

# Python Starter

Введение в Python

# Python Starter

После урока обязательно



Повторите этот урок в видео формате на [ITVDN.com](http://itvdn.com)

Доступ можно получить через руководство вашего учебного центра



Проверьте как Вы усвоили данный материал на [TestProvider.com](http://testprovider.com)

# Python Starter

Введение в Python

Общие сведения

# Python Starter

## Язык программирования Python

**Python** – высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода.



# Python Starter

## Особенности Python

- минималистичность синтаксиса;
- простота изучения;
- широкие возможности;
- поддержка разных парадигм программирования;
- интерпретируемый;
- динамическая типизация;
- открытость;
- кроссплатформенность;
- существование альтернативных реализаций, которые решают определённые проблемы (Stackless Python, PyPy) или интегрируют его в определённую платформу (IronPython для .NET, Jython для Java);
- активно развивается;



# Python Starter

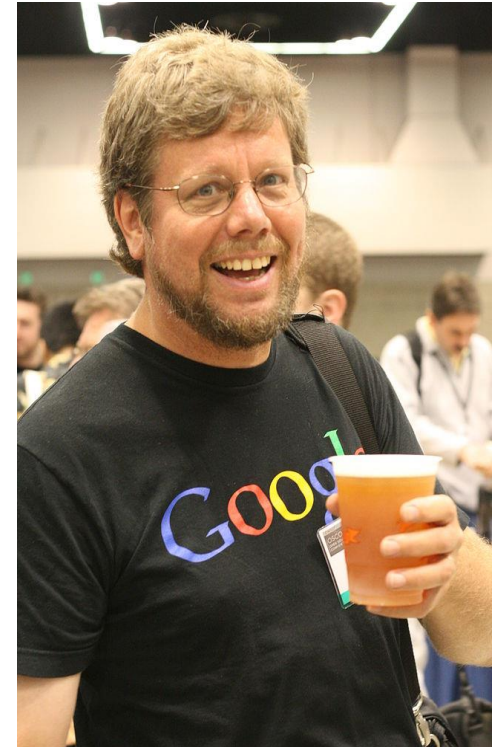
Язык программирования Python

История языка

# Python Starter

## История языка

- Разрабатывался с конца 80-х, выпущен в феврале 1991 г.
- Создатель – Гвидо ван Россум (тогда сотрудник голландского института CWI, сейчас – разработчик в Dropbox).
- Назван в честь британского телешоу «Летающий цирк Монти Пайтона».
- Создавался под влиянием других языков, вобрал в себя множество их возможностей.
- На данный момент существуют две актуальные ветки языка – Python 2 и Python 3. Новые проекты следует создавать на Python 3, если для них не планируется использовать какие-либо специфические библиотеки, которые работают только с Python 2 или нет других явных причин использовать Python 2.



Гвидо ван Россум

# Python Starter

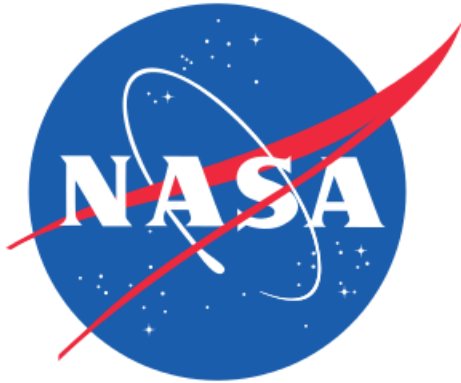
Язык программирования Python

Применение Python



# Python Starter

## Области применения



- скрипты, утилиты;
- научная сфера;
- исследование данных;
- веб-приложения;
- сервисы;
- разработка игр;



**mozilla**  
FOUNDATION



# Python Starter

## Сферы, в которых применить Python нельзя

- создание драйверов устройств;
- низкоуровневое программирование;

```
typedef struct _DRIVER_OBJECT
{
    SHORT Type;
    SHORT Size;
    PDEVICE_OBJECT DeviceObject;
    ULONG Flags;
    PVOID DriverStart;
    ULONG DriverSize;
    PVOID DriverSection;
    PDRIVER_EXTENSION DriverExtension;
    UNICODE_STRING DriverName;
    PUNICODE_STRING HardwareDatabase;
    PFAST_IO_DISPATCH FastIoDispatch;
    LONG * DriverInit;
    PVOID DriverStartIo;
    PVOID DriverUnload;
    LONG * MajorFunction[28];
} DRIVER_OBJECT, *PDRIVER_OBJECT;
```

# Python Starter

Язык программирования Python

Преимущества и недостатки

# Python Starter

## Преимущества

- простота;
- читабельность;
- мощная стандартная библиотека и огромный выбор сторонних библиотек и модулей;
- краткость кода и экономия времени разработчика;
- ВОЗМОЖНОСТЬ СВЯЗЫВАНИЯ С КОДОМ на других языках;

## Zen of Python

```
>>> import this
```

The Zen of Python, by Tim Peters

Beautiful is better than ugly.  
Explicit is better than implicit.  
Simple is better than complex.  
Complex is better than complicated.  
Flat is better than nested.  
Sparse is better than dense.  
Readability counts.  
Special cases aren't special enough to break the rules.  
Although practicality beats purity.  
Errors should never pass silently.  
Unless explicitly silenced.  
In the face of ambiguity, refuse the temptation to guess.  
There should be one— and preferably only one --obvious way to do it.  
Although that way may not be obvious at first unless you're Dutch.  
Now is better than never.  
Although never is often better than \*right\* now.  
If the implementation is hard to explain, it's a bad idea.  
If the implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea — let's do more of those!

