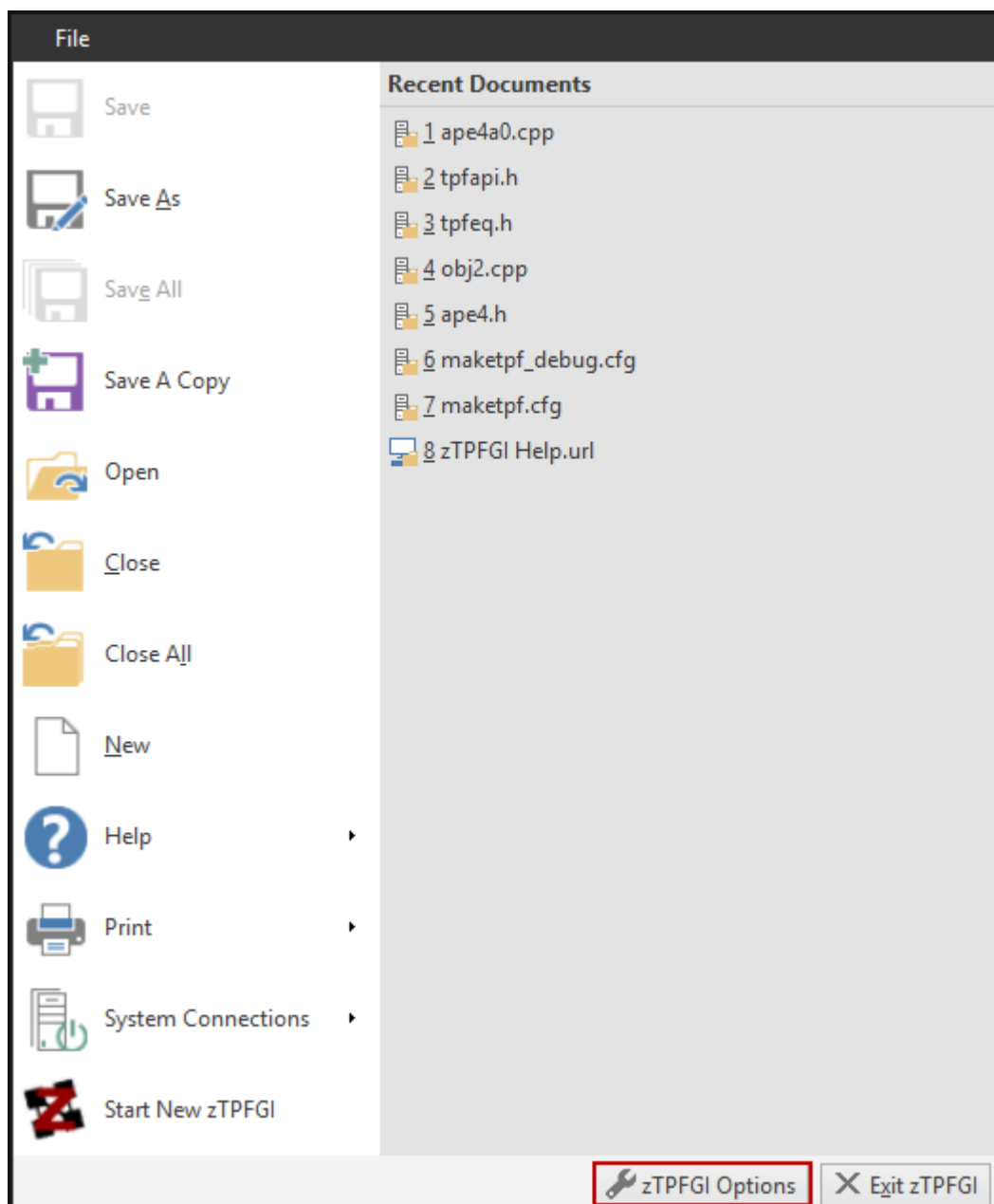
Code Sense is a powerful feature that assists programmers with editing C/C++ source code in zTPFGI. It aids in writing well-formatted code. It also helps to maintain a uniform coding standard across the team/organization.

