

# Converting program from GTK3 to GTK4

Original program: `python-help-gui.gtk3.py`

Final program: `python-help-gui.gtk4.py`

Migration Documentation: <https://docs.gtk.org/gtk4/migrating-3to4.html>

Code in: <https://github.com/irsbugs/python-help>

Hamilton Python Users Group

Ian Stewart

8 August 2022

# Converting program from GTK3 to GTK4

## Original GTK3 program

```
1  #!/usr/bin/env python3
2  #!
3  # python-help-gui-gtk3.py
4  #
5  # The python help categories; to
6  # are made selectable with a mou
7  # displayed in a scrollable wind
8  #
9  # Tested on:
10 # ubuntu 20.04 / python 3.8.10
11 # ubuntu 22.04 / python 3.10.4
12 # manjaro 21.2.2 / python 3.10.5
13 #
```

## To become the GTK4 program

```
1  #!/usr/bin/env python3
2  #!
3  # python-help-gui-gtk3_upgrade.py
4  #
5  # The python help categories; top
6  # are made selectable with a mous
7  # displayed in a scrollable windo
8  #
9  # Tested on:
10 # manjaro 21.2.2 / python 3.10.5
```

## Gtk 3.0 to 4.0

Change gi.require\_version to Gtk version 4.0...

```
18 import gi
19 try:
20     gi.require_version("Gtk", "3.0")
21 except ValueError as e:
22     print(e)
23     sys.exit("Unable to run {} program. Exiting...".format(sys.argv[0]))
24 from gi.repository import Gtk, Gdk
```

```
18 import gi
19 try:
20     gi.require_version("Gtk", "4.0")
21 except ValueError as e:
22     print(e)
23     sys.exit("Unable to run {} program. Exiting...".format(sys.argv[0]))
24 from gi.repository import Gtk, Gdk
```

Start running the program and see what errors are detected...

## Gtk 3.0 to 4.0

Gtk.Window.add() replaced .set\_child()

```
Traceback (most recent call last):
```

```
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 268, in <module>
```

```
    win = Window()
```

```
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 57, in __init__
```

```
    self.add(self.grid)
```

```
AttributeError: 'Window' object has no attribute 'add'
```

```
56         self.grid = Gtk.Grid()
```

```
57         self.add(self.grid)
```

```
56         self.grid = Gtk.Grid()
```

```
57         self.add(self.grid)
```

```
58         self.set_child(self.grid)
```

## Gtk 3.0 to 4.0

.set\_border\_width() removed.

```
Traceback (most recent call last):
  File "/home/ian/python-help-gui-gtk3_upgrade.py", line 269, in <module>
    win = Window()
  File "/home/ian/python-help-gui-gtk3_upgrade.py", line 60, in __init__
    self.create_textview()
  File "/home/ian/python-help-gui-gtk3_upgrade.py", line 88, in create_textview
    scrolled_window.set_border_width(10)
AttributeError: 'ScrolledWindow' object has no attribute 'set_border_width'
```

```
88 | scrolled_window.set_border_width(10)
```

```
88 | #scrolled_window.set_border_width(10)
```

Quick fix – comment out the code

## Gtk 3.0 to 4.0

.add\_provider\_for\_screen()...

```
self.create_textview()  
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 89, in create_textview  
    self.set_style()  
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 79, in set_style  
    Gtk.StyleContext.add_provider_for_screen(Gdk.Screen.get_default(),  
AttributeError: type object 'StyleContext' has no attribute 'add_provider_for_screen'. Did you mean: 'add_provider_for_display'?
```

```
79  Gtk.StyleContext.add_provider_for_screen(Gdk.Screen.get_default(),  
80      style_provider,  
81      Gtk.STYLE_PROVIDER_PRIORITY_APPLICATION)
```

...becomes .add\_provider\_for\_display()

```
79  Gtk.StyleContext.add_provider_for_display(Gdk.Screen.get_default(),  
80      style_provider,  
81      Gtk.STYLE_PROVIDER_PRIORITY_APPLICATION)
```

## Gtk 3.0 to 4.0

Gdk.Screen becomes Gdk.Display

```
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 89, in create_textview
    self.set_style()
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 79, in set_style
    Gtk.StyleContext.add_provider_for_display(Gdk.Screen.get_default(),
File "/usr/lib/python3.10/site-packages/gi/overrides/__init__.py", line 32, in
__getattr__
    return getattr(self._introspection_module, name)
File "/usr/lib/python3.10/site-packages/gi/module.py", line 123, in __getattr__
    raise AttributeError("%r object has no attribute %r" % (
AttributeError: 'gi.repository.Gdk' object has no attribute 'Screen'
```

```
79  Gtk.StyleContext.add_provider_for_display(Gdk.Display.get_default(),
80      style_provider,
81      Gtk.STYLE_PROVIDER_PRIORITY_APPLICATION)
```

More of the same...

