

```
agnieszka@naboo: ~/pipenv_example
Plik Edycja Widok Wyszukiwanie Terminal Pomoc

agnieszka@naboo:~$ cd pipenv_example/
agnieszka@naboo:~/pipenv_example$ pipenv install pygments
Creating a virtualenv for this project...
Pipfile: /home/agnieszka/pipenv_example/Pipfile
Using /usr/bin/python3 (3.6.5) to create virtualenv...
*Already using interpreter /usr/bin/python3
Using base prefix '/usr'
New python executable in /home/agnieszka/.local/share/virtualenvs/pipenv_example-pMbPa0tC/bin/python3
Also creating executable in /home/agnieszka/.local/share/virtualenvs/pipenv_example-pMbPa0tC/bin/python
Installing setuptools, pip, wheel...done.
Setting project for pipenv_example-pMbPa0tC to /home/agnieszka/pipenv_example

Virtualenv location: /home/agnieszka/.local/share/virtualenvs/pipenv_example-pMbPa0tC
Creating a Pipfile for this project...
Installing pygments...
Collecting pygments
  Downloading https://files.pythonhosted.org/packages/02/ee/b6e02dc6529e82b75bb06823ff7d005b141037cb1416b10c6f00fc419dca/Pygments-2.
2.0-py2.py3-none-any.whl (84kB)
Installing collected packages: pygments
Successfully installed pygments-2.2.0

Adding pygments to Pipfile's [packages]...
Pipfile.lock not found, creating...
Locking [dev-packages] dependencies...
Locking [packages] dependencies...
Updated Pipfile.lock (fea9e8)!
Installing dependencies from Pipfile.lock (fea9e8)...
 1/1 -
To activate this project's virtualenv, run pipenv shell.
Alternatively, run a command inside the virtualenv with pipenv run.
agnieszka@naboo:~/pipenv_example$ pipenv shell
Launching subshell in virtual environment.
agnieszka@naboo:~/pipenv_example$ . /home/agnieszka/.local/share/virtualenvs/pipenv_example-pMbPa0tC/bin/activate
(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$
```

Rysunek 1.1. Utworzenie i aktywowanie środowiska wirtualnego Pythona w nowy sposób

```
agnieszka@naboo: ~/pipenv_example
Plik Edycja Widok Wyszukiwanie Terminal Pomoc

(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$ pip install pygments==2.0
Collecting pygments==2.0
  Using cached https://files.pythonhosted.org/packages/b0/df/f3b9f10d4bdad8c9831ae234972fdd48b14c94bfd800240e0ecb306932a4/Pygments-2.
0-py3-none-any.whl
Installing collected packages: pygments
  Found existing installation: Pygments 2.2.0
    Uninstalling Pygments-2.2.0:
      Successfully uninstalled Pygments-2.2.0
Successfully installed pygments-2.0
You are using pip version 18.0, however version 18.1 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$
```

Rysunek 1.2. Zainstalowanie starszej wersji pakietu pygments

```
#include <Python.h>

int
main(int argc, char *argv[])
{
    wchar_t *program = Py_DecodeLocale(argv[0], NULL);
    if (program == NULL) {
        fprintf(stderr, "Fatal error: cannot decode argv[0]\n");
        exit(1);
    }
    Py_SetProgramName(program); /* optional but recommended */
    Py_Initialize();
    PyRun_SimpleString("from time import time,ctime\n"
                      "print('Today is', ctime(time()))\n");
    if (Py_FinalizeEx() < 0) {
        exit(120);
    }
    PyMem_RawFree(program);
    return 0;
}
```

Rysunek 1.3. Fragment kodu źródłowego Pythona

```
agnieszka@naboo: ~/pipenv_example
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$ pip list
Package            Version
-----
Click              7.0
flake8             3.5.0
Flask              1.0.2
itsdangerous       0.24
Jinja2             2.10
MarkupSafe         1.0
mccabe             0.6.1
pip               18.1
pycodestyle        2.3.1
pyflakes          1.6.0
Pygments          2.0
setuptools         40.4.3
Werkzeug           0.14.1
wheel             0.32.0
(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$
```

Rysunek 1.4. Lista aktualnie zainstalowanych pakietów Pythona

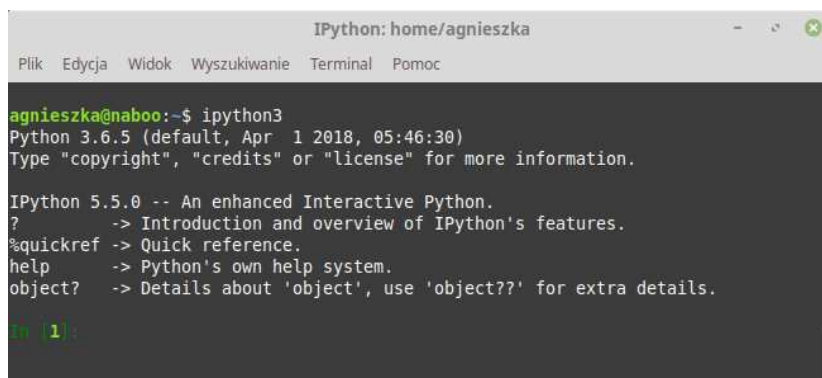
```
agnieszka@naboo: ~/pipenv_example
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$ pip show flask --verbose
Name: Flask
Version: 1.0.2
Summary: A simple framework for building complex web applications.
Home-page: https://www.palletsprojects.com/p/flask/
Author: Armin Ronacher
Author-email: armin.ronacher@active-4.com
License: BSD
Location: /home/agnieszka/.local/share/virtualenvs/pipenv_example-pMbPa0tC/lib/python3.6/site-packages
Requires: itsdangerous, Jinja2, click, Werkzeug
Required-by:
Metadata-Version: 2.1
Installer: pip
Classifiers:
  Development Status :: 5 - Production/Stable
  Environment :: Web Environment
  Framework :: Flask
  Intended Audience :: Developers
  License :: OSI Approved :: BSD License
  Operating System :: OS Independent
  Programming Language :: Python
  Programming Language :: Python :: 2
  Programming Language :: Python :: 2.7
  Programming Language :: Python :: 3
  Programming Language :: Python :: 3.4
  Programming Language :: Python :: 3.5
  Programming Language :: Python :: 3.6
  Topic :: Internet :: WWW/HTTP :: Dynamic Content
  Topic :: Internet :: WWW/HTTP :: WSGI :: Application
  Topic :: Software Development :: Libraries :: Application Frameworks
  Topic :: Software Development :: Libraries :: Python Modules
Entry-points:
  [console scripts]
  flask = flask.cli:main
(pipenv_example-pMbPa0tC) agnieszka@naboo:~/pipenv_example$
```

Rysunek 1.5. Przykład użycia opcji --verbose narzędzia pip

```
#include <Python.h>

int
main(int argc, char *argv[])
{
    wchar_t *program = Py_DecodeLocale(argv[0], NULL);
    if (program == NULL) {
        fprintf(stderr, "Fatal error: cannot decode argv[0]\n");
        exit(1);
    }
    Py_SetProgramName(program); /* optional but recommended */
    Py_Initialize();
    PyRun_SimpleString("from time import time,ctime\n"
                      "print('Today is', ctime(time()))\n");
    if (Py_FinalizeEx() < 0) {
        exit(120);
    }
    PyMem_RawFree(program);
    return 0;
}
```