The Code Sense feature provides the following sub-features to assist you while coding:

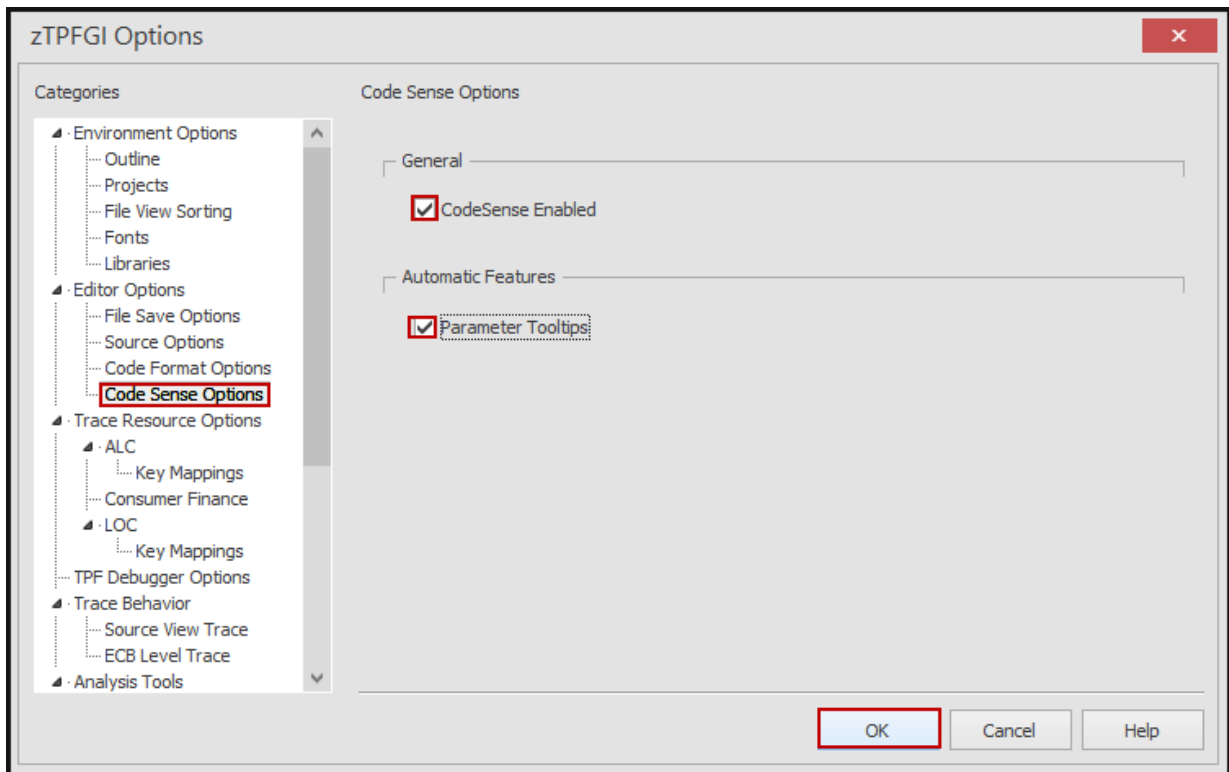
- Code Formatting
- Code Completion & Parameter Tooltips
- Imports Navigation

How do I enable code sense options?

1. Launch zTPFGI, open the **File** tab, then click .



The zTPFGI Options pop-up window displays



2. In the **zTPFGI Options** pop-up window, select the **Code Sense Options** from the **Categories** section.
3. Check the **CodeSense Enabled** and **Parameter Tooltips** checkboxes to enable code sense.

