



## So Many Tasks, So Little Time...

*by Jerry Karasz*

As a BBJ® developer, you have a job to get done. Or, more likely, you have a number of jobs to get done. All of them are important, but you can't work on all of them at one time. The pressures of an upcoming release and the need to "do more with less" sometimes result in you being forced to leave a task incomplete - with the intention of coming back later to finish it off, of course. A task might be something as simple as a reminder to add comments to a particular section of code, or it might be a large task such as needing to add an email notification module. So a task represents something that needs to be done - perhaps associated with a particular spot in code, or perhaps not. Can the Business BASIC Development Tools (BDT) help?

### Eclipse Tasks to the Rescue!

The important questions are: If there is a location in the code for the task, how do you find that location when you are ready to work there? And whether or not there is a location, where do you look for a reminder that the task is waiting to be done, and to find the details of that task? The answers are simple: use Eclipse tasks. There are two types of tasks in Eclipse: BDT tasks, which are associated with a specific line of code in a BBJ source file, and generic tasks, which are not associated with any specific location in any file. In both cases, you define a text description for the task to remind you of any details or requirements.

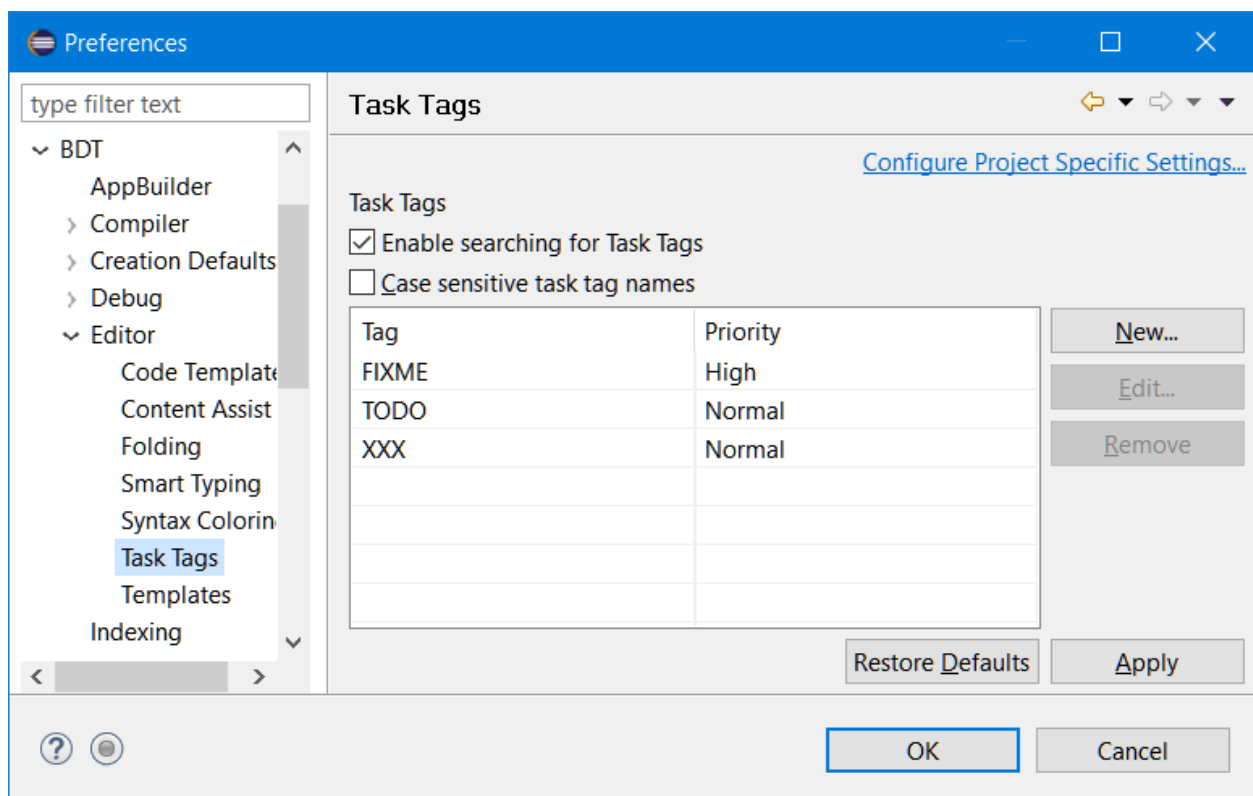
We're going to look at both types of tasks, but we'll focus on the BDT tasks and the tags that you put in BBJ code to mark their locations. We'll see how to find all of your tasks in one list (the Tasks view), and how to view, search, and filter your tasks to help you manage them, including having the ability to jump directly to any task from the list.