Rysunek 2.1. Przykład osadzenia Pythona w kodzie C



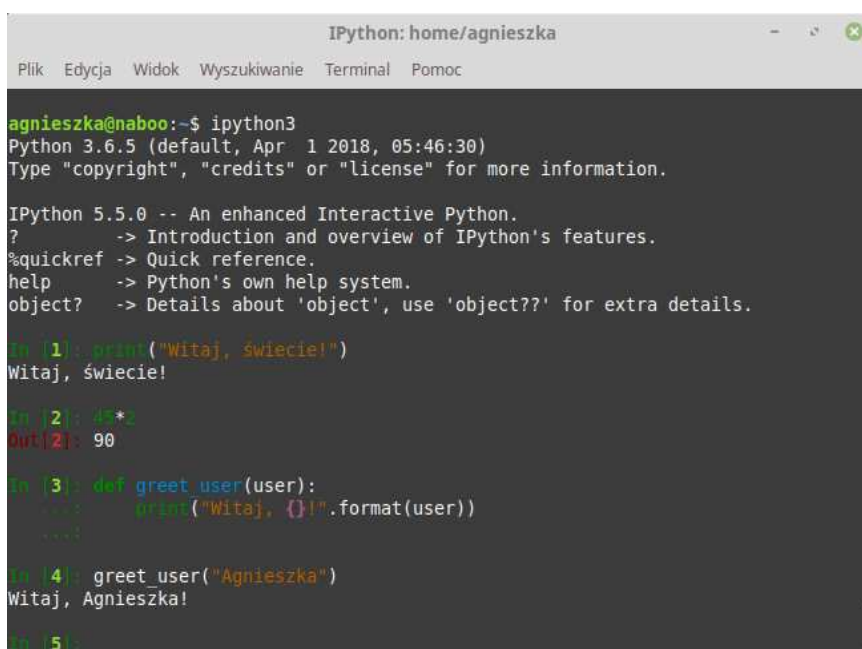
```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

agnieszka@naboo:~$ ipython3
Python 3.6.5 (default, Apr 1 2018, 05:46:30)
Type "copyright", "credits" or "license" for more information.

IPython 5.5.0 -- An enhanced Interactive Python.
? -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.

In [1]:
```

Rysunek 2.2. Uruchomienie sesji interaktywnej IPython



```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

agnieszka@naboo:~$ ipython3
Python 3.6.5 (default, Apr 1 2018, 05:46:30)
Type "copyright", "credits" or "license" for more information.

IPython 5.5.0 -- An enhanced Interactive Python.
? -> Introduction and overview of IPython's features.
%quickref -> Quick reference.
help -> Python's own help system.
object? -> Details about 'object', use 'object??' for extra details.

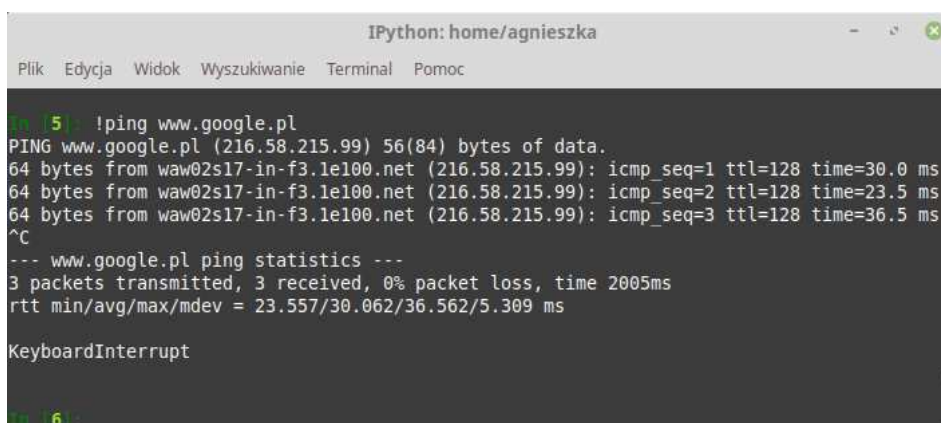
In [1]: print("Witaj, świecie!")
Witaj, świecie!

In [2]: 45*2
Out[2]: 90

In [3]: def greet_user(user):
...:     print("Witaj, {}!".format(user))
...:
In [4]: greet_user("Agnieszka")
Witaj, Agnieszka!

In [5]:
```

Rysunek 2.3. Przykłady poleceń wydanych w powłoce interpretera IPython



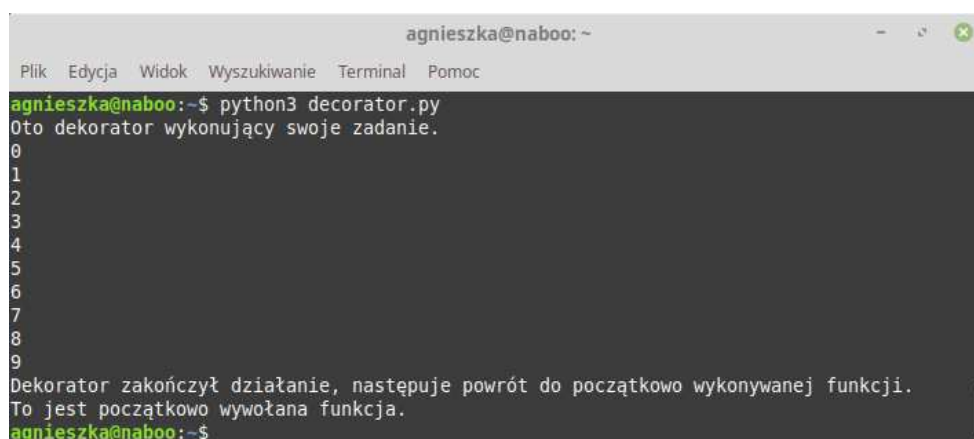
```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

In [5]: !ping www.google.pl
PING www.google.pl (216.58.215.99) 56(84) bytes of data.
64 bytes from waw02s17-in-f3.1e100.net (216.58.215.99): icmp_seq=1 ttl=128 time=30.0 ms
64 bytes from waw02s17-in-f3.1e100.net (216.58.215.99): icmp_seq=2 ttl=128 time=23.5 ms
64 bytes from waw02s17-in-f3.1e100.net (216.58.215.99): icmp_seq=3 ttl=128 time=36.5 ms
^C
--- www.google.pl ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2005ms
rtt min/avg/max/mdev = 23.557/30.062/36.562/5.309 ms

KeyboardInterrupt

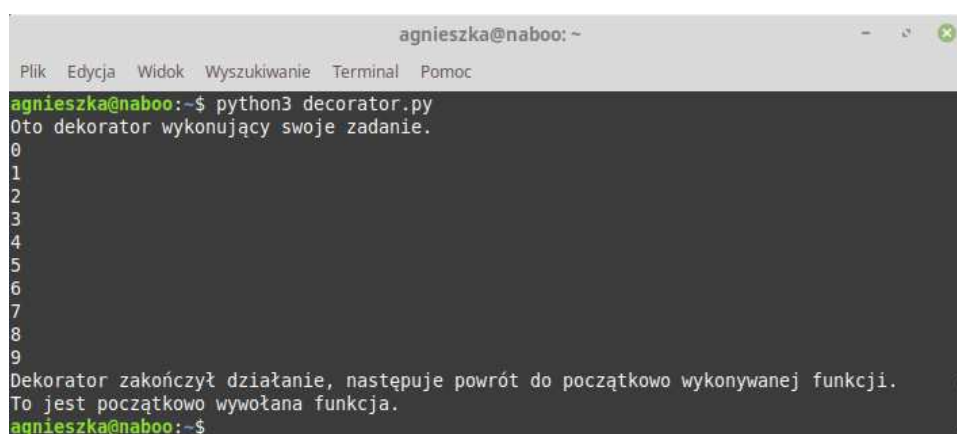
In [6]:
```