Gtk 3.0 to 4.0

```
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 96, in create_textview
    scrolled_window.add(self.textview)
```

```
AttributeError: 'ScrolledWindow' object has no attribute 'add'
```

```
96         #scrolled_window.add(self.textview)
97         scrolled_window.set_child(self.textview)
```

```
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 104, in setup_treeview
    scrolled_window.set_border_width(10)
```

```
AttributeError: 'ScrolledWindow' object has no attribute 'set_border_width'
```

```
104         #scrolled_window.set_border_width(10)
```

```
File "/home/ian/python-help-gui-gtk3_upgrade.py", line 138, in setup_treeview
    scrolled_window.add(self.treeview)
```

```
AttributeError: 'ScrolledWindow' object has no attribute 'add'
```

```
138         #scrolled_window.add(self.treeview)
139         scrolled_window.set_child(self.treeview)
```



## Gtk 3.0 to 4.0

Gtk.main\_quit() removed...

```
Traceback (most recent call last):
  File "/home/ian/python-help-gtk3_upgrade.py", line 271, in <module>
    win.connect("destroy", Gtk.main_quit)
  File "/usr/lib/python3.10/site-packages/gi/overrides/__init__.py", line 32, in
__getattr__
    return getattr(self._introspection_module, name)
  File "/usr/lib/python3.10/site-packages/gi/module.py", line 123, in __getattr__
    raise AttributeError("%r object has no attribute %r" % (
AttributeError: 'gi.repository.Gtk' object has no attribute 'main_quit'
```

## Gtk 3.0 to 4.0

Gtk3 way to launch the Window...

```
269     # Start up the window
270     win = Window()
271     win.connect("destroy", Gtk.main_quit)
272     win.show_all()
273     Gtk.main()
```

```
46  class Window(Gtk.Window):
47      def __init__(self):
48          Gtk.Window.__init__(self)
49
50      # Add the title to the window
51      version = sys.version.split(" ")[0]
52      self.set_title("Help for Python version {}".format(version))
53      # Set default window size
54      self.set_default_size(1400, 600)
```

## Gtk 3.0 to 4.0

Gtk4 code to launch “Application()”...

```
269 # Start up the window
270 #win = Window()
271 #win.connect("destroy", Gtk.main_quit)
272 #win.show_all()
273 #Gtk.main()
274
275 # Run the application
276 app = Application()
277 app.run(sys.argv)
```

## Gtk 3.0 to 4.0

Gtk.Application() function that “activates” and “presents” the Window

```
161 class Application(Gtk.Application):
162     ''' Main Application class '''
163     def __init__(self):
164         super().__init__(application_id='gtk4.python.help',
165                          flags=Gio.ApplicationFlags.FLAGS_NONE)
166
167     def do_activate(self):
168         win = self.props.active_window
169         if not win:
170             win = Window(application=self)
171         win.present()
```

# Gtk 3.0 to 4.0

## Gtk.Window

```
46 #class Window(Gtk.Window):
47 #     def __init__(self):
48 #         Gtk.Window.__init__(self)
49 class Window(Gtk.Window):
50     def __init__(self, **kwargs):
51         super(Window, self).__init__(**kwargs)
52
53     # Add the title to the window
54     version = sys.version.split(" ")[0]
```

## Gtk 3.0 to 4.0

Gio not imported...