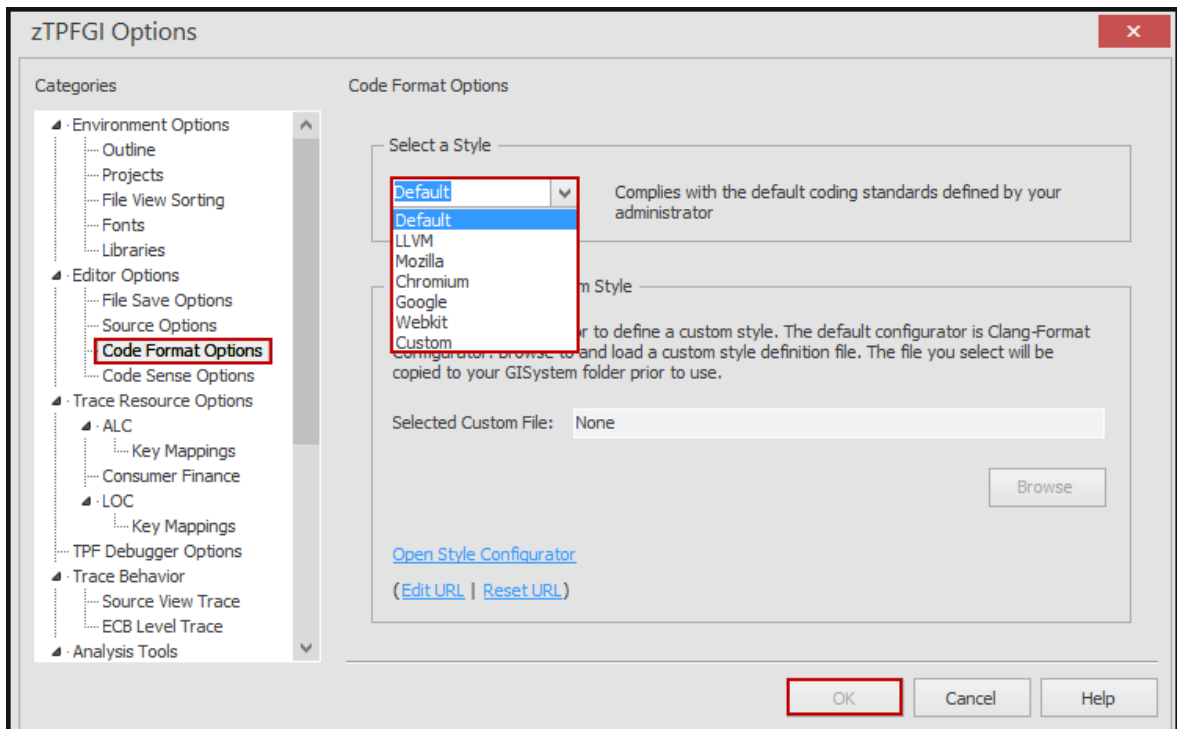
Code Formatting Overview

Code Formatting allows you to format C/C++ source files that are open in the zTPFGI editor area quickly and easily, in accordance with predefined rule-sets.

You can either use the existing formatting styles available in GI or create your own code formatting options (i.e. company-specific formatting options) by customizing the formatting settings under the zTPFGI Options.

How do I set code format options?

1. In the **zTPFGI Options** pop-up window, select the **Code Format Options** from the **Categories** section.



2. In the **Code Format Options** section, the **Default** style would be selected, if your administrator has defined a custom style for your company or team. Otherwise, the default is LLVM.



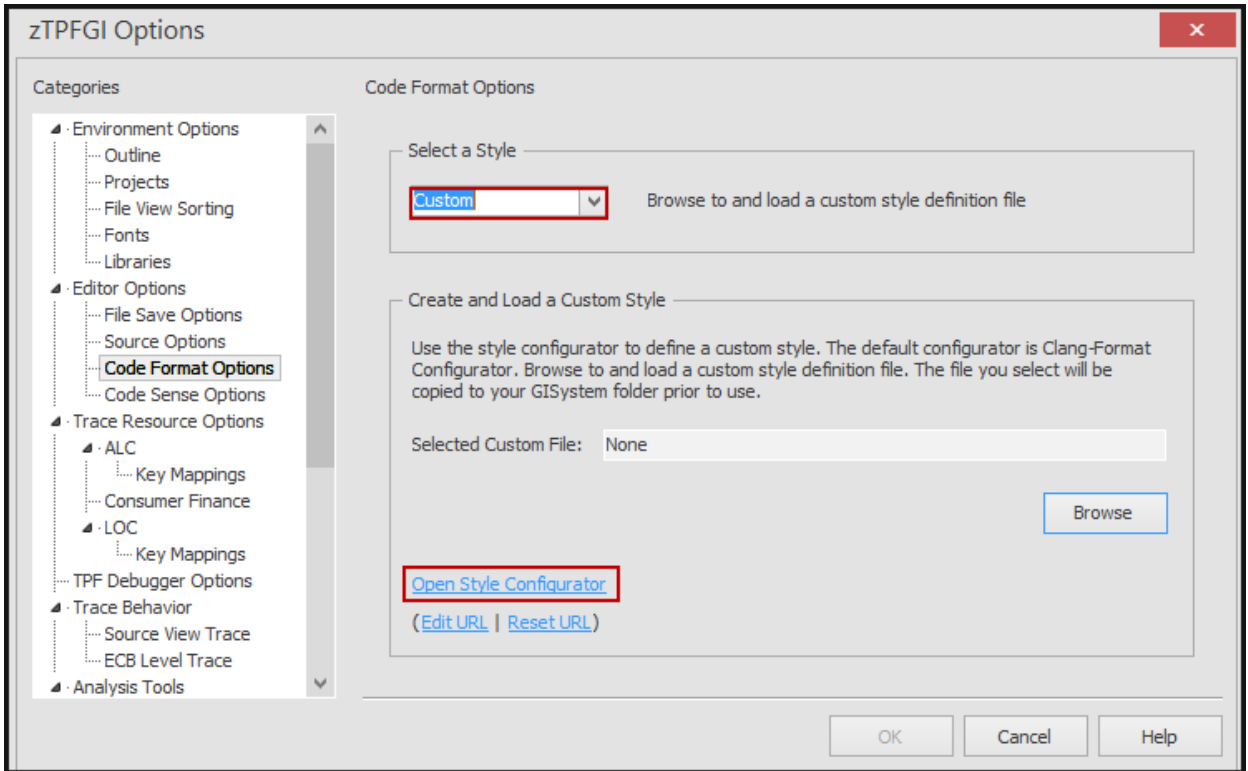
Note: You can change the formatting style that is used by selecting any of the following options from the Select a Style drop-down menu.

