

Лекция 01

Установка IDLE

Страница загрузки

<http://www.python.org/>





GO

Socialize

Sign In

[About](#)[Downloads](#)[Documentation](#)[Community](#)[Success Stories](#)[News](#)[Events](#)[Python](#) >>> [Downloads](#) >>> [Windows](#)

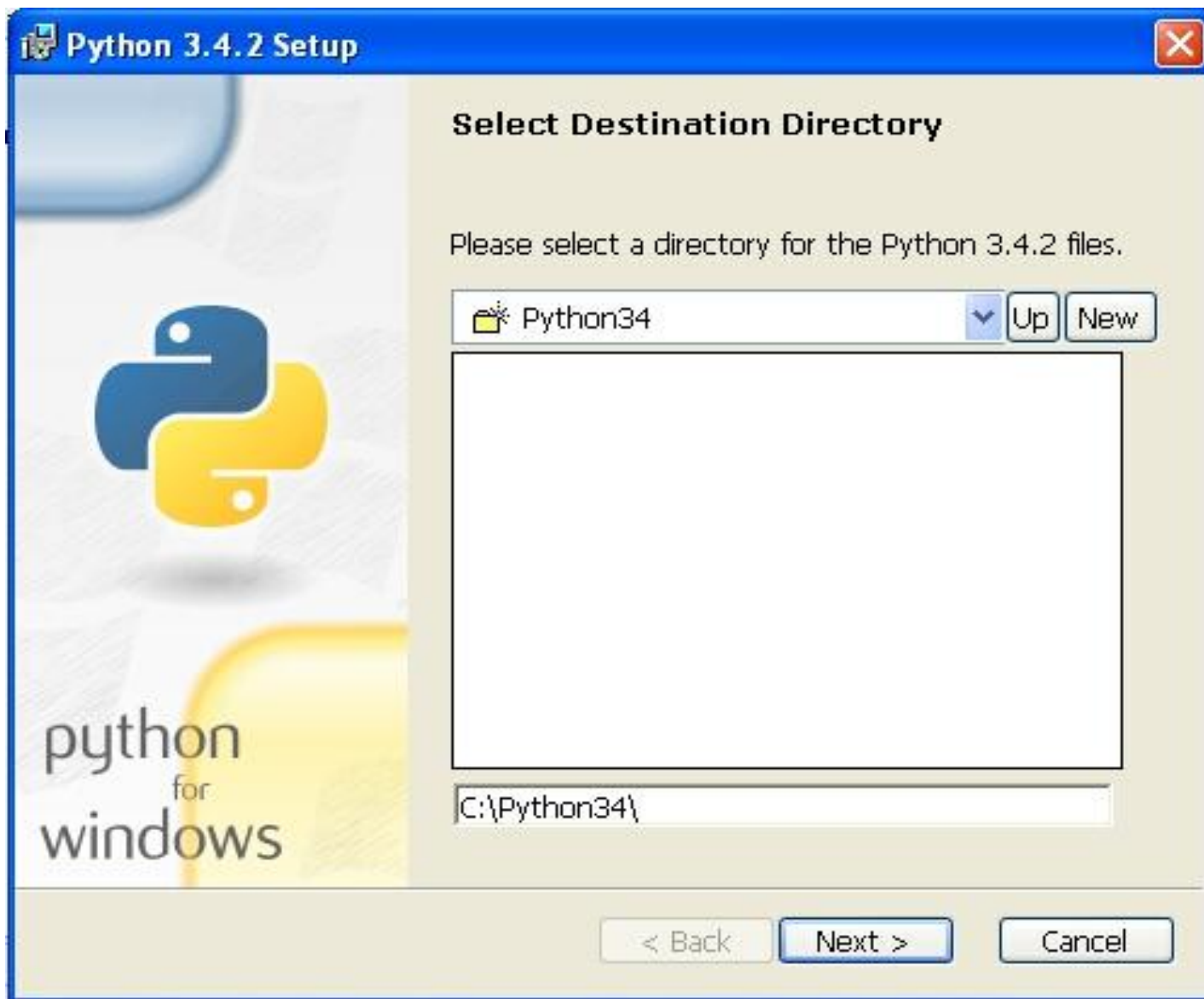
Python Releases for Windows

- [Latest Python 2 Release - Python 2.7.8](#)
- [Latest Python 3 Release - Python 3.4.2](#)
- [Python 3.4.2 - 2014-10-13](#)
 - Download [Windows x86 MSI installer](#)
 - Download [Windows x86-64 MSI installer](#)
 - Download [Windows help file](#)
 - Download [Windows debug information files for 64-bit binaries](#)
 - Download [Windows debug information files](#)
- [Python 3.4.2rc1 - 2014-09-22](#)
 - Download [Windows x86 MSI installer](#)
 - Download [Windows x86-64 MSI installer](#)
 - Download [Windows help file](#)