This article will help developers like you learn how to use Eclipse's tasks to be more efficient in responding to daily priorities without allowing important assignments to fall through the cracks. It provides examples of how to create and manage both types of tasks so that you can put them to work for you in your development work.

To demonstrate, we'll start with an example of a BDT task where we add a BDT task tag to a comment in a BBj source file. TaskTags.bbj is a representative file created just for this article, but the principles we discuss will apply equally well in your BDT development.

## What Is a BDT Task Tag?

A BDT task tag is simply a piece of text that you define in your BDT Preferences so that you can add it to comments in your code, marking the code as the location for a BDT task. There are a few limitations on what text can be used as a task tag, but in general you can choose text that is meaningful to you, and you can have as many as you need (well, okay, you probably don't want to make over a couple of hundred since BDT has to find them all, and you won't want to wait every time it searches). There are some BDT task tags that BDT creates for you as part of its installation. Let's look at that list first. From Eclipse's main menu select **Window > Preferences**, and navigate in the left side of the window to **BDT > Editor > Task Tags** as seen in **Figure 1**.



**Figure 1:** The Task Tags display in BDT preferences

The FIXME, TODO, and XXX task tags are available for you to place in your code without any further effort on your part. The first thing you may notice is that each BDT task tag has a priority associated with it. In **Figure 1** you can see that FIXME is High priority, while TODO and XXX are Normal. There are no Low priority BDT task tags defined yet.

Here are the actions you can perform to manage the available list of tags that BDT will recognize in your code as marking a BDT task:

- To delete task tags, select them in the list and click the **[Remove]** button.
- To change a task tag's text or priority, select it in the list and click the **[Edit...]** button, make your changes, and click **[OK]**.
- To add a new task tag, click the **[New...]** button, set the text and priority, and click **[OK]**. Task tags can contain almost any printable ASCII characters (including characters such as ü, ß, ñ, €, and Ä), although whitespace characters like spaces, tabs, carriage returns, and line feeds are not valid.
- To disable all of your task tags without deleting them, simply uncheck the **Enable searching for Task Tags** checkbox at the top of the Preferences window.
- To have BDT recognize your tags' text in a comment only if the case matches exactly, check the **Case sensitive task tag names** checkbox at the top of the Preferences window. By default, BDT does a case-insensitive match so that todo, TODO, and ToDo all match the TODO tag shown.

Finally, remember to click the **[Apply]** button or click the **[OK]** button and dismiss the Preferences window in order to save your changes.

## Using BDT Task Tags to Create Tasks

Now that you know that TODO is a valid BDT task tag, let's use it in code. When you are editing a BBj source file, create a reminder there by simply adding a REM comment line like this:

```
REM TODO Refactor this to follow the coding guidelines
```

One thing to remember: the task tag (in this case, TODO) can be placed anywhere in a comment line, with the following limitations:

- The official "Description" for the task starts at the task tag and continues to the right to the end of the line. In other words, text that precedes the task tag is not part of the task Description.
- In order to be recognized as a task tag, the text must be delimited by whitespace. For example, although these comment lines will be detected as having a TODO task tag:

```
REM TODO Remember...
```

```
REM this is not part of the task TODO Remember...
```