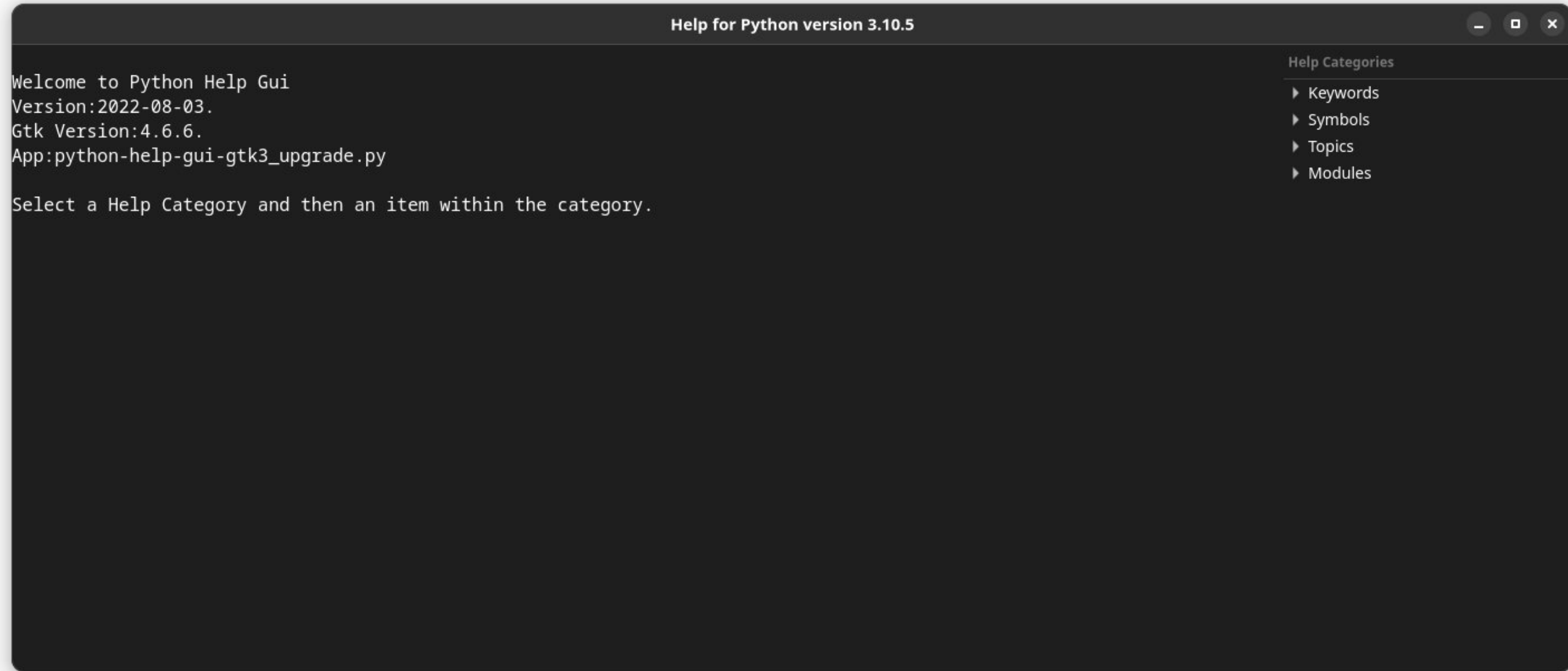
```
Traceback (most recent call last):
  File "/home/ian/python-help-gui-gtk3_upgrade.py", line 291, in <module>
    app = Application()
  File "/home/ian/python-help-gui-gtk3_upgrade.py", line 168, in __init__
    flags=Gio.ApplicationFlags.FLAGS_NONE)
NameError: name 'Gio' is not defined
```

Add Gio to the list of gi.repository imports...

```
18 import gi
19 try:
20     gi.require_version("Gtk", "4.0")
21 except ValueError as e:
22     print(e)
23     sys.exit("Unable to run {} program. Exiting...".format(sys.argv[0]))
24 from gi.repository import Gtk, Gdk, Gio
```

# Gtk 3.0 to 4.0

Program launches using Gtk4 for the first time...



Note that there is no border / margin.

## Gtk 3.0 to 4.0

One method of adding margins, which works but needs 4 lines of code...

```
93     #scrolled_window.set_border_width(10)
94     scrolled_window.set_margin_bottom(10)
95     scrolled_window.set_margin_end(10)
96     scrolled_window.set_margin_start(10)
97     scrolled_window.set_margin_top(10)
```

Alternative method of calling the set\_style function and using CSS...

```
93     #scrolled_window.set_border_width(10)
94     #scrolled_window.set_margin_bottom(10)
95     #scrolled_window.set_margin_end(10)
96     #scrolled_window.set_margin_start(10)
97     #scrolled_window.set_margin_top(10)
98     self.set_style()
```