Rysunek 2.4. Wykonanie polecenia powłoki bash w powłoce IPython



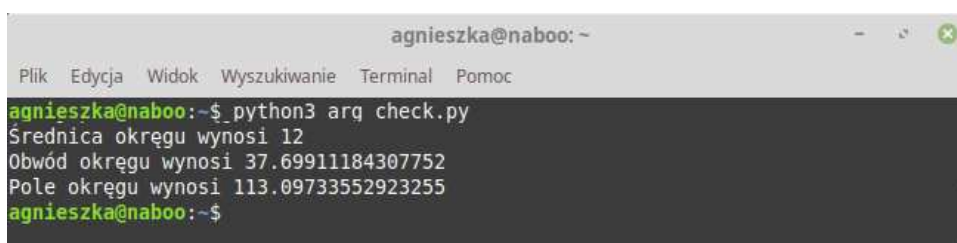
```
agnieszka@naboo: ~
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
agnieszka@naboo:~$ python3 decorator.py
Oto dekorator wykonujący swoje zadanie.
0
1
2
3
4
5
6
7
8
9
Dekorator zakończył działanie, następuje powrót do początkowo wykonywanej funkcji.
To jest początkowo wywołana funkcja.
agnieszka@naboo:~$
```

Rysunek 3.1. Efekt uruchomienia omawianego programu



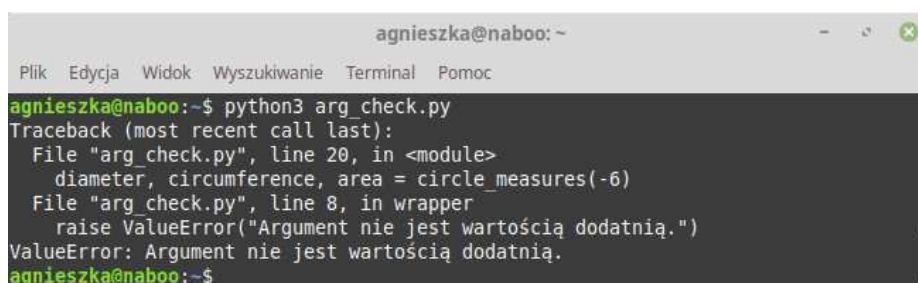
```
agnieszka@naboo: ~
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
agnieszka@naboo:~$ python3 decorator.py
Oto dekorator wykonujący swoje zadanie.
0
1
2
3
4
5
6
7
8
9
Dekorator zakończył działanie, następuje powrót do początkowo wykonywanej funkcji.
To jest początkowo wywołana funkcja.
agnieszka@naboo:~$
```

Rysunek 3.2. Zmodyfikowana wersja programu decorator.py działa identycznie jak poprzednia



```
agnieszka@naboo: ~
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
agnieszka@naboo:~$ python3 arg_check.py
Średnica okręgu wynosi 12
Obwód okręgu wynosi 37.69911184307752
Pole okręgu wynosi 113.09733552923255
agnieszka@naboo:~$
```

Rysunek 3.3. Wynik uruchomienia omawianego programu



```
agnieszka@naboo: ~
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc
agnieszka@naboo:~$ python3 arg_check.py
Traceback (most recent call last):
  File "arg_check.py", line 20, in <module>
    diameter, circumference, area = circle_measures(-6)
  File "arg_check.py", line 8, in wrapper
    raise ValueError("Argument nie jest wartością dodatnią.")
ValueError: Argument nie jest wartością dodatnią.
agnieszka@naboo:~$
```

Rysunek 3.4. Zgłoszenie wyjątku po przekazaniu argumentu w postaci ujemnej liczby całkowitej



```
agnieszka@naboo: ~
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
agnieszka@naboo:~$ python3 arg_check.py
Traceback (most recent call last):
  File "arg_check.py", line 20, in <module>
    diameter, circumference, area = circle_measures(6.0)
  File "arg_check.py", line 6, in wrapper
    raise TypeError("Argument nie jest liczbą całkowitą.")
TypeError: Argument nie jest liczbą całkowitą.
agnieszka@naboo:~$
```

Rysunek 3.5. Zgłoszenie wyjątku po przekazaniu argumentu w postaci innej niż liczba całkowita

```
agnieszka@naboo: ~
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
agnieszka@naboo:~$ python3 arg_check.py
Podaj promień okręgu: 4
Traceback (most recent call last):
  File "arg_check.py", line 21, in <module>
    diameter, circumference, area = circle_measures(r)
  File "arg_check.py", line 6, in wrapper
    raise TypeError("Argument nie jest liczbą całkowitą.")
TypeError: Argument nie jest liczbą całkowitą.
agnieszka@naboo:~$
```

Rysunek 3.6. Błąd spowodowany przez dane wejściowe użytkownika

```
agnieszka@naboo: ~
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
agnieszka@naboo:~$ python3 arg_check.py
Podaj promień okręgu: 6.9
Traceback (most recent call last):
  File "arg_check.py", line 21, in <module>
    diameter, circumference, area = circle_measures(int(r))
ValueError: invalid literal for int() with base 10: '6.9'
agnieszka@naboo:~$
```

Rysunek 3.7. Kolejny problem spowodowany przez omawiany program

```
agnieszka@naboo: ~
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
agnieszka@naboo:~$ python3 time_dec.py
3628800
347875.032659178
agnieszka@naboo:~$
```

Rysunek 3.8. Przykład użycia własnej funkcji dekoratora

```
agnieszka@naboo: ~
Plik Edycja Widok Wyszukiwanie Terminal Pomoc
...
>>> for p in Point(3, 4), Point(14, 5/7):
...     print(p)
...
Point: x= 3.000 y= 4.000 hypot= 5.000
Point: x=14.000 y= 0.714 hypot=14.018
>>>
```

Rysunek 4.1. Przykładowe dane wyjściowe wygenerowane przez podklasę krotki

```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

In [8]: from collections import ChainMap

In [9]: x = {"a": 10, "b": 20}

In [10]: y = {"b": 30, "c": 40}

In [11]: z = ChainMap(y, x)

In [12]: for key, value in z.items():
        print(key, value)
        ...
a 10
b 30
c 40

In [13]: z = ChainMap(x, y)

In [14]: for key, value in z.items():
        print(key, value)
        ...
c 40
a 10
b 20

In [15]:
```

Rysunek 4.2. Przykład użycia obiektu typu ChainMap

```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

In [1]: from collections import ChainMap

In [2]: class DeepChainMap(ChainMap):
        """
        Wersja klasy ChainMap pozwalająca na bezpośrednie uaktualnienie wewnętrznego zakresu.
        """
        def __setitem__(self, key, value):
            for mapping in self.maps:
                if key in mapping:
                    mapping[key] = value
                    return
            self.maps[0][key] = value

        def __delitem__(self, key):
            for mapping in self.maps:
                if key in mapping:
                    del mapping[key]
                    return
            raise KeyError(key)

In [3]: d = DeepChainMap({'zebra': 'czarny'}, {'słoń': 'niebieski'}, {'lew': 'żółty'})

In [4]: d['lew'] = 'pomarańczowy' # Uaktualnienie istniejącego klucza.

In [5]: d['wąż'] = 'czerwony' # Nowy klucz został dodany do słownika.

In [6]: del d['słoń'] # Usunięcie istniejącego klucza.

In [7]: print(d)
DeepChainMap({'zebra': 'czarny', 'wąż': 'czerwony'}, {}, {'lew': 'pomarańczowy'})

In [8]:
```