Старт инсталляции



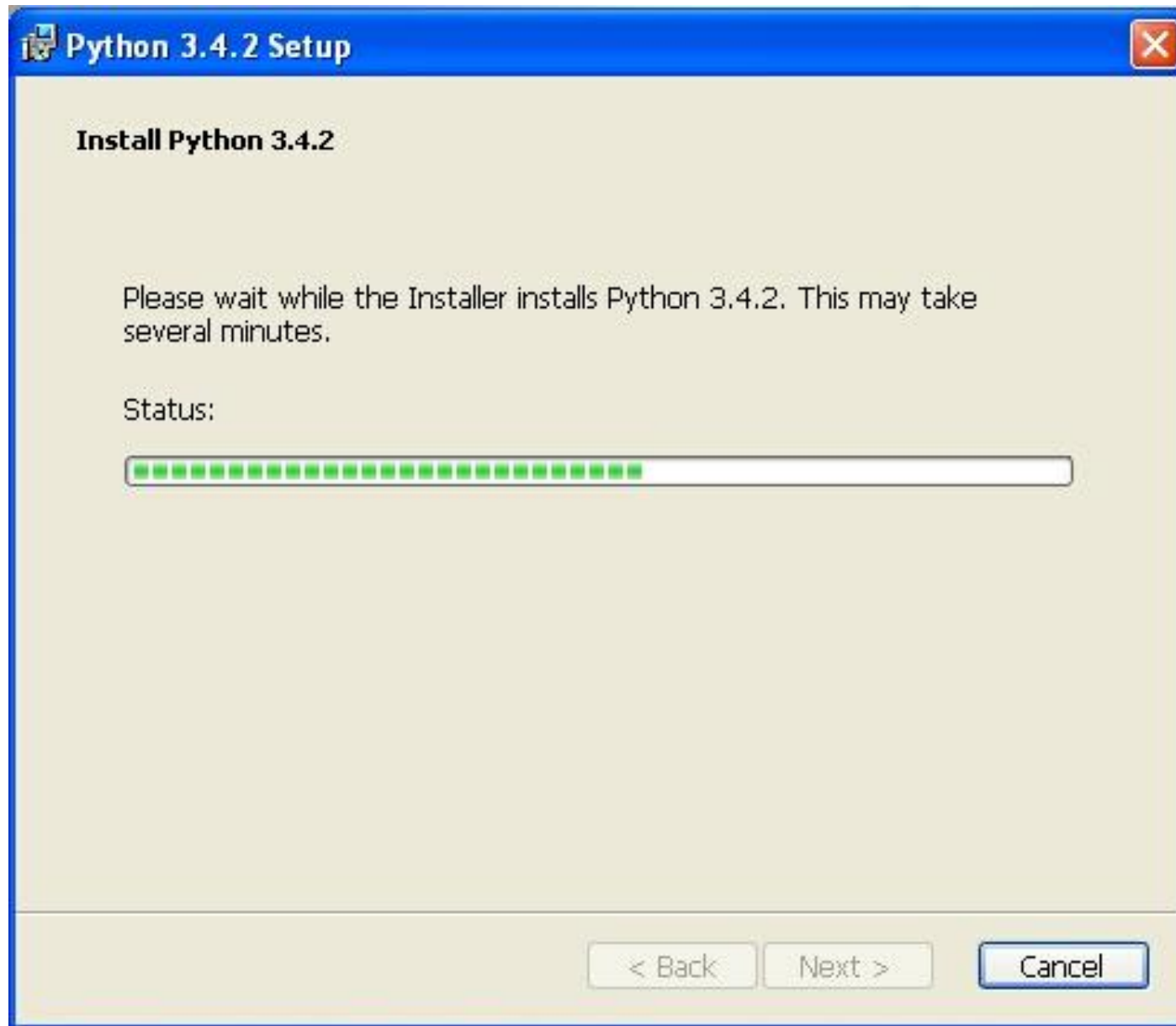
Указываем каталог инсталляции



Выбор устанавливаемых компонентов



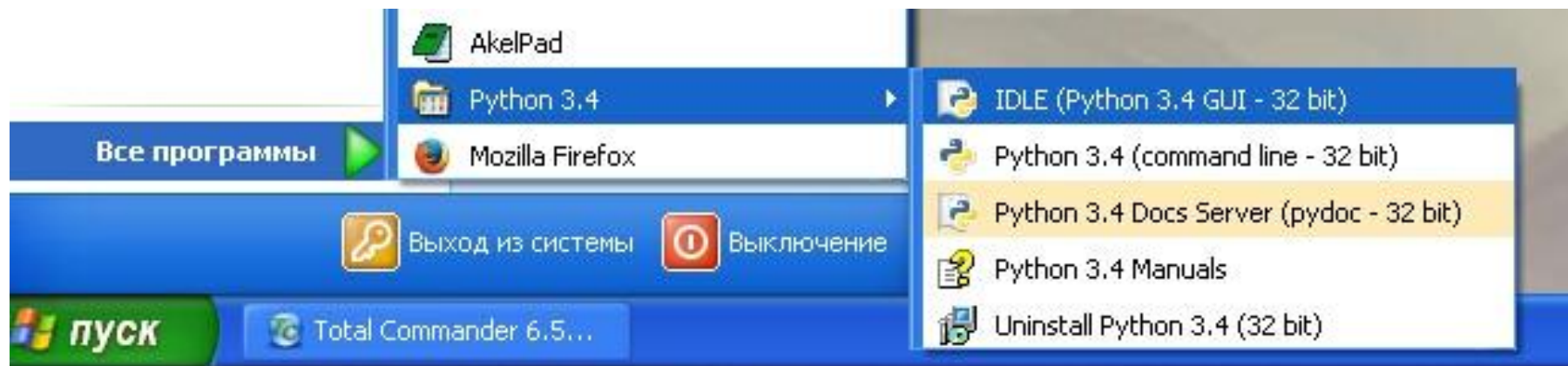
Продолжение инсталляции



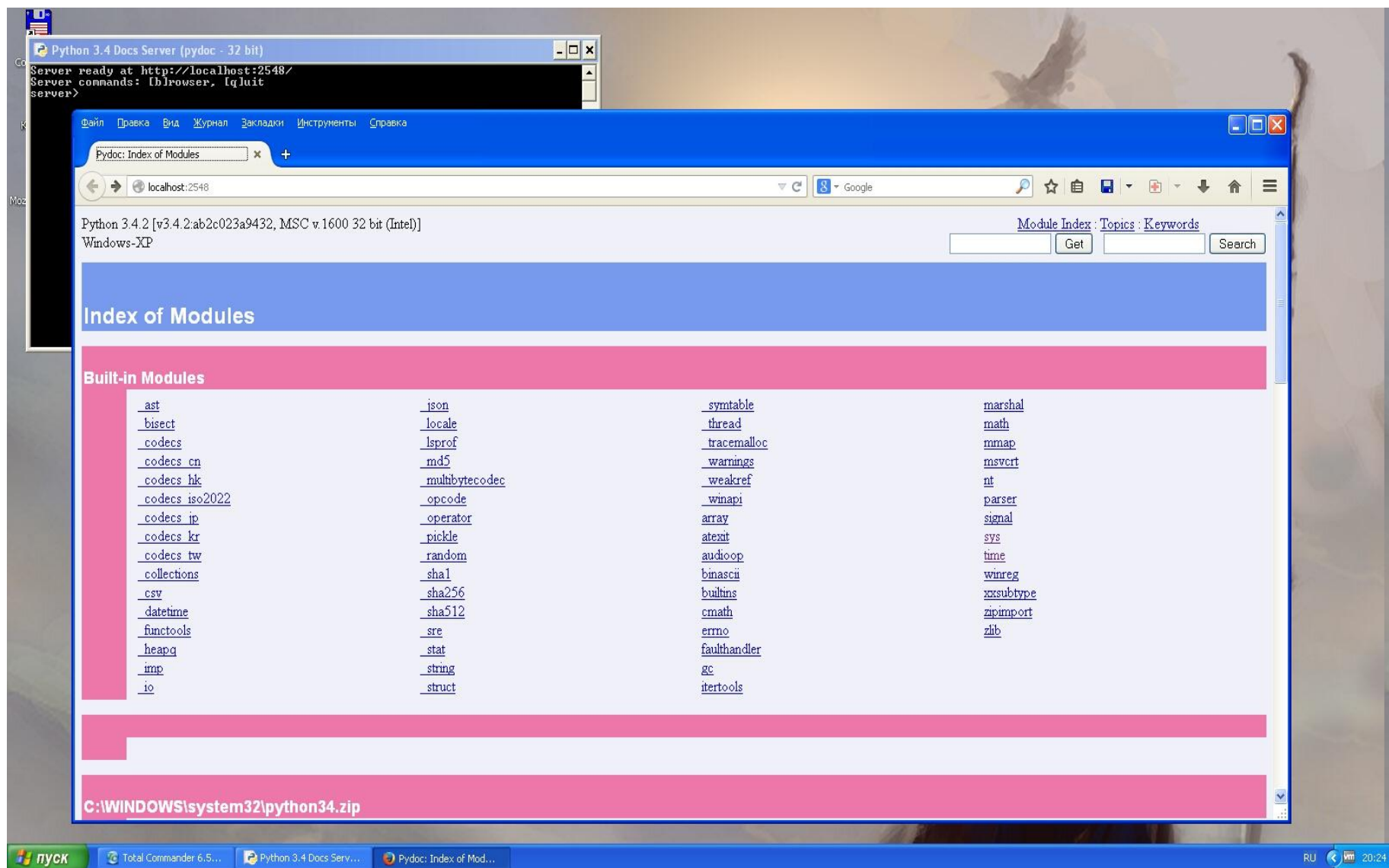
Завершение инсталляции



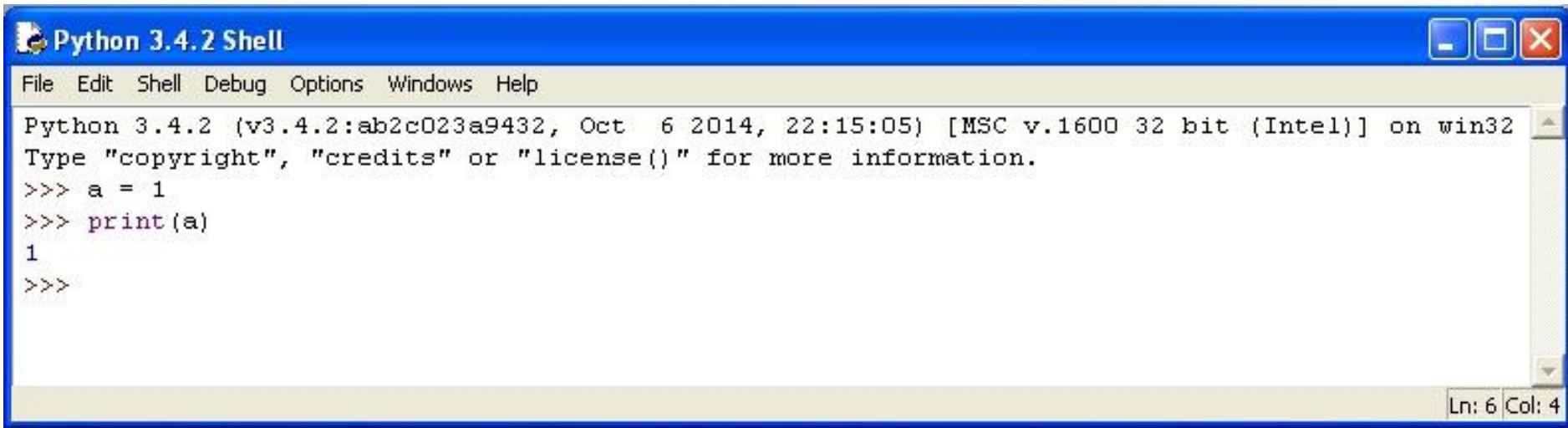
Программная группа Python



Интерактивная справка