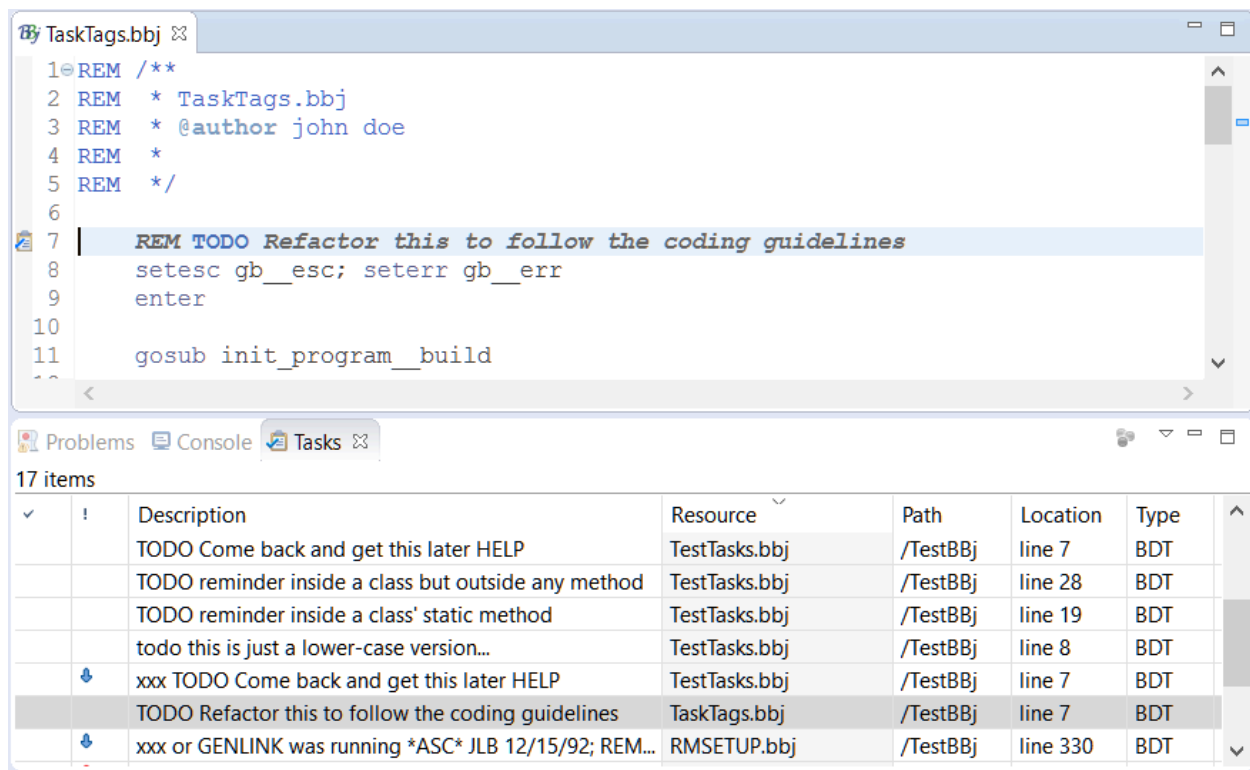
These should not:

```
REM xTODO Remember...
```

```
REM TODO: Remember...
```

```
REM --TODO-- Remember...
```