Rysunek 4.3. Uruchomiony kod zdefiniowany w pliku deep\_chainmap.py

```
agnieszka@naboo: ~
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

>>> a
{'a': 1}
>>> d
{'d': 3}
>>>
```

Rysunek 4.4. Przykład użycia obiektu klasy UserDict

```
IPython: home/agnieszka
Plik Edycja Widok Wyszukiwanie Terminal Pomoc

In [11]: from collections import UserList

In [12]: class ExtendList(UserList):
...:     def __setitem__(self, i, value):
...:         if i == len(self.data):
...:             self.data.append(value)
...:         else:
...:             self.data[i] = value
...:

In [13]: l = ExtendList()

In [14]: for i in range(10):
...:     l[i] = i
...:

In [15]: print(l)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

In [16]: l[10] = 10

In [17]: print(l)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

In [18]: l[1] = 43

In [19]: print(l)
[0, 1, 43, 3, 4, 5, 6, 7, 8, 9, 10]

In [20]: l[12] = 46

Traceback (most recent call last)
<ipython-input-20-677e80caecb0> in <module>()
----> 1 l[12] = 46

<ipython-input-12-330093f428ad> in __setitem__(self, i, value)
      4     self.data.append(value)
      5     else:
----> 6         self.data[i] = value
      7

IndexError: list assignment index out of range

In [21]:
```

Rysunek 4.5. Przykład użycia klasy UserList

```
IPython: home/agnieszka
Plik Edycja Widok Wyszukiwanie Terminal Pomoc

In [1]: import collections

In [2]: bookCatalog = collections.defaultdict(lambda: "Niedostępny")

In [3]: bookCatalog["a"] = "Anglistyka"

In [4]: bookCatalog["b"] = "Biologia"

In [5]: bookCatalog["c"] = "Chemia"

In [6]: bookCatalog["d"] = "Dziennikarstwo"

In [7]: print(bookCatalog)
defaultdict(<function <lambda> at 0x7f7119720488>, {'a': 'Anglistyka', 'b': 'Biologia',
'c': 'Chemia', 'd': 'Dziennikarstwo'})

In [8]: for k in bookCatalog:
...:     print(k, bookCatalog[k])
...:
a Anglistyka
b Biologia
c Chemia
d Dziennikarstwo

In [9]: bookCatalog["z"]
Out[9]: 'Niedostępny'

In [10]: for k in bookCatalog:
...:     print(k, bookCatalog[k])
...:
a Anglistyka
b Biologia
c Chemia
d Dziennikarstwo
z Niedostępny

In [11]:
```

Rysunek 4.6. Uruchomiony przykład zdefiniowany w pliku book\_catalog.py

```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

In [3]: age_groups = {}

In [4]: for person in people:
    age = person.age
    if age in age_groups: # Czy klucz age już istnieje w słowniku?
        age_groups[age].append(person) # Jeżeli tak, trzeba dodać nowy.
    else:
        age_groups[age] = [person] # Dodanie wartości age do słownika.

In [5]: for i in age_groups:
    print(i)

40
18
42
25
23
80
67

In [6]: age_groups.items()
Out[6]: dict_items([(40, [40, 40]), (18, [18, 18, 18]), (42, [42]), (25, [25]), (23, [23]), (80, [80]), (67, [67])])

In [7]: for k in age_groups:
    print(k, age_groups[k])

40 [40, 40]
18 [18, 18, 18]
42 [42]
25 [25]
23 [23]
80 [80]
67 [67]

In [8]:
```

Rysunek 4.7. Przykład grupowania elementów według zmiennej

```
IPython: home/agnieszka
Plik  Edycja  Widok  Wyszukiwanie  Terminal  Pomoc

In [8]: import collections

In [9]: salesReceipt = collections.namedtuple("salesReceipt", ["storeID", "saleDate", "saleAmount", "totalGuests"])

In [10]: store22 = salesReceipt(22, "12-12-2017", 45.32, 3)
In [11]: store15 = salesReceipt(15, "12-12-2017", 22.5, 1)

In [12]: print("Identyfikator sklepu = ", store22.storeID)
Identyfikator sklepu = 22

In [13]: print("Kwota sprzedaży = ", store15.saleAmount)
Kwota sprzedaży = 22.5

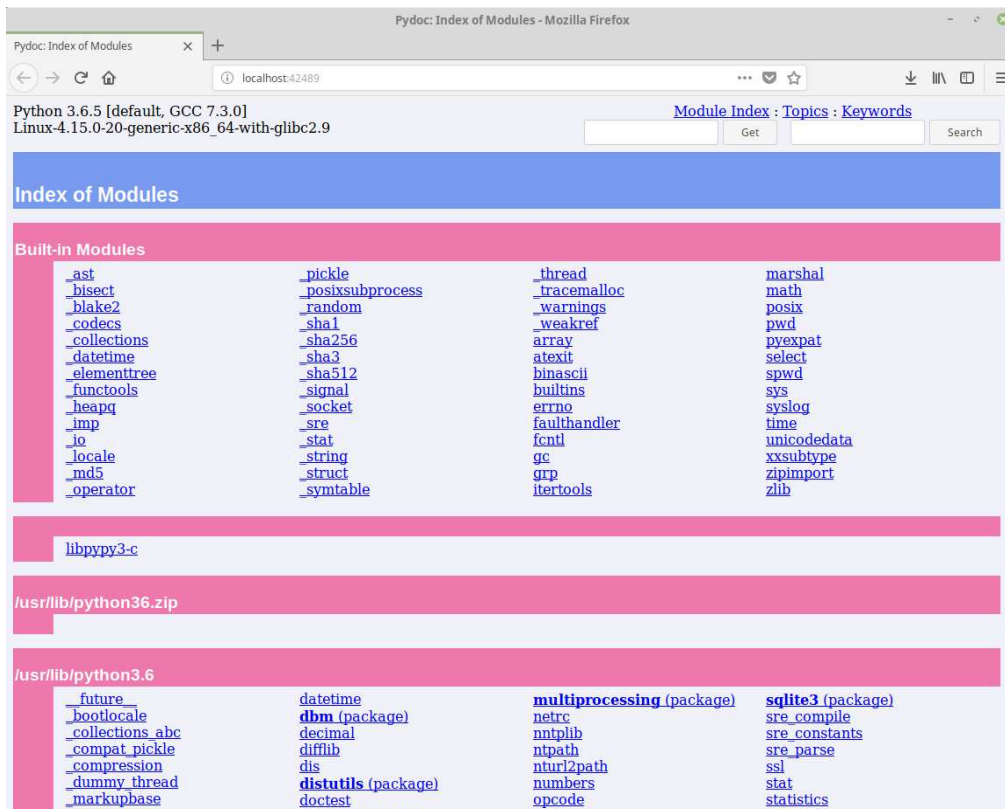
In [14]: for i in store22:
    print(i)

22
12-12-2017
45.32
3

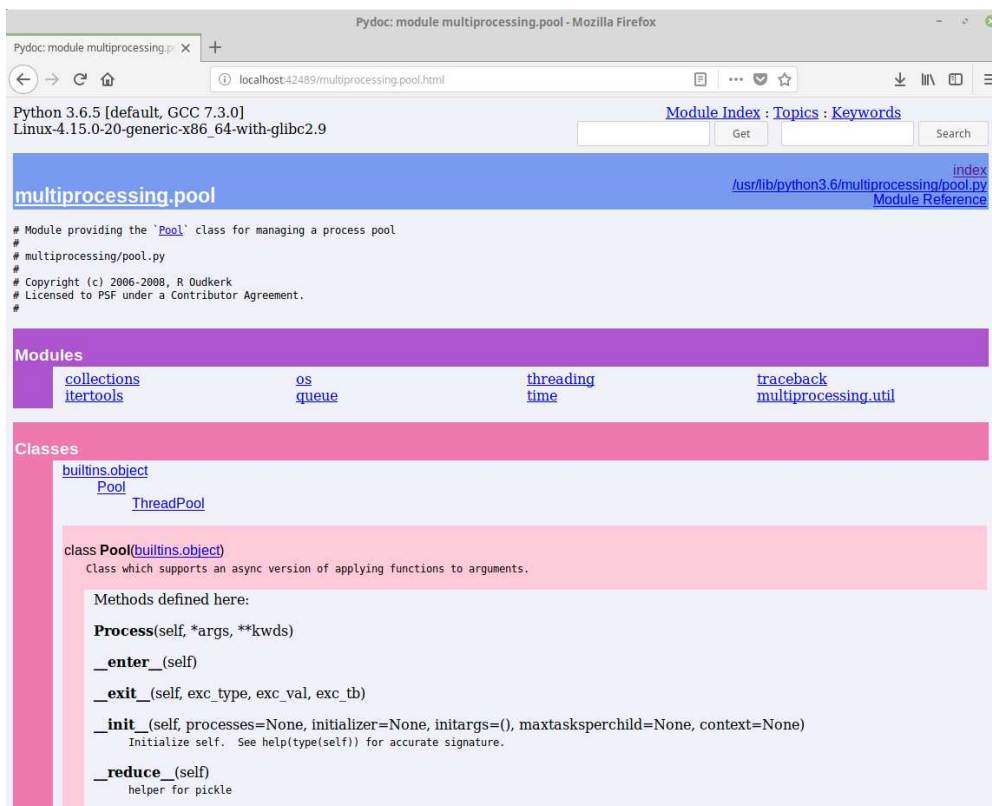
In [15]:
```

