

:mod:`bdb` --- Debugger framework

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 1); [backlink](#)

Unknown interpreted text role "mod".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 4)

Unknown directive type "module".

```
.. module:: bdb
   :synopsis: Debugger framework.
```

Source code: :source:`Lib/bdb.py`

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 7); [backlink](#)

Unknown interpreted text role "source".

The `:mod:`bdb`` module handles basic debugger functions, like setting breakpoints or managing execution via the debugger.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 11); [backlink](#)

Unknown interpreted text role "mod".

The following exception is defined:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 16)

Unknown directive type "exception".

```
.. exception:: BdbQuit

   Exception raised by the :class:`Bdb` class for quitting the debugger.
```

The `:mod:`bdb`` module also defines two classes:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 21); [backlink](#)

Unknown interpreted text role "mod".

This class implements temporary breakpoints, ignore counts, disabling and (re-)enabling, and conditionals.

Breakpoints are indexed by number through a list called `:attr:`bpbynumber`` and by (file, line) pairs through `:attr:`bplist``. The former points to a single instance of class `:class:`Breakpoint``. The latter points to a list of such instances since there may be more than one breakpoint per line.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 28); [backlink](#)

Unknown interpreted text role "attr".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 28); [backlink](#)

Unknown interpreted text role "attr".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 28); [backlink](#)

Unknown interpreted text role "class".

When creating a breakpoint, its associated filename should be in canonical form. If a *funcname* is defined, a breakpoint hit will be counted when the first line of that function is executed. A conditional breakpoint always counts a hit.

`:class: Breakpoint` instances have the following methods:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 38); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 40)

Unknown directive type "method".

```
.. method:: deleteMe()
```

Delete the breakpoint from the list associated to a file/line. If it is the last breakpoint in that position, it also deletes the entry for the file/line.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 47)

Unknown directive type "method".

```
.. method:: enable()
```

Mark the breakpoint as enabled.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 52)

Unknown directive type "method".

```
.. method:: disable()
```

Mark the breakpoint as disabled.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 57)

Unknown directive type "method".

```
.. method:: bpformat()
```

Return a string with all the information about the breakpoint, nicely formatted:

- * The breakpoint number.
- * If it is temporary or not.
- * Its file, line position.
- * The condition that causes a break.
- * If it must be ignored the next N times.
- * The breakpoint hit count.

```
.. versionadded:: 3.2
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 71)

Unknown directive type "method".

```
.. method:: bpprint(out=None)

Print the output of :meth:`bpformat` to the file *out*, or if it is
`None`, to standard output.
```

The `:class:`Bdb`` class acts as a generic Python debugger base class.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 79); [backlink](#)

Unknown interpreted text role "class".

This class takes care of the details of the trace facility; a derived class should implement user interaction. The standard debugger class (`:class:`pdb.Pdb``) is an example.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 81); [backlink](#)

Unknown interpreted text role "class".

The `skip` argument, if given, must be an iterable of glob-style module name patterns. The debugger will not step into frames that originate in a module that matches one of these patterns. Whether a frame is considered to originate in a certain module is determined by the `__name__` in the frame globals.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 91)

Unknown directive type "versionadded".

```
.. versionadded:: 3.1
   The *skip* argument.
```

The following methods of `:class:`Bdb`` normally don't need to be overridden.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 94); [backlink](#)

Unknown interpreted text role "class".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 96)

Unknown directive type "method".

```
.. method:: canonic(filename)

Auxiliary method for getting a filename in a canonical form, that is, as a
case-normalized (on case-insensitive filesystems) absolute path, stripped
of surrounding angle brackets.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 102)

Unknown directive type "method".

```
.. method:: reset()

Set the :attr:`botframe`, :attr:`stopframe`, :attr:`returnframe` and
:attr:`quitting` attributes with values ready to start debugging.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 107)

Unknown directive type "method".

```
.. method:: trace_dispatch(frame, event, arg)
```

This function is installed as the trace function of debugged frames. Its return value is the new trace function (in most cases, that is, itself).

The default implementation decides how to dispatch a frame, depending on the type of event (passed as a string) that is about to be executed. *event* can be one of the following:

- * ``"line"``: A new line of code is going to be executed.
- * ``"call"``: A function is about to be called, or another code block entered.
- * ``"return"``: A function or other code block is about to return.
- * ``"exception"``: An exception has occurred.
- * ``"c_call"``: A C function is about to be called.
- * ``"c_return"``: A C function has returned.
- * ``"c_exception"``: A C function has raised an exception.

For the Python events, specialized functions (see below) are called. For the C events, no action is taken.

The *arg* parameter depends on the previous event.

See the documentation for :func:`sys.settrace` for more information on the trace function. For more information on code and frame objects, refer to :ref:`types`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 134)

Unknown directive type "method".