- **LLVM:** This style complies with the LLVM coding standard.
- **Mozilla:** This style complies with the Mozilla coding standard.
- **Chromium:** This style complies with the Chromium coding standard.
- **Google:** This style complies with the Google coding standard.
- **Webkit:** This style complies with the Webkit coding standard.
- **Custom:** Select this option if you want to define your own custom style.

3. Click the **OK** button to set the selected format style for code formatting.

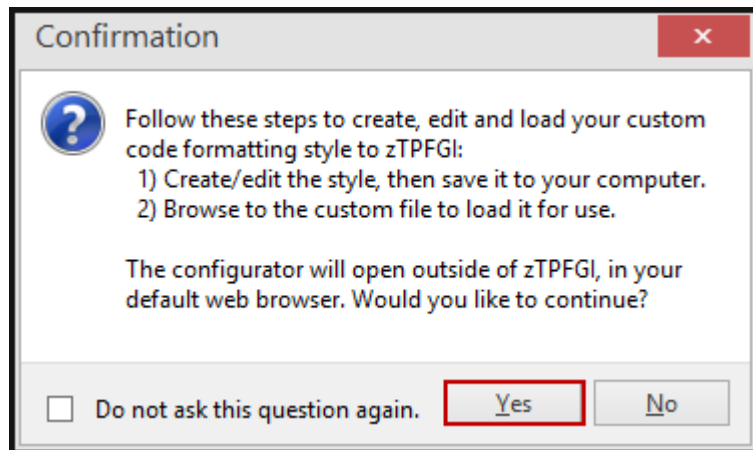
How do I create a custom code formatting style?

1. Select **Custom** from the **Select a Style** drop-down menu.



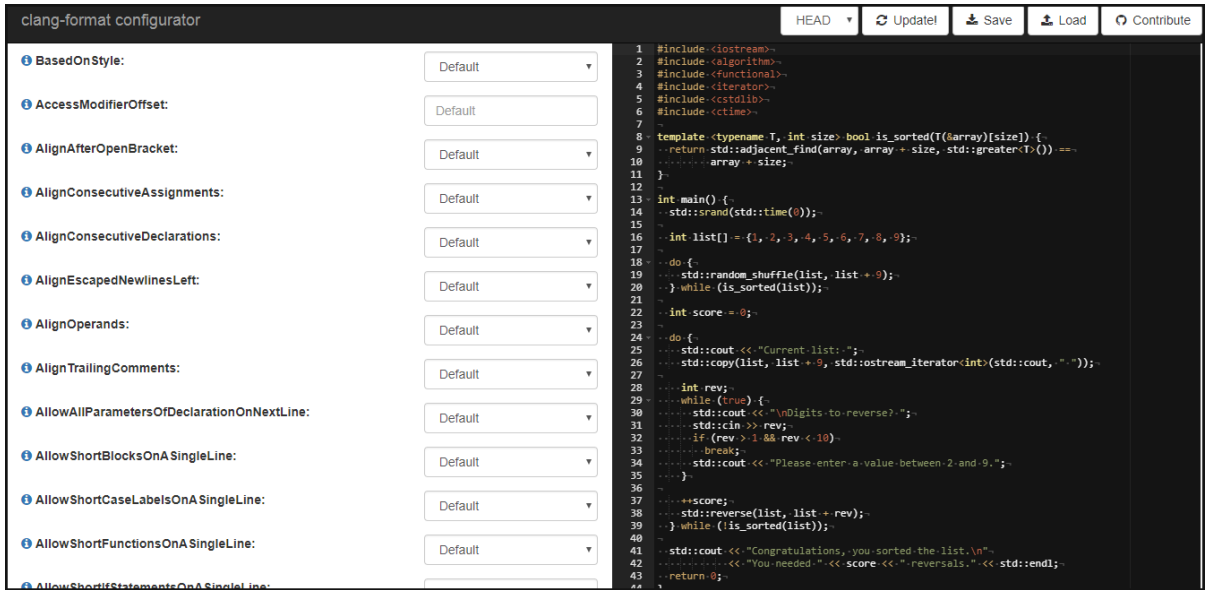
2. Click the **Open Style Configurator** link.