**Figure 2** shows the BBj CodeEditor after saving the code changes.



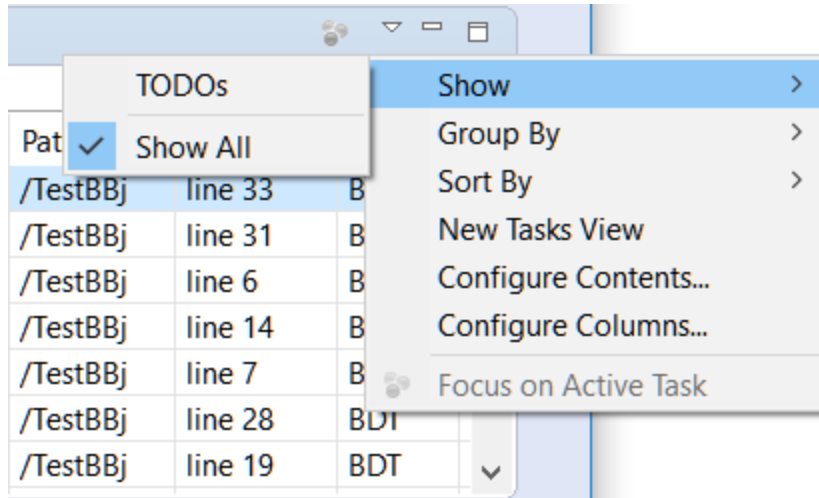
**Figure 2:** A TODO BDT task and its task tag in a BBj source file

In the code editor you see a tag marker in the far left column on line 7 (the REM TODO line). Hover over this marker to see the same text as is shown in the Description column of the Tasks view, “TODO Refactor this to follow the coding guidelines”. The tag marker allows you to visually locate task tags in your code as you scroll through it, but to see them you must have the code open in a BBj CodeEditor. If you have very many files to search, this method may not be fast enough, so let’s look at how to use the Tasks view to manage your tasks.

## Managing Tasks

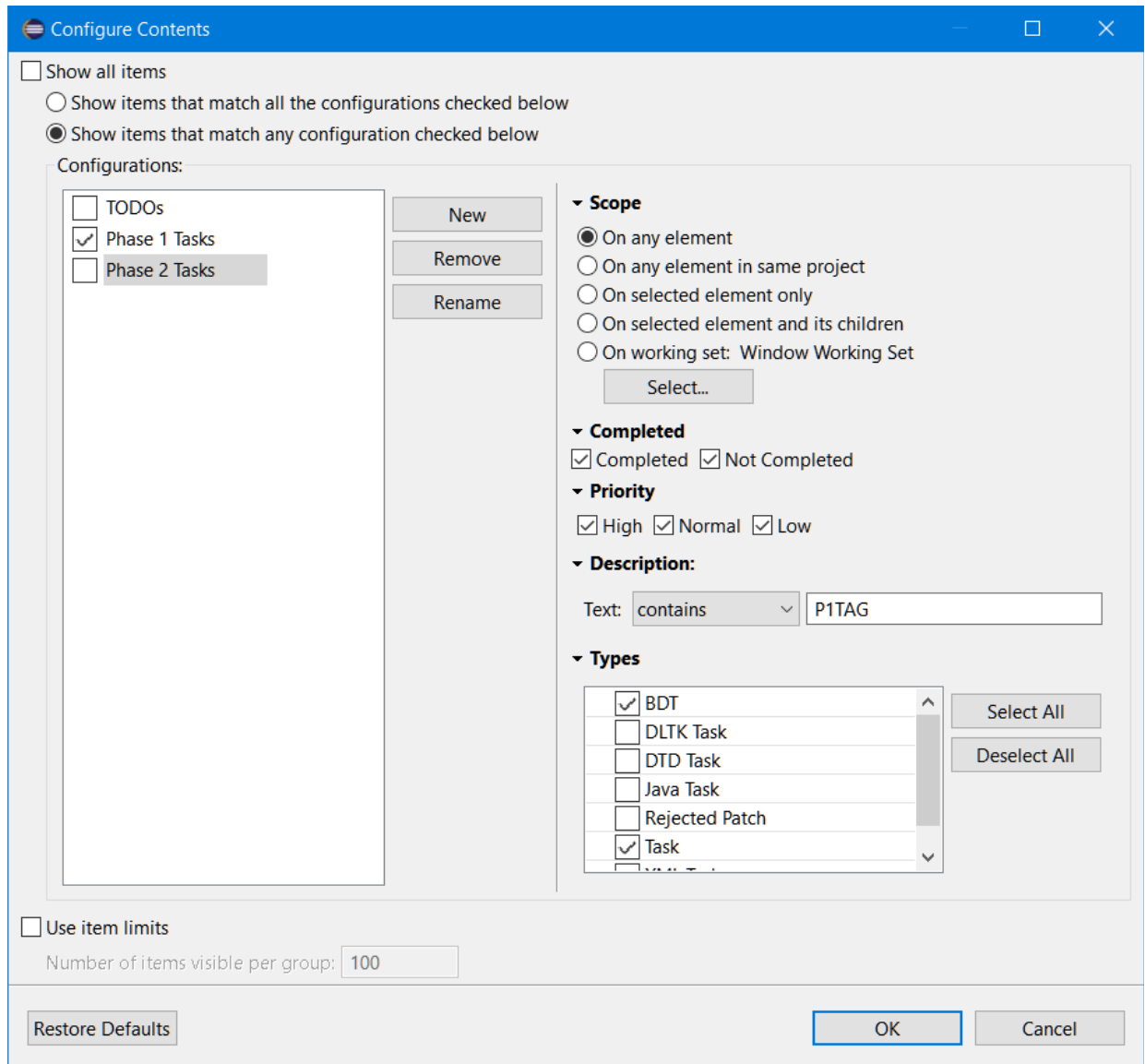
If the Tasks view is not already open, open it by selecting **Window > Show View > Other...** from Eclipse’s main menu, selecting **General > Tasks**, and clicking **[OK]**. You will see the Tasks view as shown at the bottom of **Figure 2**. The first thing you might notice is the list of tasks. If you have a very large number of tasks, you may want to filter the list or change what is being displayed to suit your own needs. These choices are available:

- In the Tasks view’s far right title bar area, click the down-triangle to choose from the view menu. The options here are numerous, and you can explore them on your own, but some of the options appear in **Figure 3**:



**Figure 3:** The Tasks view's View Menu

The most significant option here is **Configure Contents...** Select this to open a **Configure Contents** display (see **Figure 4** below) where you can control exactly which tasks to show and how many of them (in case the default maximum of 100 is too small for you):



**Figure 4:** The Tasks view's Configure Contents display

- To clean up your list by removing all of your completed generic tasks (more on that later), you can right-click in the Tasks view and select **Delete Completed Tasks**. You should note that this is not an undoable action, though, so once generic tasks are deleted they are gone. **NOTE:** BDT tasks can only be deleted or cleaned up by manually removing the task tag from the comment in the code.

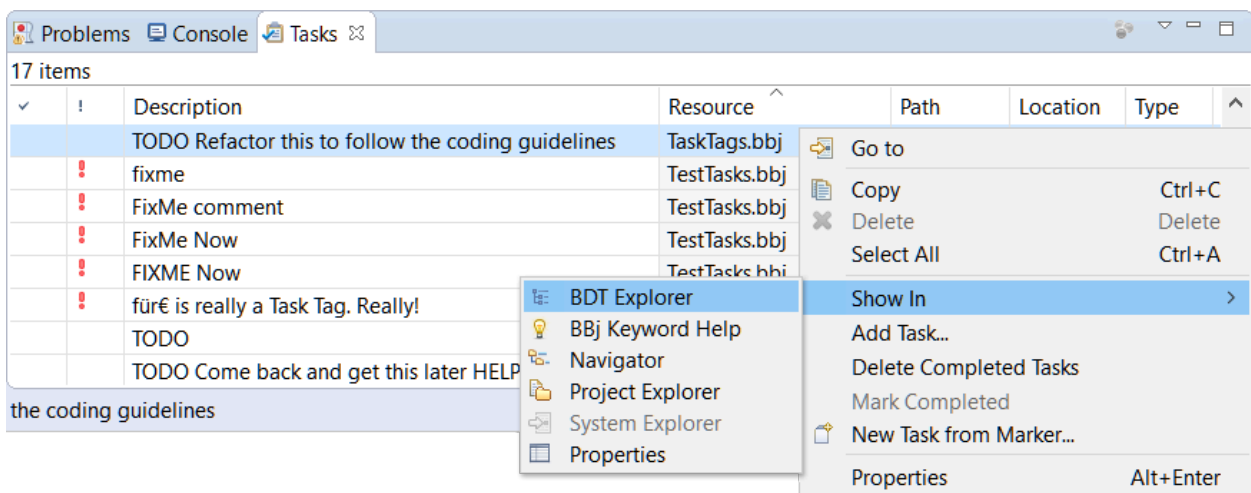
Here are some useful options to navigate directly to an individual task tag:

- Double-click any task to have BDT open an editor on the appropriate file and to find and highlight the text shown in the Description column. If you have the **[Link with Editor]** button depressed in the BDT Explorer as shown in **Figure 5**, then the BDT Explorer will also scroll as necessary and select the file containing the task tag.



**Figure 5:** BDT Explorer's [Link With Editor] button (Hover to see the name)

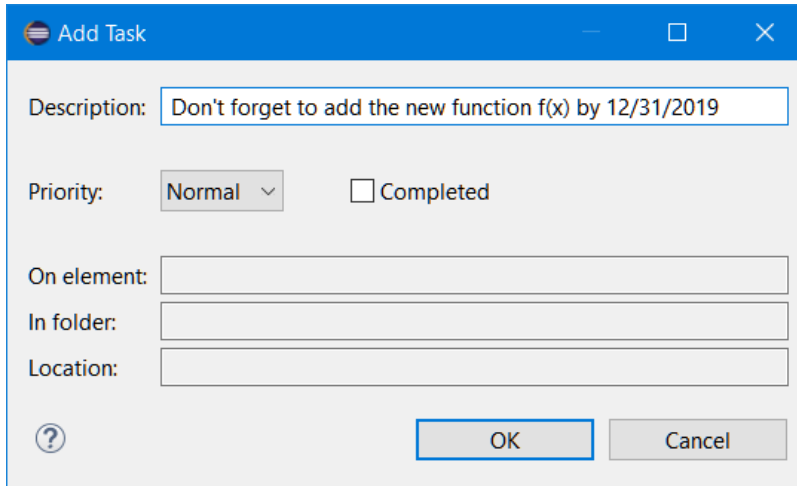
- Right-click any task and select **Go to**. This has the same effect as double-clicking the task.
- Right-click any task and select **Show In > BDT Explorer** as shown in **Figure 6**. This causes the BDT Explorer to scroll as necessary and select the file containing the task tag, but avoids opening the file and selecting the task tag in an editor.



**Figure 6:** Locating a task tag in the BDT Explorer from the Tasks view

There are also some other general Tasks view features you may find useful:

- Right-click one or more tasks and select **Copy**. This copies the text displayed for the selected tasks into the cut-and-paste buffer so that you can paste it elsewhere as text.
- Right-click a task and select **New Task from Marker....** This will allow you to add that task to a task repository such as Mylyn. Without a task repository, this option is not helpful.
- Even if the Tasks view is empty, you can right-click and select **Add task...** to open the Add Task window. Here you can create an arbitrary task (one that is not tied to a BDT task tag or to a specific location in one of your files, such as a general reminder that you need to add a new function someday) as shown in **Figure 7**.