Интерпретатор команд IDLE

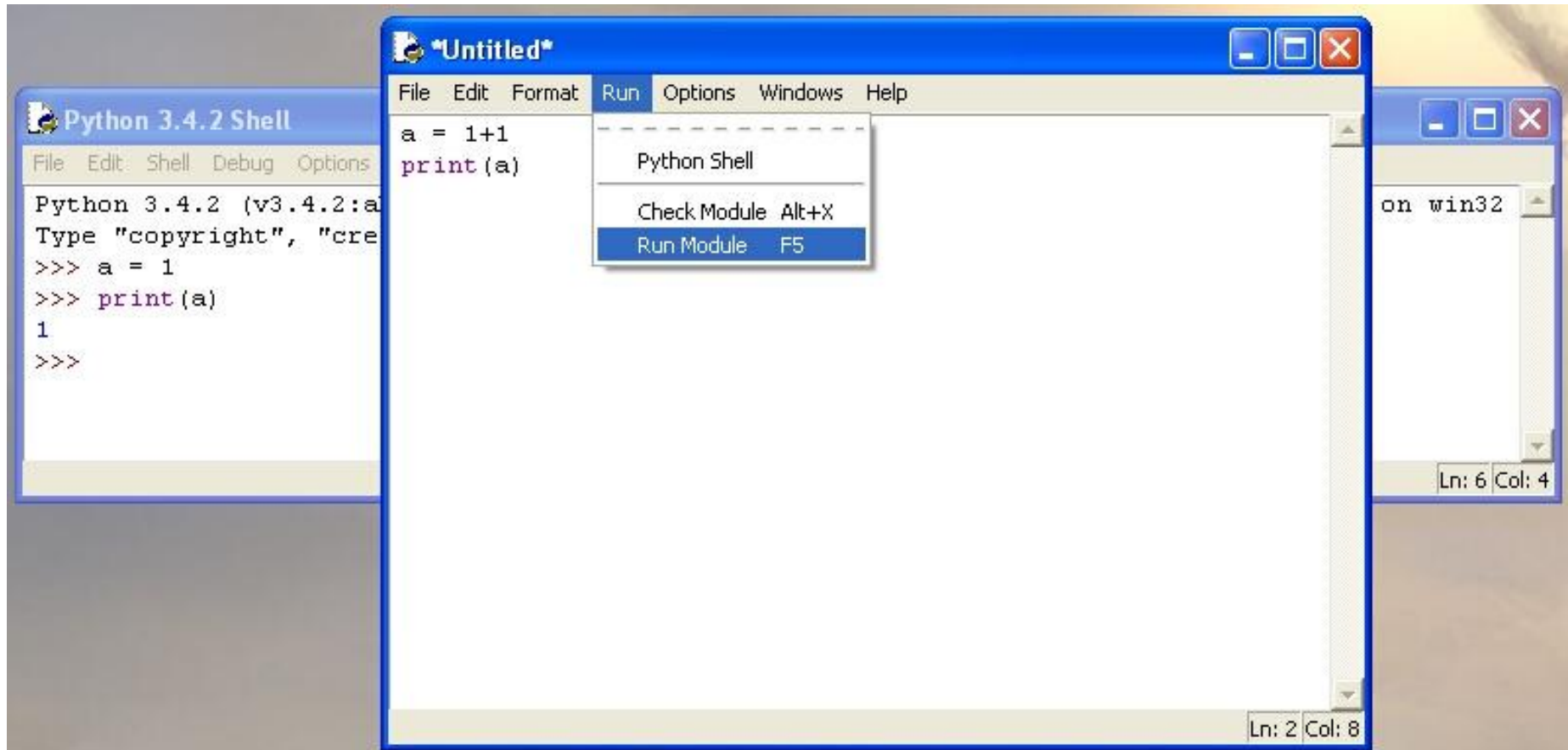


The image shows a screenshot of the Python 3.4.2 Shell window. The window has a blue title bar with the text "Python 3.4.2 Shell" and standard window control buttons (minimize, maximize, close). Below the title bar is a menu bar with the following items: File, Edit, Shell, Debug, Options, Windows, and Help. The main area of the window is a text editor with a light beige background. It contains the following text:

```
Python 3.4.2 (v3.4.2:ab2c023a9432, Oct 6 2014, 22:15:05) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> a = 1
>>> print(a)
1
>>>
```

At the bottom right of the window, there is a status bar that displays "Ln: 6 Col: 4".