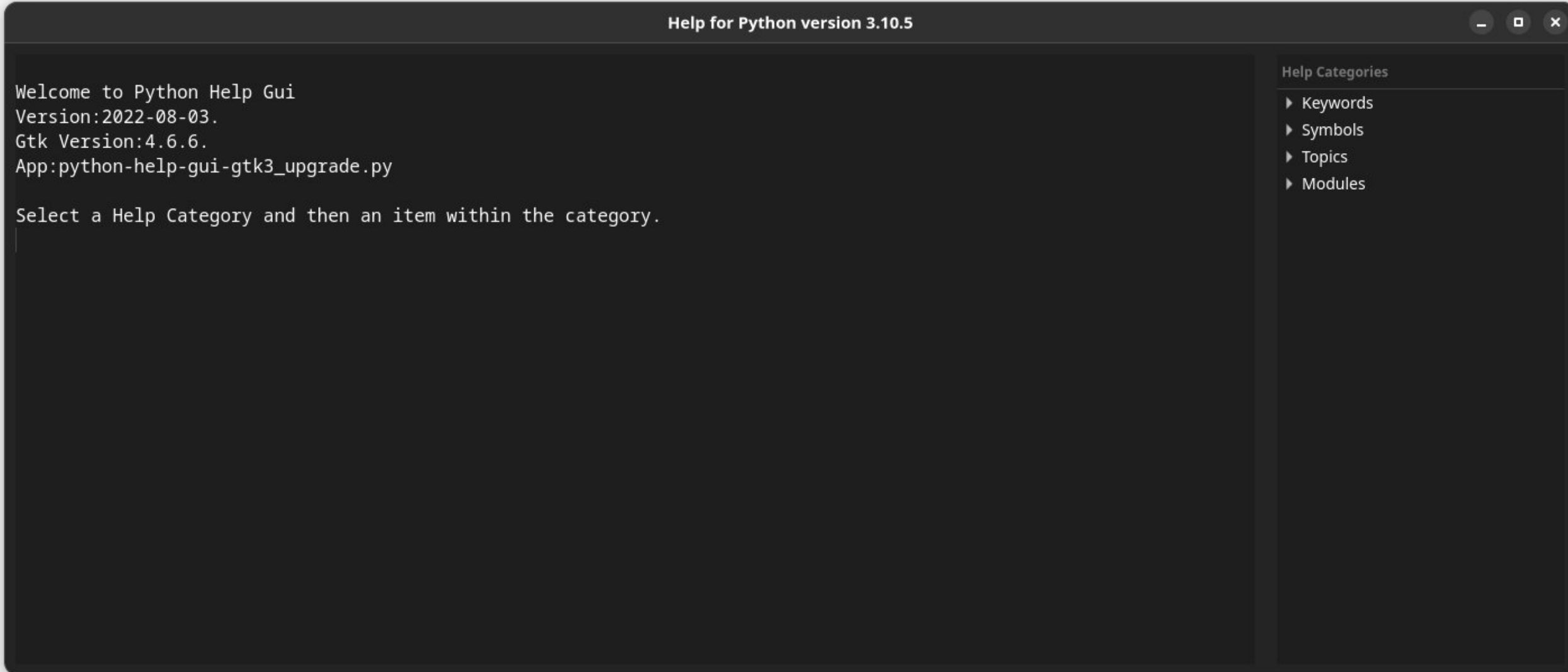
## Gtk 3.0 to 4.0

Add “margin: 10px” to both the Textview and the Treeview widgets...

```
69  def set_style(self):
70      """ Loads custom CSS for textview and treeview """
71      style_provider = Gtk.CssProvider()
72      style_provider.load_from_data(b"""
73      #textview {
74          font: 16px "Monospace";
75          margin: 10px;
76          /*color: #000000;  Black conflicts with dark theme*/
77      }
78      #treeview {
79          font: 14px Sans;
80          margin: 10px;
81          /* border-width: 10px; Around the column header */
82      }
83      """)
```