Rysunek 4.8. Kod zdefiniowany w pliku namedtuples\_sales.py

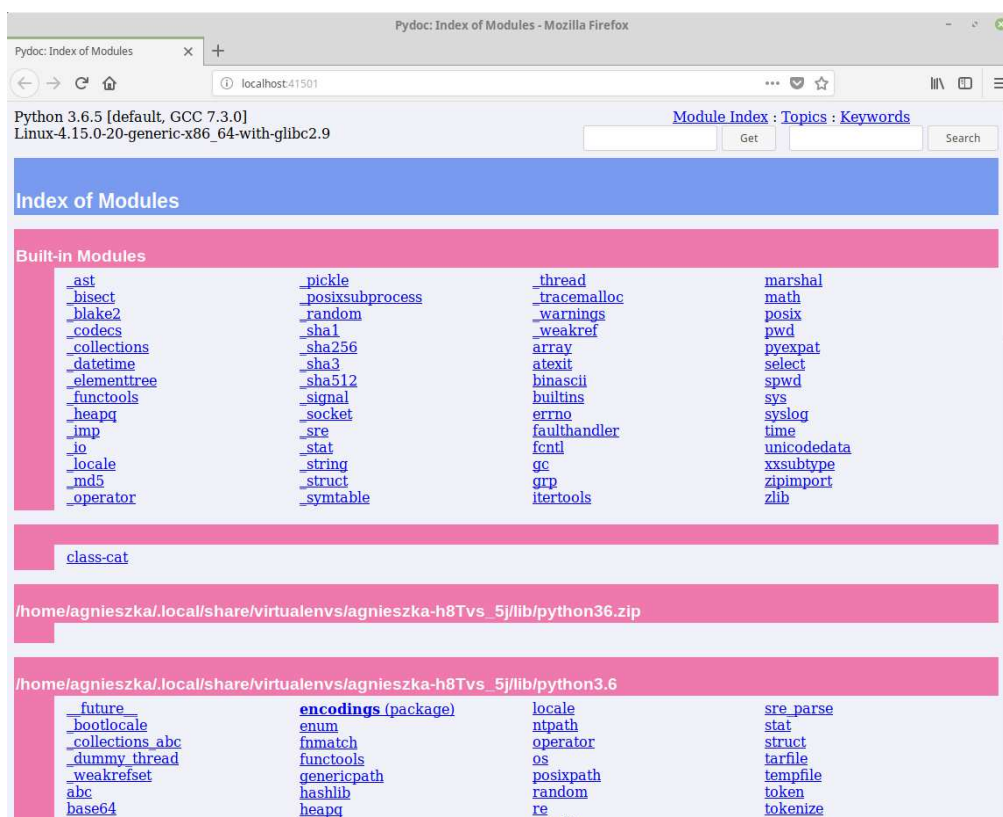




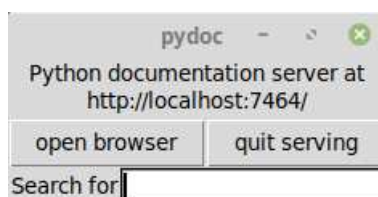
Rysunek 9.1. Oficjalna dokumentacja Pythona wyświetlona w przeglądarce WWW



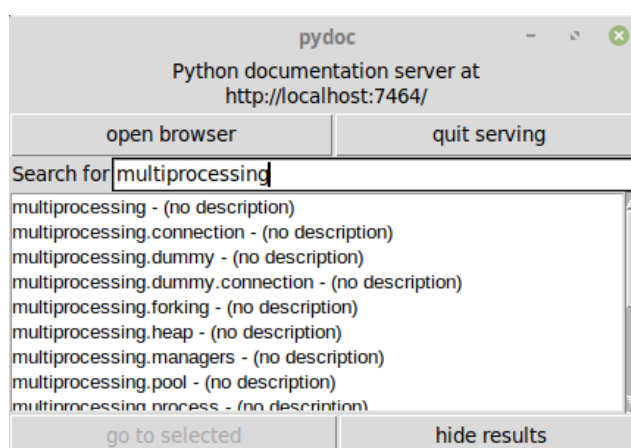
Rysunek 9.2. Dokumentacja multiprocessing.pool wyświetlona w przeglądarce WWW



Rysunek 9.3. Informacje o środowisku wirtualnym Pythona wyświetlone w przeglądarce WWW