```
.. method:: dispatch_line(frame)
```

If the debugger should stop on the current line, invoke the :meth:`user_line` method (which should be overridden in subclasses). Raise a :exc:`BdbQuit` exception if the :attr:`Bdb.quitting` flag is set (which can be set from :meth:`user_line`). Return a reference to the :meth:`trace_dispatch` method for further tracing in that scope.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 142)

Unknown directive type "method".

```
.. method:: dispatch_call(frame, arg)
```

If the debugger should stop on this function call, invoke the :meth:`user_call` method (which should be overridden in subclasses). Raise a :exc:`BdbQuit` exception if the :attr:`Bdb.quitting` flag is set (which can be set from :meth:`user_call`). Return a reference to the :meth:`trace_dispatch` method for further tracing in that scope.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 150)

Unknown directive type "method".

```
.. method:: dispatch_return(frame, arg)
```

If the debugger should stop on this function return, invoke the :meth:`user_return` method (which should be overridden in subclasses). Raise a :exc:`BdbQuit` exception if the :attr:`Bdb.quitting` flag is set (which can be set from :meth:`user_return`). Return a reference to the :meth:`trace_dispatch` method for further tracing in that scope.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 158)

Unknown directive type "method".

```
.. method:: dispatch_exception(frame, arg)
```

If the debugger should stop at this exception, invokes the :meth:`user_exception` method (which should be overridden in subclasses).

```
Raise a :exc:`BdbQuit` exception if the :attr:`Bdb.quitting` flag is set
(which can be set from :meth:`user_exception`). Return a reference to the
:meth:`trace_dispatch` method for further tracing in that scope.
```

Normally derived classes don't override the following methods, but they may if they want to redefine the definition of stopping and breakpoints.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 169)

Unknown directive type "method".

```
.. method:: stop_here(frame)
```

This method checks if the `*frame*` is somewhere below `:attr:`botframe`` in the call stack. `:attr:`botframe`` is the frame in which debugging started.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 174)

Unknown directive type "method".

```
.. method:: break_here(frame)
```

This method checks if there is a breakpoint in the filename and line belonging to `*frame*` or, at least, in the current function. If the breakpoint is a temporary one, this method deletes it.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 180)

Unknown directive type "method".

```
.. method:: break_anywhere(frame)
```

This method checks if there is a breakpoint in the filename of the current frame.

Derived classes should override these methods to gain control over debugger operation.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 188)

Unknown directive type "method".

```
.. method:: user_call(frame, argument_list)
```

This method is called from `:meth:`dispatch_call`` when there is the possibility that a break might be necessary anywhere inside the called function.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 194)

Unknown directive type "method".

```
.. method:: user_line(frame)
```

This method is called from `:meth:`dispatch_line`` when either `:meth:`stop_here`` or `:meth:`break_here`` yields ```True```.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 199)

Unknown directive type "method".

```
.. method:: user_return(frame, return_value)
```

This method is called from `:meth:`dispatch_return`` when `:meth:`stop_here``

```
yields ``True``.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 204)

Unknown directive type "method".

```
.. method:: user_exception(frame, exc_info)
```

This method is called from :meth:`dispatch_exception` when
:meth:`stop_here` yields ``True``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 209)

Unknown directive type "method".

```
.. method:: do_clear(arg)
```

Handle how a breakpoint must be removed when it is a temporary one.

This method must be implemented by derived classes.

Derived classes and clients can call the following methods to affect the stepping state.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 219)

Unknown directive type "method".

```
.. method:: set_step()
```

Stop after one line of code.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 223)

Unknown directive type "method".

```
.. method:: set_next(frame)
```

Stop on the next line in or below the given frame.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 227)

Unknown directive type "method".

```
.. method:: set_return(frame)
```

Stop when returning from the given frame.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 231)

Unknown directive type "method".

```
.. method:: set_until(frame)
```

Stop when the line with the line no greater than the current one is
reached or when returning from current frame.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 236)

Unknown directive type "method".

```
.. method:: set_trace([frame])
```

Start debugging from **frame**. If **frame** is not specified, debugging starts from caller's frame.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 241)

Unknown directive type "method".

```
.. method:: set_continue()
```

Stop only at breakpoints or when finished. If there are no breakpoints, set the system trace function to ``None``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 246)

Unknown directive type "method".

```
.. method:: set_quit()
```

Set the :attr:`quitting` attribute to ``True``. This raises :exc:`BdbQuit` in the next call to one of the :meth:`dispatch_` methods.