A confirmation pop-up window displays to confirm whether you want to create a custom formatting style

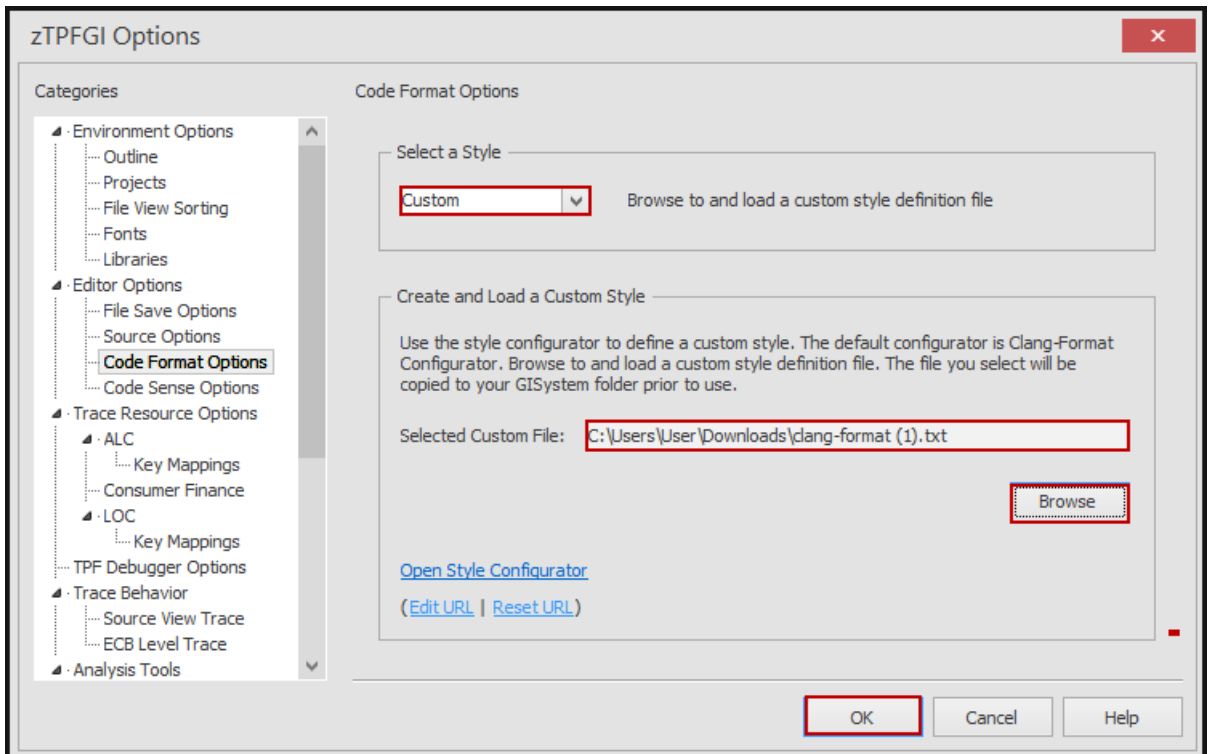


3. Click the **Yes** button to confirm.

The Third Party Configurator displays



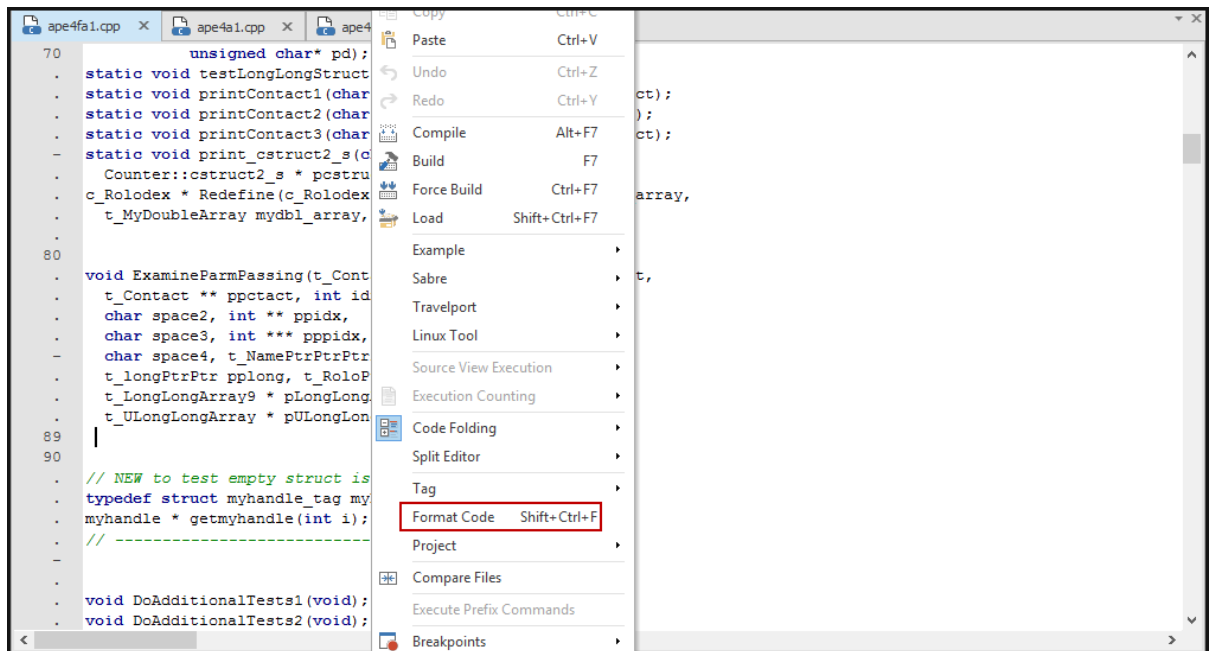
4. Define your own formatting styles and save it to your computer.
5. Browse and choose the path of the custom formatting style file created.



6. Click the **OK** button to upload the custom formatting style file to GI.