Запуск скриптов внутри IDLE



Инсталляция в среде Linux на примере Ubuntu

Проверка установки Python:

```
python -V
```

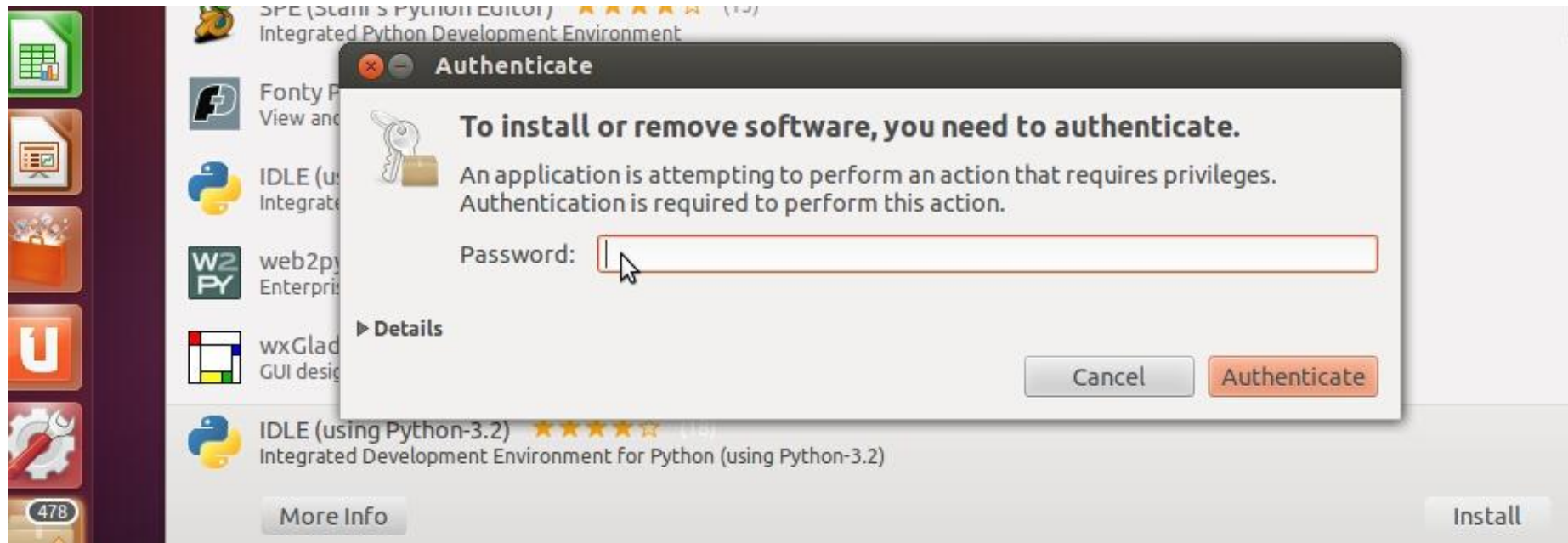
Центр инсталляции ПО Ubuntu



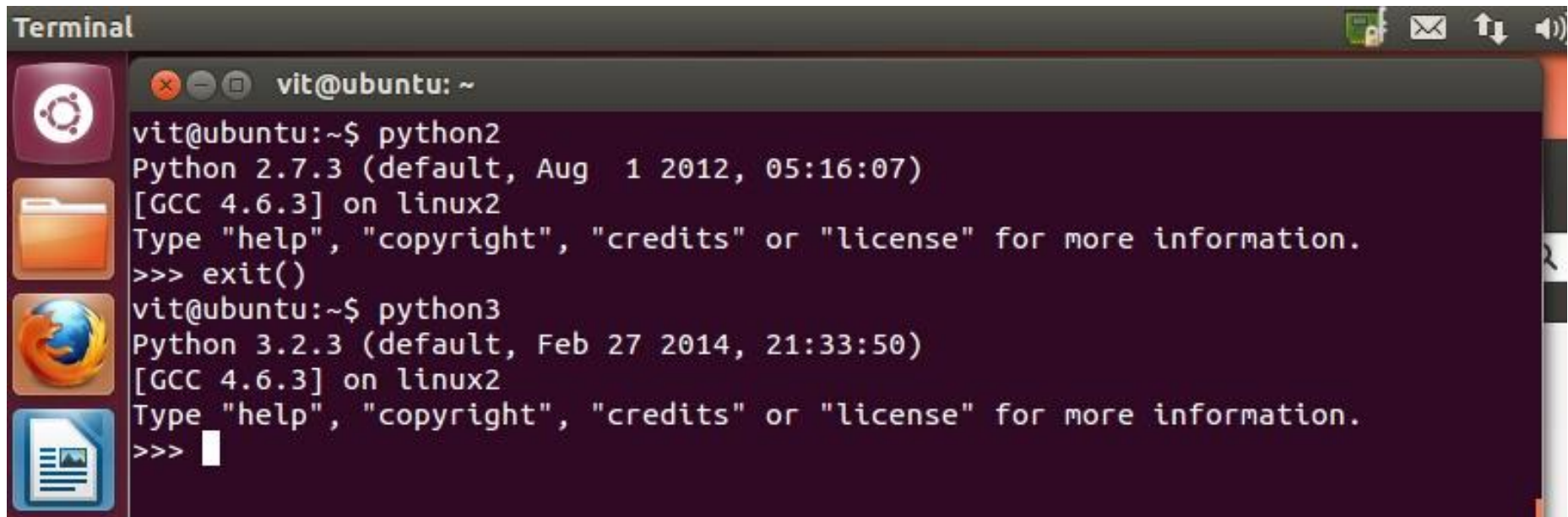
Поиск подходящего программного пакета



Подтверждение прав

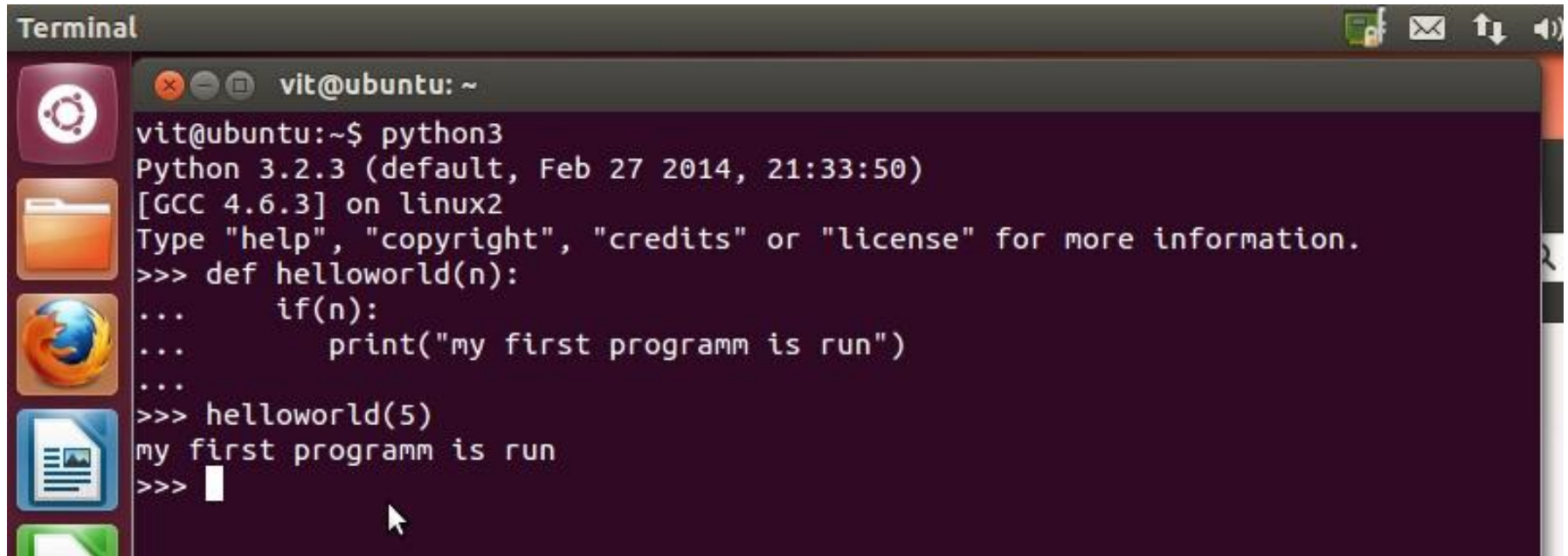


Консольный режим

A screenshot of a Linux terminal window titled "Terminal". The window has a dark grey title bar with standard window controls (close, minimize, maximize) and system icons (network, mail, volume) on the right. The terminal background is dark purple. The prompt is "vit@ubuntu: ~". The user enters "python2", which outputs "Python 2.7.3 (default, Aug 1 2012, 05:16:07)", "[GCC 4.6.3] on linux2", and instructions to type "help", "copyright", "credits", or "license" for more information. The user then enters ">>> exit()", which returns to the shell prompt. Next, the user enters "python3", which outputs "Python 3.2.3 (default, Feb 27 2014, 21:33:50)", "[GCC 4.6.3] on linux2", and the same instructions. The user enters ">>>" and a white cursor is visible. On the left side of the terminal window, there is a vertical dock with four icons: the Ubuntu logo, a folder, the Firefox logo, and a document icon.

```
Terminal
vit@ubuntu: ~
vit@ubuntu:~$ python2
Python 2.7.3 (default, Aug 1 2012, 05:16:07)
[GCC 4.6.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> exit()
vit@ubuntu:~$ python3
Python 3.2.3 (default, Feb 27 2014, 21:33:50)
[GCC 4.6.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> 
```