Derived classes and clients can call the following methods to manipulate breakpoints. These methods return a string containing an error message if something went wrong, or `None` if all is well.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 256)

Unknown directive type "method".

```
.. method:: set_break(filename, lineno, temporary=0, cond, funcname)
```

Set a new breakpoint. If the **lineno** line doesn't exist for the **filename** passed as argument, return an error message. The **filename** should be in canonical form, as described in the :meth:`canonic` method.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 262)

Unknown directive type "method".

```
.. method:: clear_break(filename, lineno)
```

Delete the breakpoints in **filename** and **lineno**. If none were set, an error message is returned.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 267)

Unknown directive type "method".

```
.. method:: clear_bpbynumber(arg)
```

Delete the breakpoint which has the index **arg** in the :attr:`Breakpoint.bpbynumber`. If **arg** is not numeric or out of range, return an error message.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 273)

Unknown directive type "method".

```
.. method:: clear_all_file_breaks(filename)
```

Delete all breakpoints in **filename**. If none were set, an error message

is returned.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 278)

Unknown directive type "method".

```
.. method:: clear_all_breaks()

Delete all existing breakpoints.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 282)

Unknown directive type "method".

```
.. method:: get_bpbynumber(arg)

Return a breakpoint specified by the given number. If *arg* is a string,
it will be converted to a number. If *arg* is a non-numeric string, if
the given breakpoint never existed or has been deleted, a
:exc:`ValueError` is raised.

.. versionadded:: 3.2
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 291)

Unknown directive type "method".

```
.. method:: get_break(filename, lineno)

Check if there is a breakpoint for *lineno* of *filename*.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 295)

Unknown directive type "method".

```
.. method:: get_breaks(filename, lineno)

Return all breakpoints for *lineno* in *filename*, or an empty list if
none are set.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 300)

Unknown directive type "method".

```
.. method:: get_file_breaks(filename)

Return all breakpoints in *filename*, or an empty list if none are set.
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 304)

Unknown directive type "method".

```
.. method:: get_all_breaks()

Return all breakpoints that are set.
```

Derived classes and clients can call the following methods to get a data structure representing a stack trace.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 312)

Unknown directive type "method".

```
.. method:: get_stack(f, t)
```

Get a list of records for a frame and all higher (calling) and lower frames, and the size of the higher part.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 317)

Unknown directive type "method".

```
.. method:: format_stack_entry(frame_lineno, lprefix=': ')
```

Return a string with information about a stack entry, identified by a ``(frame, lineno)`` tuple:

- * The canonical form of the filename which contains the frame.
- * The function name, or ``"<lambda>"``.
- * The input arguments.
- * The return value.
- * The line of code (if it exists).

The following two methods can be called by clients to use a debugger to debug a `.term`statement``, given as a string

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 329); [backlink](#)

Unknown interpreted text role "term".

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 332)

Unknown directive type "method".

```
.. method:: run(cmd, globals=None, locals=None)
```

Debug a statement executed via the `:func:`exec`` function. `*globals*` defaults to `:attr:`__main__.__dict__``, `*locals*` defaults to `*globals*`.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 337)

Unknown directive type "method".

```
.. method:: runeval(expr, globals=None, locals=None)
```

Debug an expression executed via the `:func:`eval`` function. `*globals*` and `*locals*` have the same meaning as in `:meth:`run``.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 342)

Unknown directive type "method".

```
.. method:: runctx(cmd, globals, locals)
```

For backwards compatibility. Calls the `:meth:`run`` method.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 346)

Unknown directive type "method".

```
.. method:: runcall(func, /, *args, **kwargs)
```

Debug a single function call, and return its result.

Finally, the module defines the following functions:

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 353)

Unknown directive type "function".

```
.. function:: checkfuncname(b, frame)
```

Check whether we should break here, depending on the way the breakpoint *b* was set.

If it was set via line number, it checks if ``b.line`` is the same as the one in the frame also passed as argument. If the breakpoint was set via function name, we have to check we are in the right frame (the right function) and if we are in its first executable line.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 363)

Unknown directive type "function".

```
.. function:: effective(file, line, frame)
```

Determine if there is an effective (active) breakpoint at this line of code. Return a tuple of the breakpoint and a boolean that indicates if it is ok to delete a temporary breakpoint. Return ``(None, None)`` if there is no matching breakpoint.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\ (cpython-main) (Doc) (library)bdb.rst, line 370)

Unknown directive type "function".

```
.. function:: set_trace()
```

Start debugging with a :class:`Bdb` instance from caller's frame.