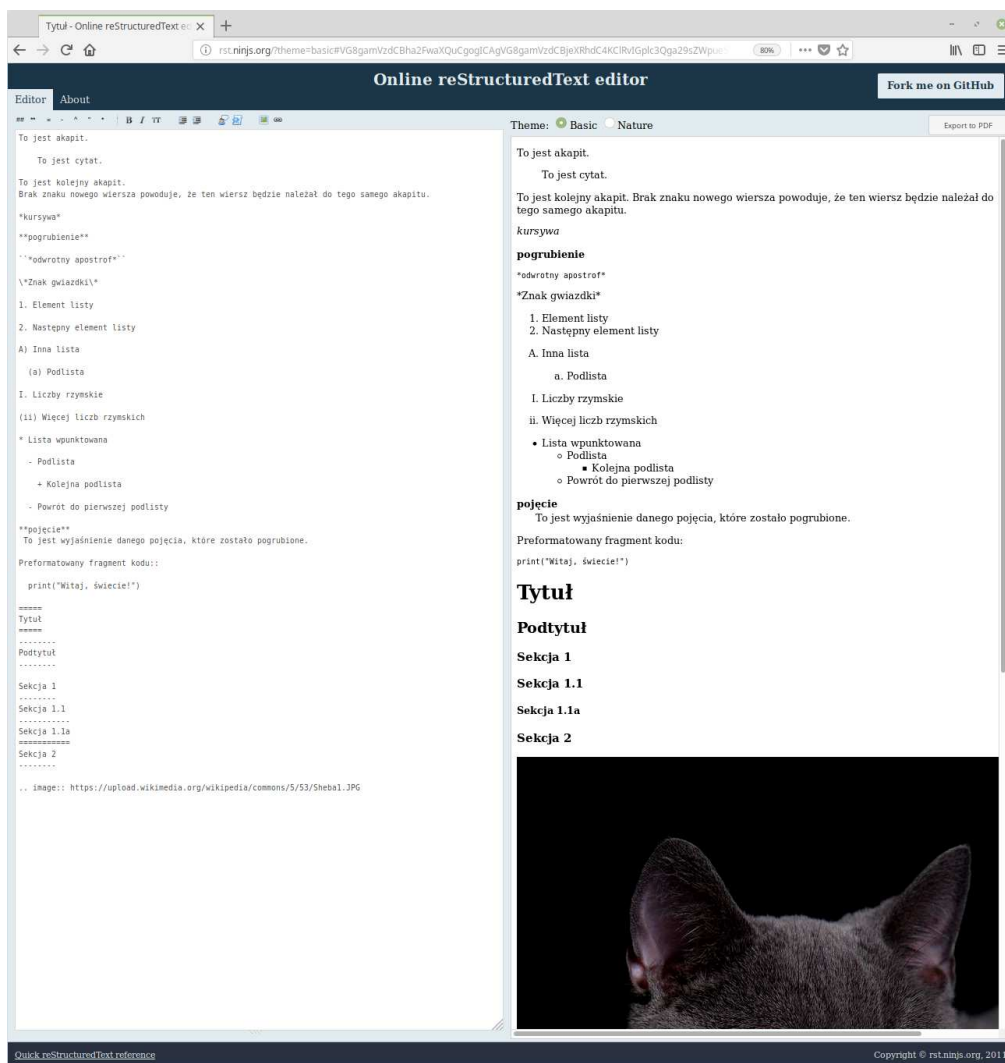
Rysunek 9.4. Proste okno dialogowe umożliwiające przeszukiwanie dokumentacji Pythona



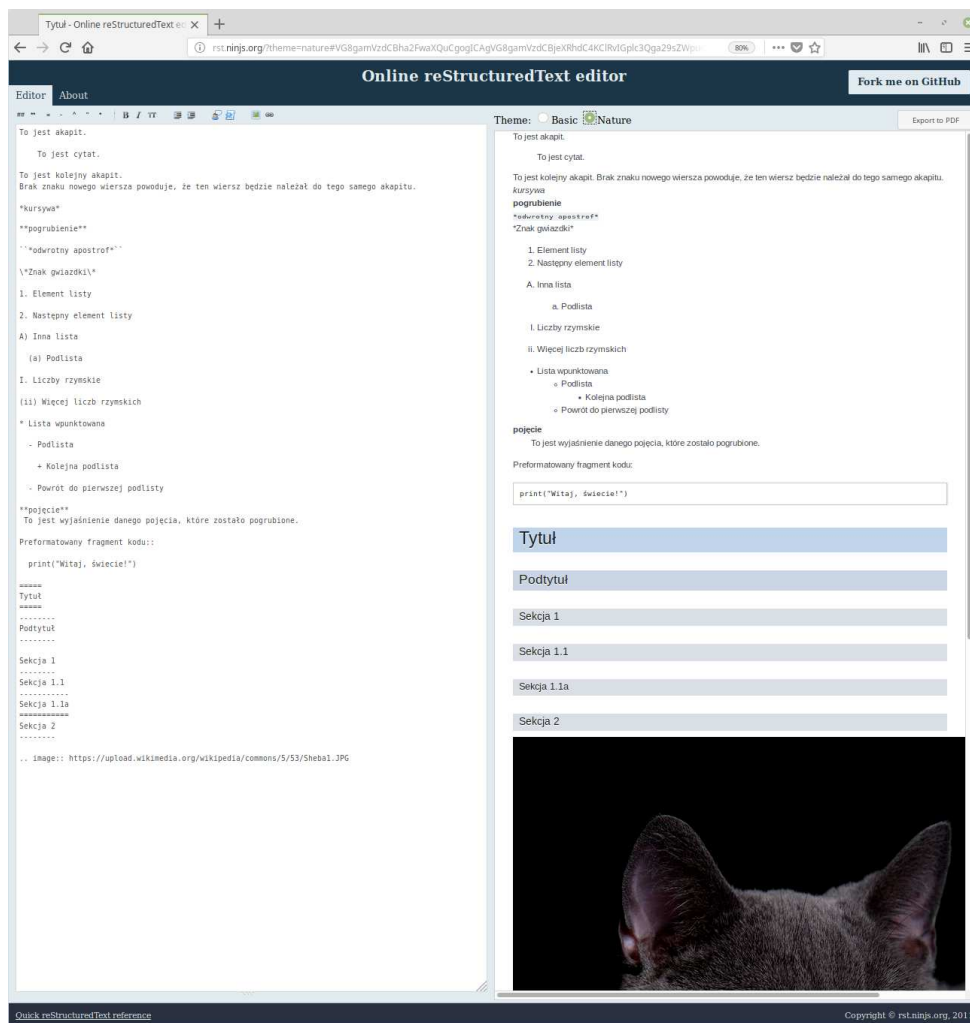
Rysunek 9.5. Symboliczne wręcz informacje wyświetlone przez okno dialogowe przeszukiwania dokumentacji Pythona



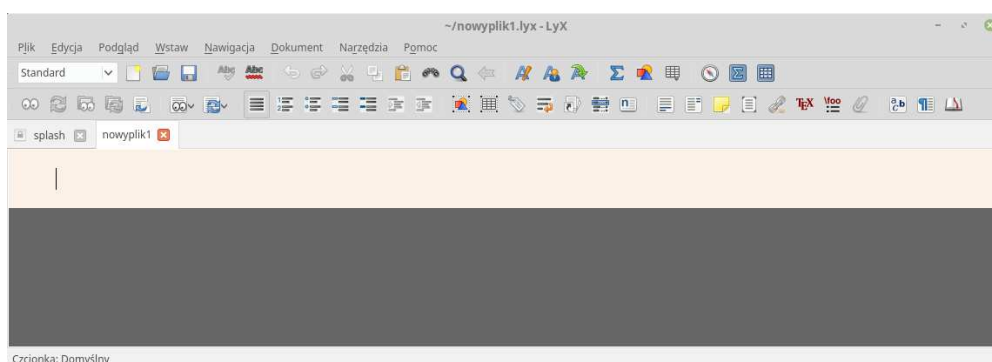
Rysunek 9.6. Dokumentacja multiprocessing.pool wyświetlona w oknie przeglądarki WWW



Rysunek 9.7. Przykład tekstu w formacie reST i dokumentu wygenerowanego na jego bazie