How do I format source code?

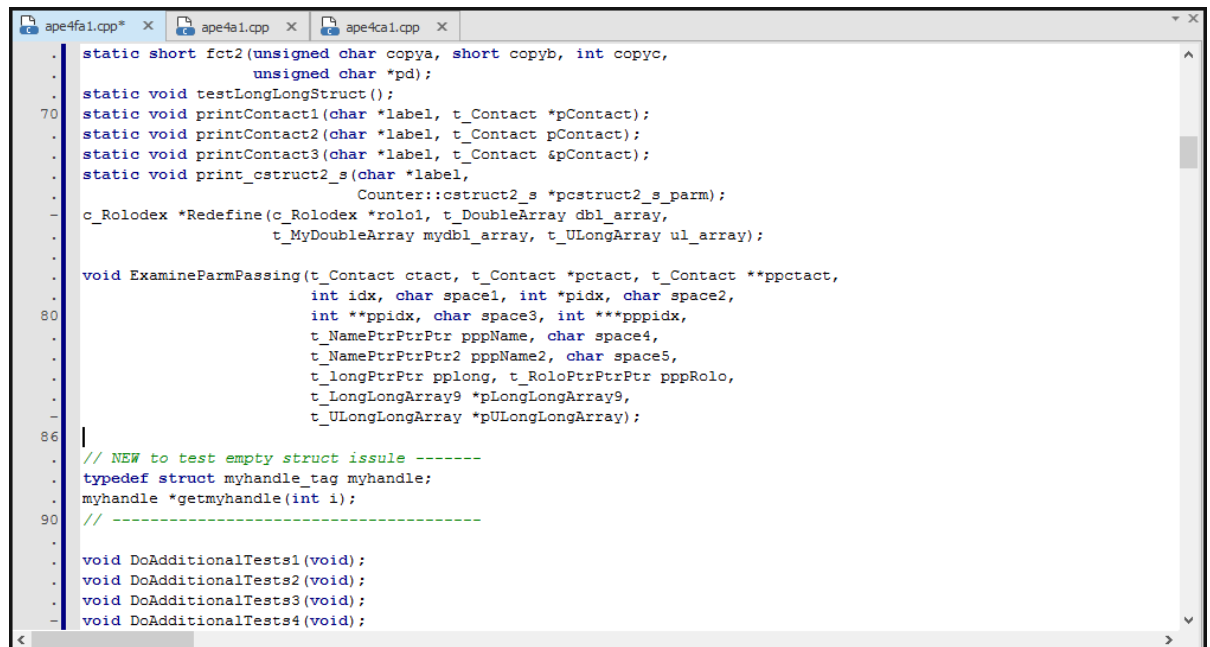
1. Open a C/C++ source file in the editor.
2. Place the cursor in a valid location in the file.



3. Select a specific region of code to be formatted, right-click and select **Format Code** option.
<OR>

Right-click and select the **Format Code** option to format the active file.

Based on the selected formatting options, the code is formatted and the changes are shown in the editor.