**Add Task**

Description:

Priority: Normal ☐ Completed

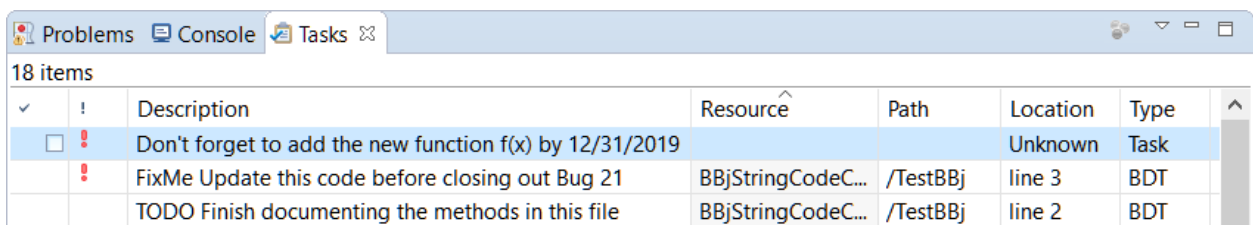
On element:

In folder:

Location:

**Figure 7:** Adding a generic task from the Tasks view

Once you add the new task, it appears in the Tasks view as shown in **Figure 8**:

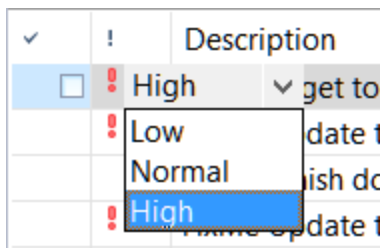


| ✓                        | ! | Description                                             | Resource          | Path     | Location | Type |
|--------------------------|---|---------------------------------------------------------|-------------------|----------|----------|------|
| <input type="checkbox"/> | ! | Don't forget to add the new function f(x) by 12/31/2019 |                   |          | Unknown  | Task |
|                          | ! | FixMe Update this code before closing out Bug 21        | BBjStringCodeC... | /TestBBj | line 3   | BDT  |
|                          |   | TODO Finish documenting the methods in this file        | BBjStringCodeC... | /TestBBj | line 2   | BDT  |

**Figure 8:** A generic task displayed in the Tasks view

There are some obvious differences between tasks based on a BDT task tag and generic tasks created in this manner. Many of these can be seen in **Figure 8**.

- There is a checkbox in the far left column (hover over the column header icon to find out that this is the “Completion” column). Check this box when the task is complete. BDT tasks are considered completed when they are removed from the code; generic tasks, however, can be marked as completed or not, allowing you to tell those that you are still working on from those that you have finished.
- If you click on a generic task’s icon in the “!” column (the Priority column), you will discover that you can change the priority to Low, Normal, or High as shown in **Figure 9**.



**Figure 9:** Setting the priority on a generic task



This task has a red “!” icon because we set it to be High priority when we created it. Normal priority tasks show no priority icon, and Low priority tasks show a blue downward-pointing arrow.

- Alternatively, you could edit the Completion and Priority values by right-clicking the task and selecting **Properties** from the context menu.
- You can click in the Description column and edit the text.
- The Resource, Path, and Location columns are empty. Generic tasks are not associated with a specific file.
- The Type column shows “Task” rather than “BDT”. You can click on the “Type” column header to sort by Type, bringing all of the tasks of one type together.

## Advanced Options for Managing Tasks

When you are ready to take your task management to the next level, you may want to use an advanced tool such as [Mylyn](#) (which is available as a plug-in for Eclipse). Task management tools allow you to store your tasks in shared repositories, either standalone or in life-cycle management tools like Bugzilla. Using shared repositories can provide central backup and management for an entire team’s tasks.

## Being more Productive by Using Tasks

We looked at the two types of tasks, and how to create and manage them. We examined the Tasks view and learned how to sort and filter its list so that we can find the right task when we are ready to work on it. And finally we found out how to mark generic tasks as completed when they are done, and how to delete BDT tasks when they are no longer needed. These simple steps will help you remain focused as you work through your daily development work and allow you to avoid losing track of important tasks (“I remember there was something I was supposed to go back and do but ...”). Build these into your routines, and when they become habits you will find that you can stay focused longer and accomplish more with less interruptions - just add a task and get back to what you were doing. Just don’t forget to come back later and use the Tasks view as a list of reminders - correctly prioritized and annotated, of course!