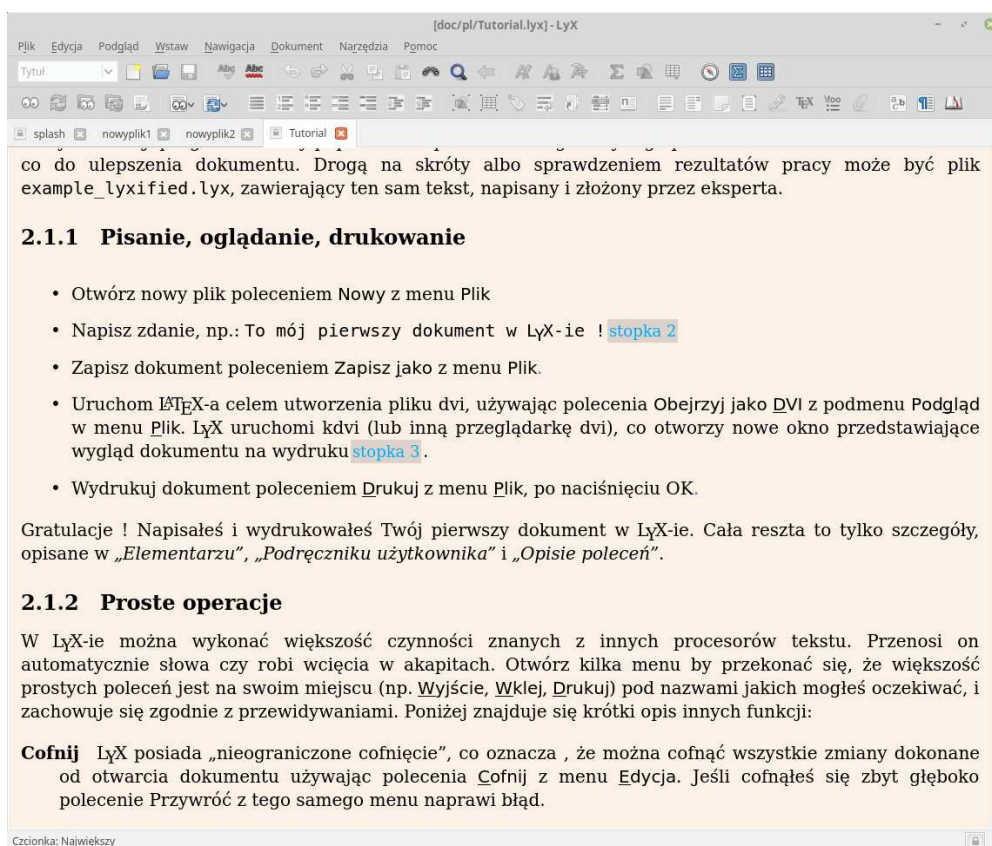


Rysunek 9.8. Dokument w innym formacie wygenerowany na podstawie tego samego kodu znaczników reST