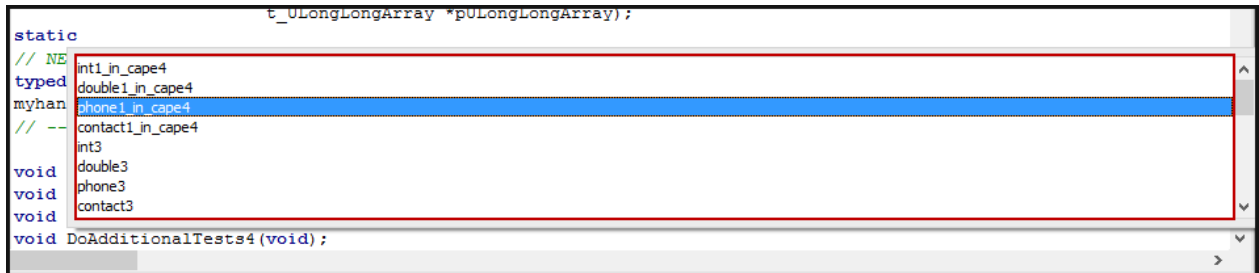


Code Completion & Parameter Tooltips

You can use code completion and parameter tooltips to help you with your coding. Code completion and parameter tooltips work with local and global variables, and with functions within the current file scope.

Code Completion

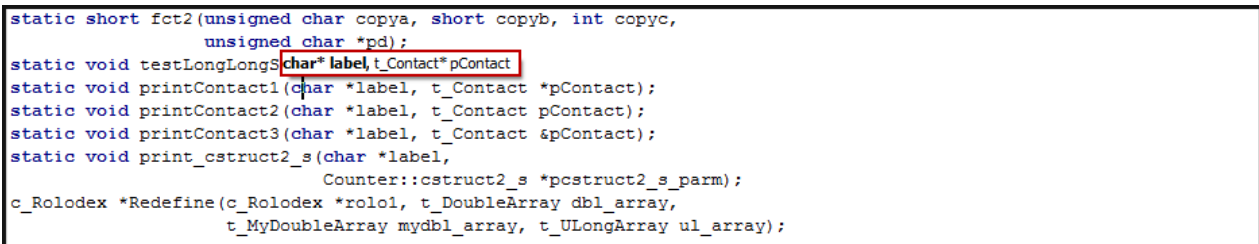
While coding, press CTRL+SPACEBAR to display suggestions for code completion. You are only shown the functions and variables that are in scope for where your cursor is in the source code. Select the suggestion that you want to apply it to your code.



```
static t_ULongLongArray *pULongLongArray);
// NE
typed int1_in_cape4
myhan double1_in_cape4
// -- phone1_in_cape4
contact1_in_cape4
int3
double3
void phone3
void contact3
void DoAdditionalTests4(void);
```

Parameter Tooltips

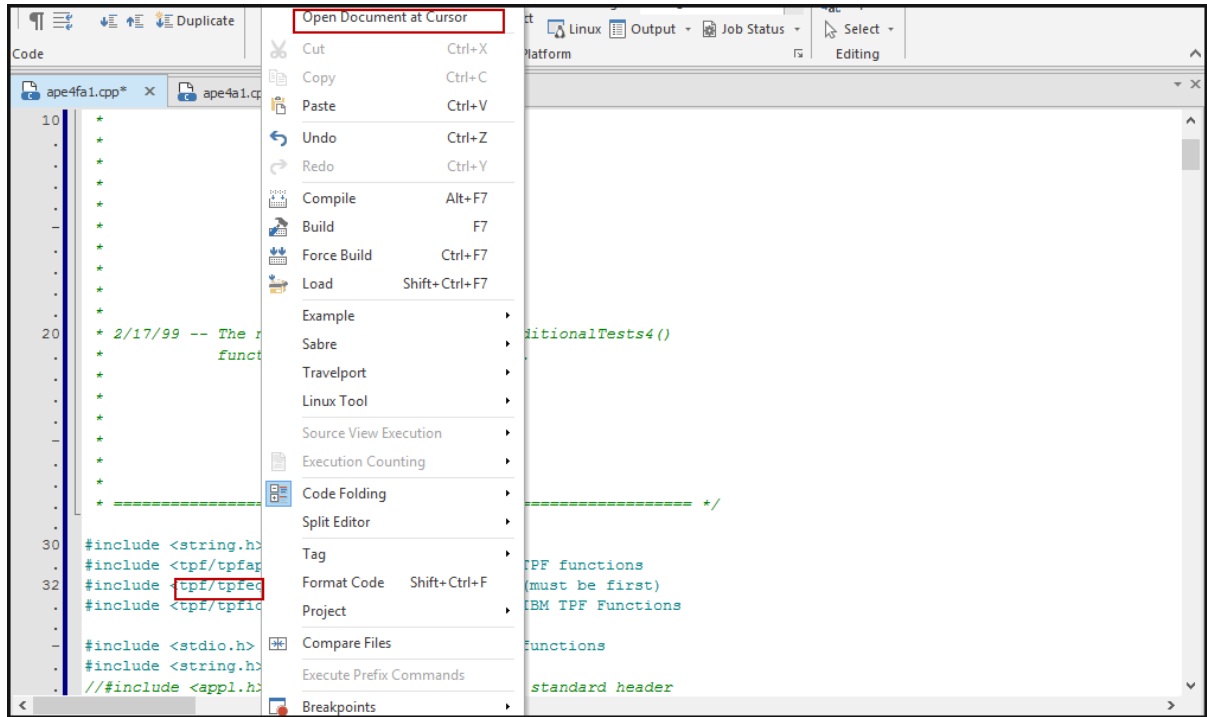
The parameter tooltips can be shown by pressing SHIFT+CTRL+SPACEBAR while the cursor is in the source code text.