Ввод программы в консольном режиме

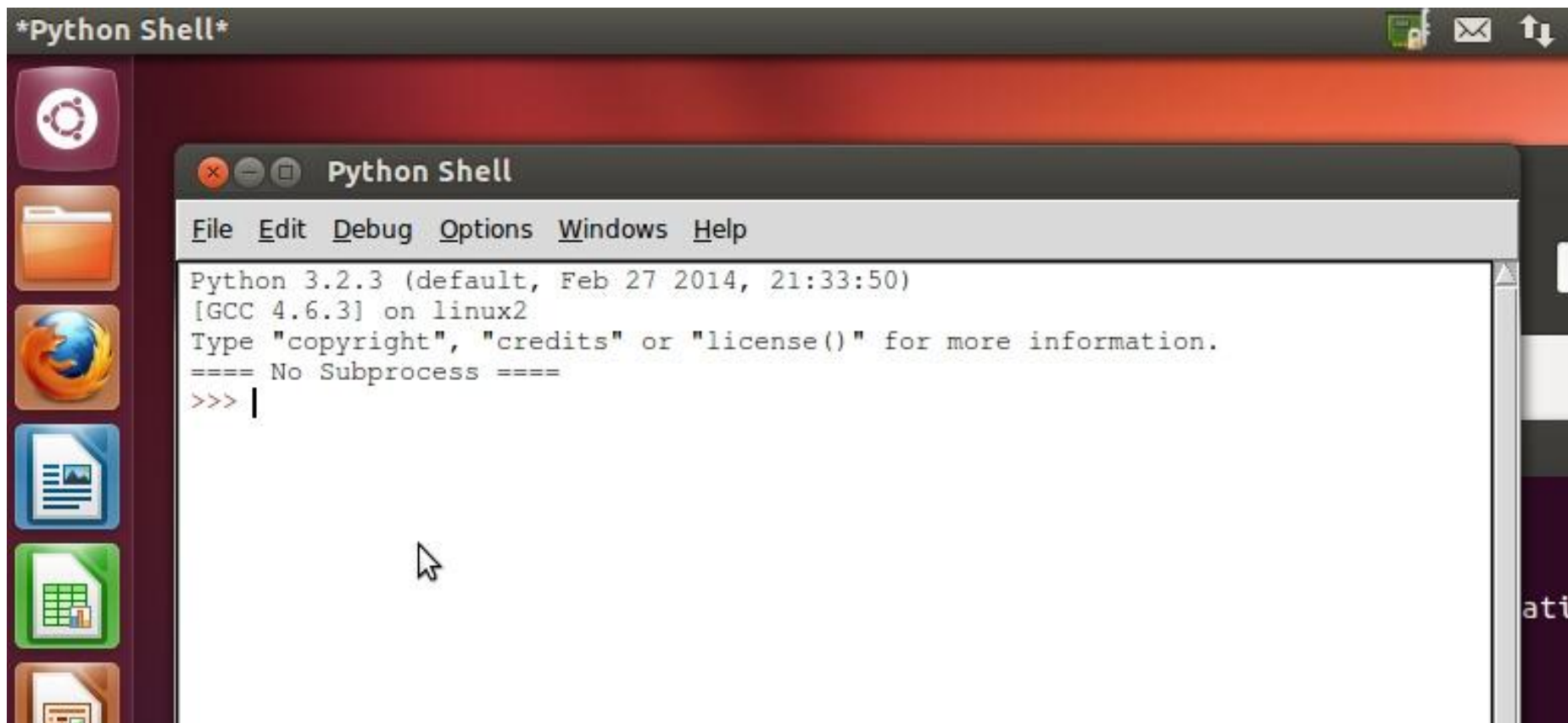
A screenshot of a Linux terminal window titled "Terminal". The window has a dark background and a light-colored title bar. On the left side, there is a vertical dock with several application icons: a red circle with a white gear, an orange folder, a Firefox browser icon, a document icon, and a green icon. The terminal content shows the user "vit@ubuntu" at the home directory "~". The user has entered the command "python3", which has installed Python 3.2.3. The output shows the version, the compiler used (GCC 4.6.3), and the operating system (linux2). The user is then prompted to type "help", "copyright", "credits", or "license" for more information. The user enters a series of commands to define a function "helloworld(n)" that prints "my first programm is run" if "n" is non-zero. The user then calls the function "helloworld(5)", and the output "my first programm is run" is displayed. The prompt ">>>" is shown at the end of the line, indicating the user is still in the Python interactive shell.

```
Terminal
vit@ubuntu: ~
vit@ubuntu:~$ python3
Python 3.2.3 (default, Feb 27 2014, 21:33:50)
[GCC 4.6.3] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> def helloworld(n):
...     if(n):
...         print("my first programm is run")
...
>>> helloworld(5)
my first programm is run
>>>
```