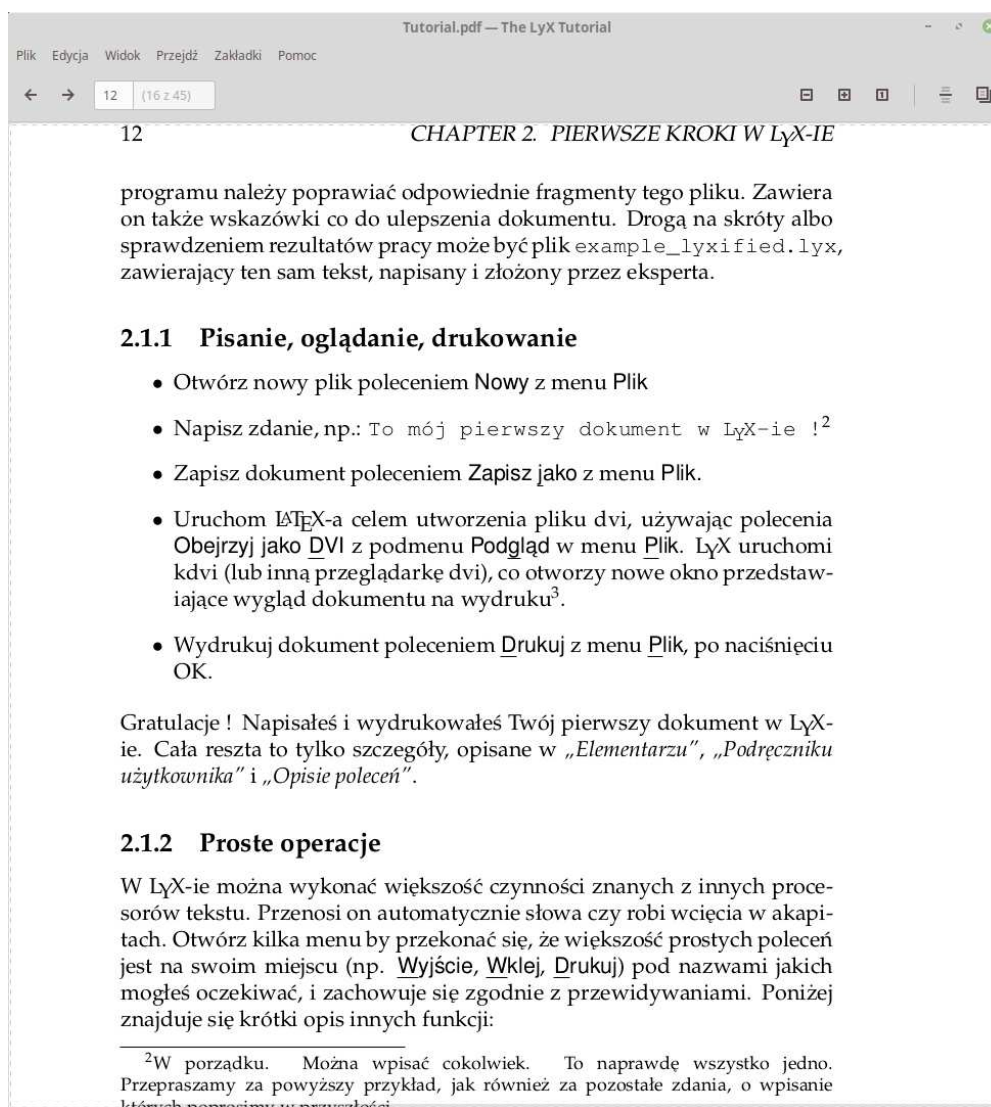


Rysunek 9.9. Okno programu LyX po pierwszym uruchomieniu tego programu

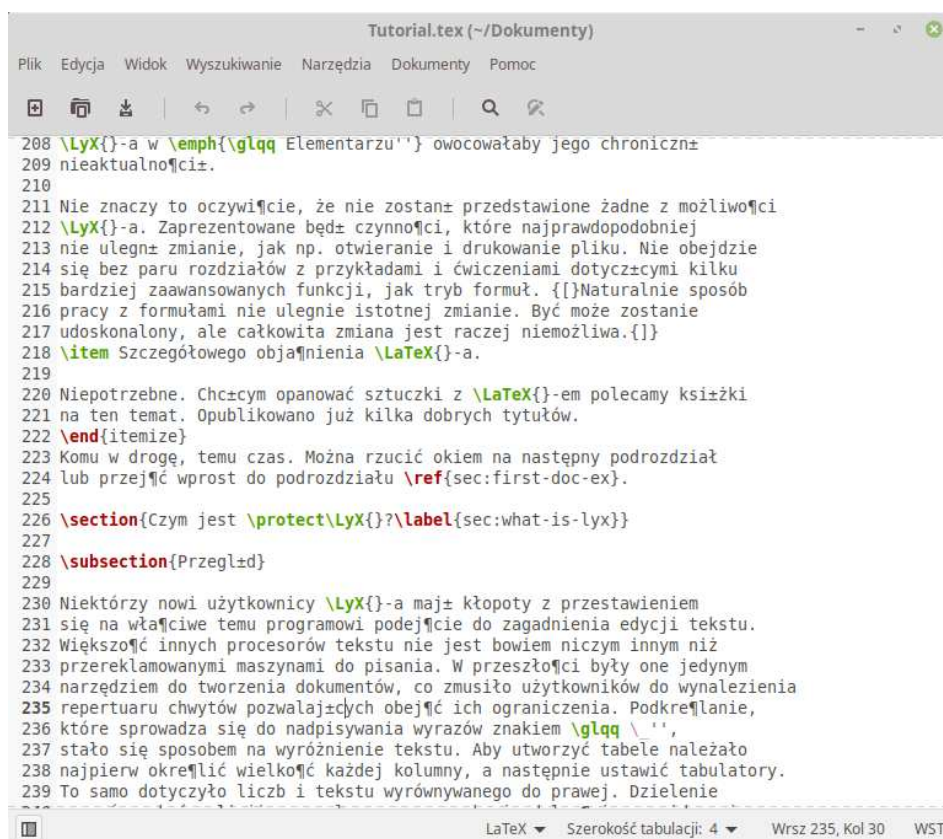




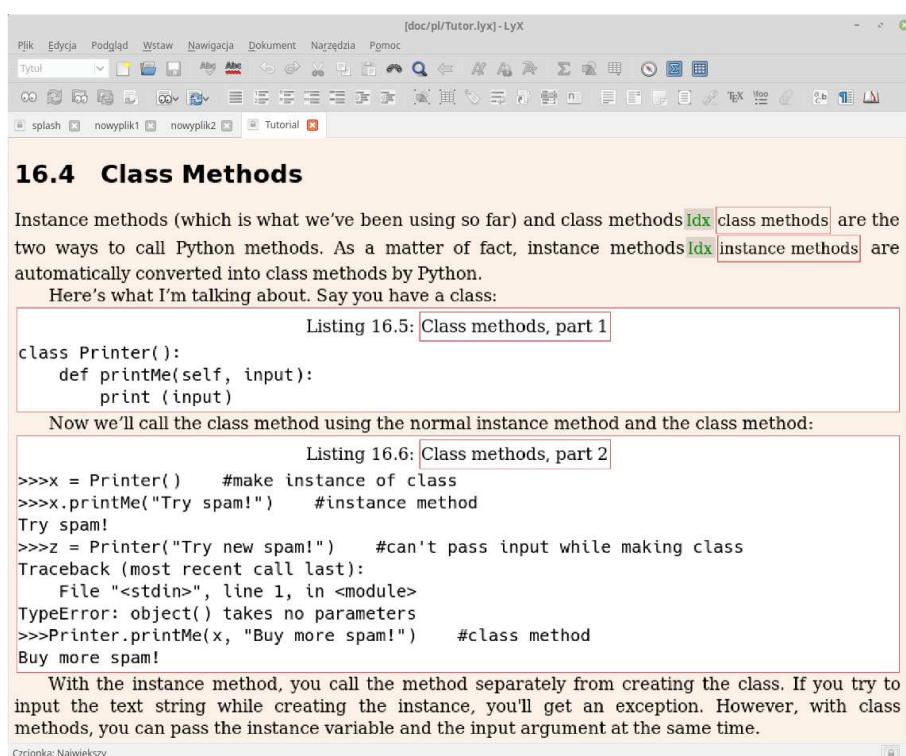
Rysunek 9.10. Fragment przykładowego dokumentu Samouczek dostarczanego wraz z programem LyX



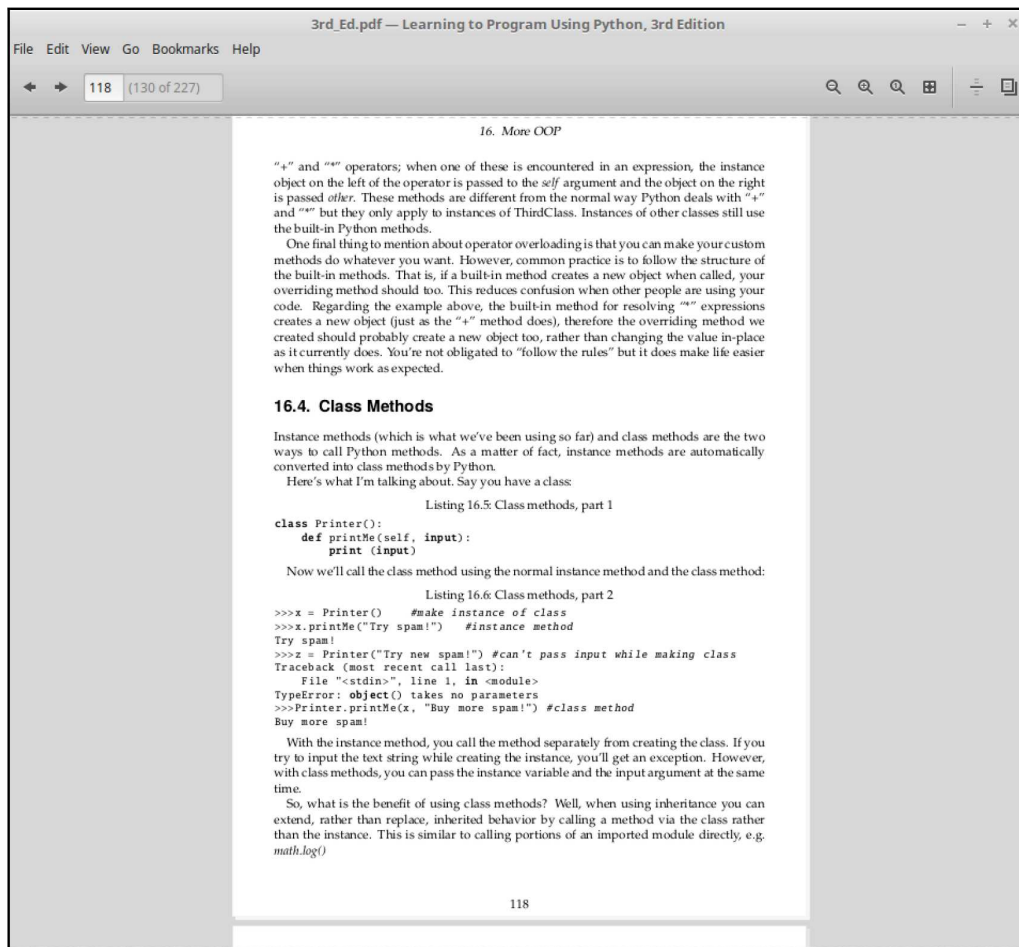
Rysunek 9.11. Fragment dokumentu Samouczek skonwertowany do formatu PDF



Rysunek 9.12. Kod znaczników LaTeX tego samego fragmentu dokumentu Samouczek



Rysunek 9.13. Fragment pierwszej mojej książki wyświetlony w programie LyX



Rysunek 9.14. Ten sam fragment mojej pierwszej książki skonwertowany do formatu PDF