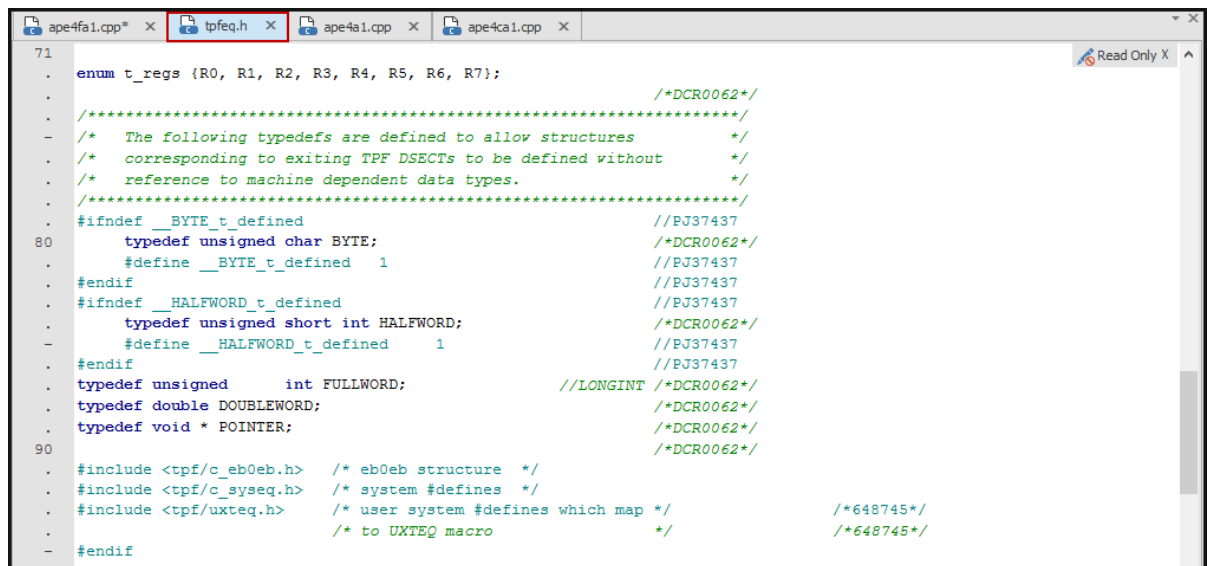
```
static short fct2(unsigned char copya, short copyb, int copyc,
                unsigned char *pd);
static void testLongLongS(char* label, t_Contact* pContact);
static void printContact1(char *label, t_Contact *pContact);
static void printContact2(char *label, t_Contact pContact);
static void printContact3(char *label, t_Contact &pContact);
static void print_cstruct2_s(char *label,
                             Counter::cstruct2_s *pcstruct2_s_parm);
c_Rolodex *Redefine(c_Rolodex *rolol, t_DoubleArray dbl_array,
                  t_MyDoubleArray mydbl_array, t_ULongArray ul_array);
```

Imports Navigation

This feature enables you to quickly and easily navigate to header files and open them in the Editor area. The header files can be shown by placing the cursor on the header file name in the #include statement, right-clicking and then selecting the **Open Document at Cursor** option.



The respective header file displays



These instructions will help you get the most benefit from using the feature – **Code Sense** in zTPFGI.

Feel free to contact us with questions or comments.

Sincerely,
 TPF Software Inc.
 3737 Glenwood Avenue, Suite 100, Raleigh, NC 27612, USA
 Website: www.tpfsoftware.com