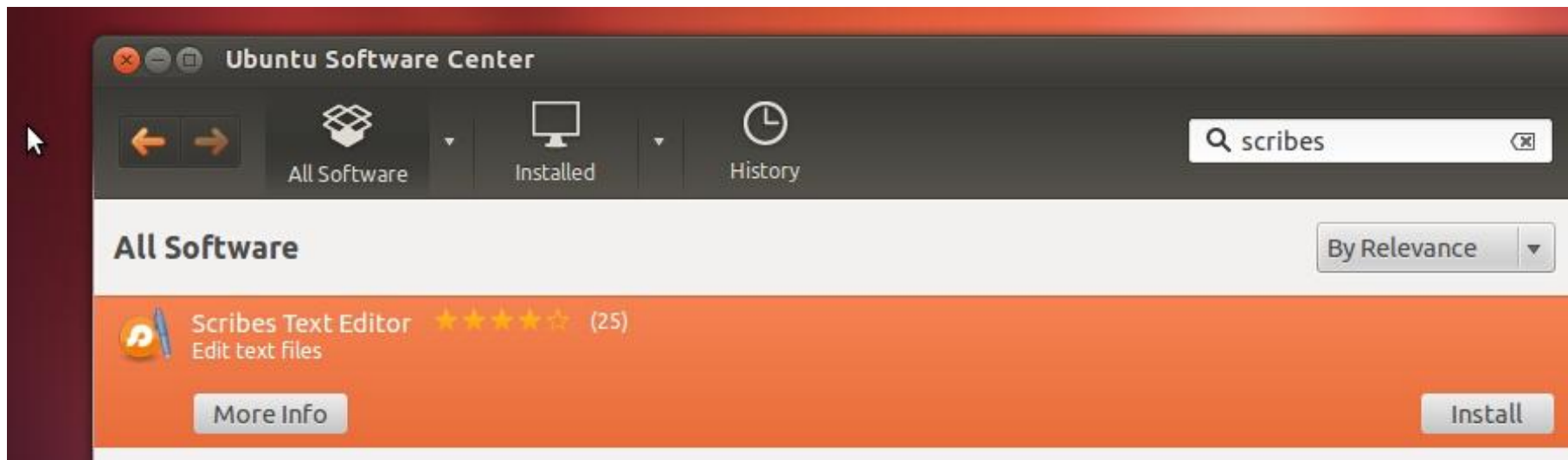
IDLE в среде Linux



Okho IDLE



Установка Scribes



Общая структура программы

- `# -*- coding: cp1251 -*-`
- `#!/usr/bin/python`
- `#!/usr/local/bin/python`
- `#!/usr/bin/env python`

Отсутствие операторных скобок, код C++

```
if ( a >= b)
{
    cout << a << " >= " << b;
} else
{
    cout << a << " <= " << b;
}
```

Отсутствие операторных скобок, код Python

```
if a >= b :
```

```
    print (a + ">=" + b)
```

```
else:
```

```
    print (a + "<=" + b)
```

Такой код выдаст ошибку:

```
if a >= b :  
    print (str(a) + ">=" + str(b))  
else:  
    print (str(a) + "<=" + str(b))
```



Обозначение «;»

$a = 1; b = 2; c = 3$

Переносы строк

Вариант 1:

$$a = 1 + 2 \backslash \\ + 3$$

Вариант 2:

$$a = (1 + 2 \\ + 3)$$

Комментарии

это офигенно длинный комментарий

""""

Это очень

Длинный и многословный комментарий,
располагающийся

На нескольких строках """"

Особенности работы в IDLE

Однострочный режим работы можно использовать в качестве калькулятора:

```
>>> 1 + 2
```

```
3
```

```
>>> _ + 10
```

```
13
```


Горячие клавиши

- <Ctrl> + <Пробел> - выбор из списка
- <Alt> + </> - завершение ввода
- <Alt> + <N> - предыдущая команда
- <Alt> + <P> - последняя команда

Ввод и вывод результатов работы

Сигнатура оператора print:

```
print([Объекты], [sep= ' '] [, end='\n'] [, file=sys.stdout])
```

Примеры:

```
print ("str1", "str2")
```

```
print ("str1", "str2", sep="")
```

```
>>> print ("str1","str2",end="@"); print(); print("str3", end='@')
```

```
str1 str2@
```

```
str3@
```

```
>>>
```

Ввод данных с консоли, сигнатура функции input

[<переменная> =] input([<Какое-то сообщение>])

Пример:

```
>>> usermessage = input("Я вас внимательно слушаю! \n"); \
      print("Вы сказали:",usermessage)
```

Я вас очень внимательно слушаю!

Привет!

Вы сказали: Привет!

```
>>>
```

```
input("Для выхода из программы нажмите \
      любую клавишу")
```

Массив argv

```
#-*- coding: utf-8 -*-
```

```
import sys
```

```
arr = sys.argv
```

```
print(arr[0])
```

```
print(arr[1])
```

```
print(arr[2])
```