# Gtk 3.0 to 4.0

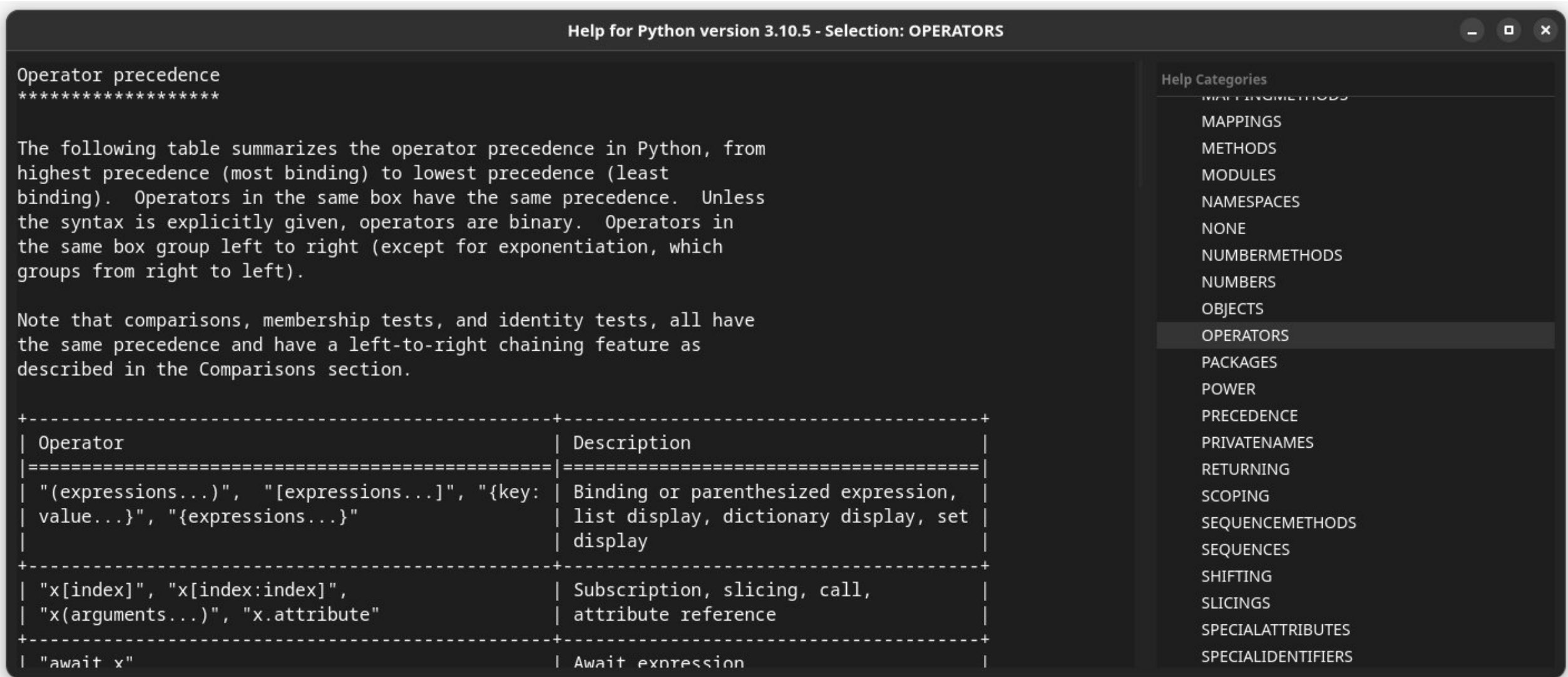
Gtk Version: 4.6.5 and Python Version 3.10.5



...with 10px margins.

# Gtk 3.0 to 4.0

## Example: Displaying the Topic on OPERATORS...



Help for Python version 3.10.5 - Selection: OPERATORS

Operator precedence  
\*\*\*\*\*

The following table summarizes the operator precedence in Python, from highest precedence (most binding) to lowest precedence (least binding). Operators in the same box have the same precedence. Unless the syntax is explicitly given, operators are binary. Operators in the same box group left to right (except for exponentiation, which groups from right to left).

Note that comparisons, membership tests, and identity tests, all have the same precedence and have a left-to-right chaining feature as described in the Comparisons section.

Operator	Description
"(expressions...)", "[expressions...]", "{key: value...}", "{expressions...}"	Binding or parenthesized expression, list display, dictionary display, set display
"x[index]", "x[index:index]", "x(arguments...)", "x.attribute"	Subscription, slicing, call, attribute reference
"await x"	Await expression

Help Categories

- MAPPINGS
- METHODS
- MODULES
- NAMESPACES
- NONE
- NUMBERMETHODS
- NUMBERS
- OBJECTS
- OPERATORS**
- PACKAGES
- POWER
- PRECEDENCE
- PRIVATENAMES
- RETURNING
- SCOPING
- SEQUENCEMETHODS
- SEQUENCES
- SHIFTING
- SLICINGS
- SPECIALATTRIBUTES
- SPECIALIDENTIFIERS

# Gtk 3.0 to 4.0

Display code / Demo program