:mod:`symtable` --- Access to the compiler's symbol tables

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

Unknown directive type "module".

```
.. module:: symtable
   :synopsis: Interface to the compiler's internal symbol tables.
```

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

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

Unknown interpreted text role "source".

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

Unknown directive type "moduleauthor".

```
.. moduleauthor:: Jeremy Hylton <jeremy@alum.mit.edu>
```

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst, **line 12)**

Unknown directive type "sectionauthor".

```
.. sectionauthor:: Benjamin Peterson <benjamin@python.org>
```

Symbol tables are generated by the compiler from AST just before bytecode is generated. The symbol table is responsible for calculating the scope of every identifier in the code. :mod:`symtable` provides an interface to examine these tables.

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst, **line 15); *backlink***

Unknown interpreted text role "mod".

Generating Symbol Tables

System Message: ERROR/3 (D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst, **line 24)**

Unknown directive type "function".

```
.. function:: symtable(code, filename, compile_type)

   Return the toplevel :class:`SymbolTable` for the Python source *code*.
   *filename* is the name of the file containing the code.  *compile_type* is
   like the *mode* argument to :func:`compile`.
```

Examining Symbol Tables

A namespace table for a block. The constructor is not public.

Unknown directive type "method".

```
.. method:: get_type()

   Return the type of the symbol table.  Possible values are ``'class'``,
   ``'module'``, and ``'function'``.
```

Unknown directive type "method".

```
.. method:: get_id()

   Return the table's identifier.
```

Unknown directive type "method".

```
.. method:: get_name()

   Return the table's name.  This is the name of the class if the table is
   for a class, the name of the function if the table is for a function, or
   ``'top'`` if the table is global (:meth:`get_type` returns ``'module'``).
```

Unknown directive type "method".

```
.. method:: get_lineno()

   Return the number of the first line in the block this table represents.
```

Unknown directive type "method".

```
.. method:: is_optimized()

   Return ``True`` if the locals in this table can be optimized.
```

Unknown directive type "method".

```
.. method:: is_nested()

   Return ``True`` if the block is a nested class or function.
```

Unknown directive type "method".

```
.. method:: has_children()

   Return ``True`` if the block has nested namespaces within it.  These can
   be obtained with :meth:`get_children`.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 70)**

Unknown directive type "method".

```
.. method:: get_identifiers()

   Return a list of names of symbols in this table.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 74)**

Unknown directive type "method".

```
.. method:: lookup(name)

   Lookup *name* in the table and return a :class:`Symbol` instance.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 78)**

Unknown directive type "method".

```
.. method:: get_symbols()

   Return a list of :class:`Symbol` instances for names in the table.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 82)**

Unknown directive type "method".

```
.. method:: get_children()

   Return a list of the nested symbol tables.
```

A namespace for a function or method. This class inherits :class:`SymbolTable`.

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 89);** *backlink*

Unknown interpreted text role "class".

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 92)**

Unknown directive type "method".

```
.. method:: get_parameters()

   Return a tuple containing names of parameters to this function.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 96)**

Unknown directive type "method".

```
.. method:: get_locals()

   Return a tuple containing names of locals in this function.
```

System Message: ERROR/3 (`D:\onboarding-resources\sample-onboarding-resources\cpython-main\Doc\library\[cpython-main][Doc][library]symtable.rst`**, line 100)**

Unknown directive type "method".

```
.. method:: get_globals()
```

```
        Return a tuple containing names of globals in this function.
```

```
    .. method:: get_nonlocals()

        Return a tuple containing names of nonlocals in this function.
```

```
    .. method:: get_frees()

        Return a tuple containing names of free variables in this function.
```

A namespace of a class. This class inherits :class:`SymbolTable`.

```
    .. method:: get_methods()

        Return a tuple containing the names of methods declared in the class.
```

An entry in a :class:`SymbolTable` corresponding to an identifier in the source. The constructor is not public.

```
    .. method:: get_name()

        Return the symbol's name.
```

```
    .. method:: is_referenced()

        Return ``True`` if the symbol is used in its block.
```

```
.. method:: is_imported()

   Return ``True`` if the symbol is created from an import statement.
```

```
.. method:: is_parameter()

   Return ``True`` if the symbol is a parameter.
```

```
.. method:: is_global()

   Return ``True`` if the symbol is global.
```

```
.. method:: is_nonlocal()

   Return ``True`` if the symbol is nonlocal.
```

```
.. method:: is_declared_global()

   Return ``True`` if the symbol is declared global with a global statement.
```

```
.. method:: is_local()

   Return ``True`` if the symbol is local to its block.
```

```
.. method:: is_annotated()

   Return ``True`` if the symbol is annotated.

   .. versionadded:: 3.6
```

```
.. method:: is_free()
```

Return ``True`` if the symbol is referenced in its block, but not assigned
to.

Unknown directive type "method".

```
.. method:: is_assigned()

    Return ``True`` if the symbol is assigned to in its block.
```

Unknown directive type "method".

```
.. method:: is_namespace()

    Return ``True`` if name binding introduces new namespace.

    If the name is used as the target of a function or class statement, this
    will be true.

    For example::

        >>> table = symtable.symtable("def some_func(): pass", "string", "exec")
        >>> table.lookup("some_func").is_namespace()
        True

    Note that a single name can be bound to multiple objects.  If the result
    is ``True``, the name may also be bound to other objects, like an int or
    list, that does not introduce a new namespace.
```

Unknown directive type "method".

```
.. method:: get_namespaces()

    Return a list of namespaces bound to this name.
```

Unknown directive type "method".

```
.. method:: get_namespace()

    Return the namespace bound to this name. If more than one or no namespace
    is bound to this name, a :exc:`ValueError` is raised.
```