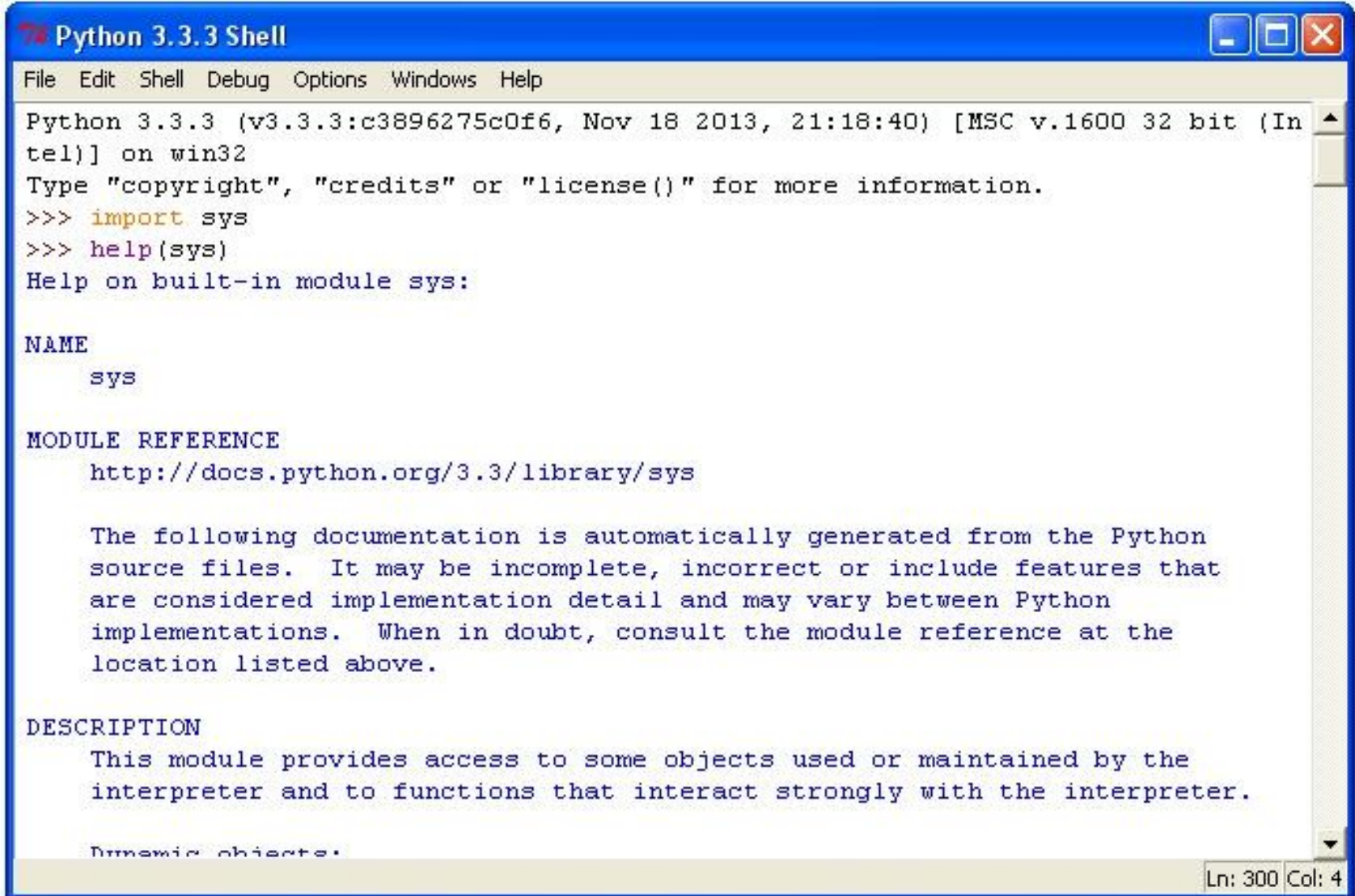
```
print(arr[3])
```

Массив argv

```
C:\Python33\work>test.py -fghdg -h -w
C:\Python33\work\test.py
-fghdg
-h
-w

C:\Python33\work>
```

Документация Python



```
Python 3.3.3 Shell
File Edit Shell Debug Options Windows Help
Python 3.3.3 (v3.3.3:c3896275c0f6, Nov 18 2013, 21:18:40) [MSC v.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> import sys
>>> help(sys)
Help on built-in module sys:

NAME
    sys

MODULE REFERENCE
    http://docs.python.org/3.3/library/sys

    The following documentation is automatically generated from the Python
    source files. It may be incomplete, incorrect or include features that
    are considered implementation detail and may vary between Python
    implementations. When in doubt, consult the module reference at the
    location listed above.

DESCRIPTION
    This module provides access to some objects used or maintained by the
    interpreter and to functions that interact strongly with the interpreter.

    Dynamic objects:
```

Ln: 300 Col: 4

Что можно делать с помощью Python?

Всё, что угодно, - от веб-сайтов и игровых программ до управления роботами и космическими кораблями!

- Развитые сетевые средства
- Доступ ко многим базам данных
- Развитые системные средства
- Отличные графические средства
- «Батарейки» в комплекте!

Первая программа

```
# -*- coding: utf-8 -*-
```

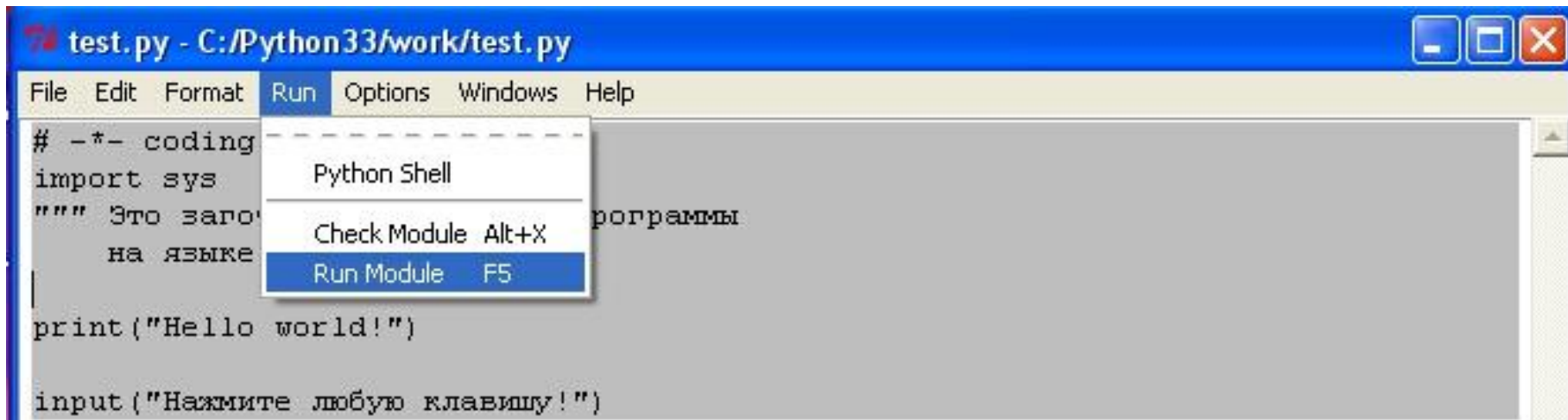
```
import sys
```

```
""" Это заготовка для первой программы  
на языке Python """
```

```
print("Hello world!")
```

```
input("Нажмите любую клавишу!")
```

Запуск программ из IDLE



>>>

Hello world!

Нажмите любую клавишу!п

>>>

Прочие IDE

- Anaconda (www.continuum.io)
- Ninja (www.ninja-ide.org)
- Eclipse (www.eclipse.org)
- Geany (www.geany.org)
- PyCharm (www.jetbrains.com)
- Еще несколько штук:
(<https://python-scripts.com/ide-for-python>)