# Python Starter

## Недостатки

- низкая по сравнению с компилируемыми языками скорость работы;
- GIL (Global Interpreter Lock);



# Python Starter

Введение в Python

Интерпретатор и  
интегрированная среда разработки

# Python Starter

## Интерпретатор Python

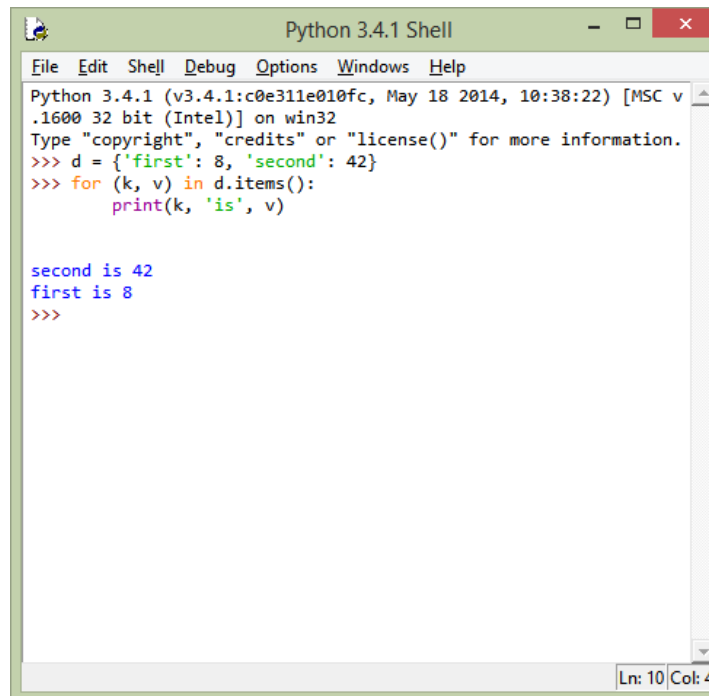
Интерпретатор — программа (разновидность транслятора), выполняющая интерпретацию.

Официальный сайт Python: <https://python.org/>

Загрузить интерпретатор: <https://www.python.org/downloads/>

Два режима работы:

- выполнение программ;
- интерактивный режим.



The screenshot shows a window titled "Python 3.4.1 Shell". The window has a menu bar with "File", "Edit", "Shell", "Debug", "Options", "Windows", and "Help". The main text area displays the following text:

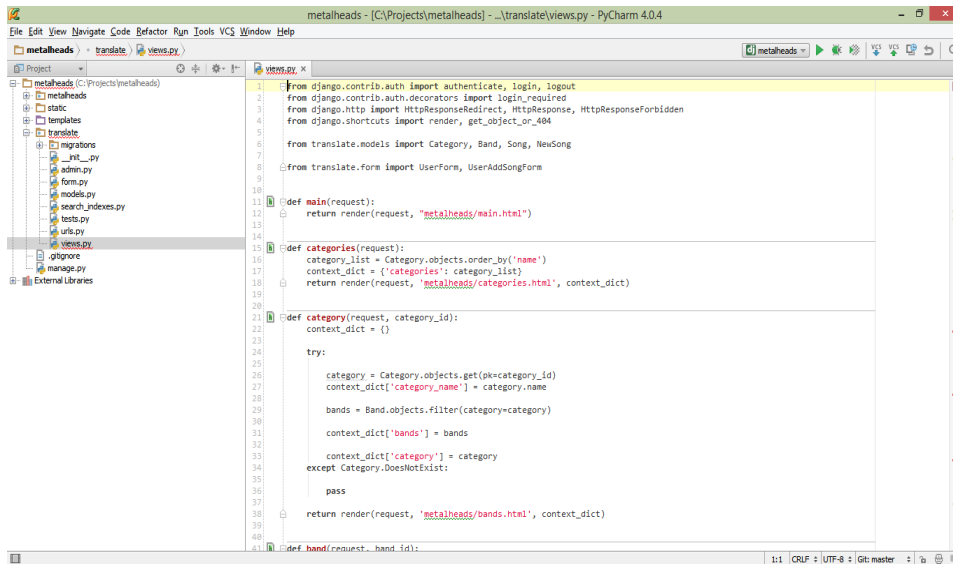
```
Python 3.4.1 (v3.4.1:c0e311e010fc, May 18 2014, 10:38:22) [MSC v
.1600 32 bit (Intel)] on win32
Type "copyright", "credits" or "license()" for more information.
>>> d = {'first': 8, 'second': 42}
>>> for (k, v) in d.items():
>>>     print(k, 'is', v)

second is 42
first is 8
>>>
```

The status bar at the bottom right indicates "Ln: 10 | Col: 4".

# Python Starter

## Интегрированная среда разработки (IDE)



Интегрированная среда разработки  
PyCharm

<http://www.jetbrains.com/pycharm/>

Интегрированная среда разработки, *IDE* (Integrated development environment) — система программных средств, используемая программистами для разработки программного обеспечения (ПО). Среда разработки обязательно включает в себя:

- текстовый редактор;
- компилятор и/или интерпретатор или средства интеграции с ним;
- средства автоматизации сборки;
- отладчик.

Также IDE может включать:

- средства интеграции с системами контроля версий;
- инструменты конструирования графического интерфейса пользователя.



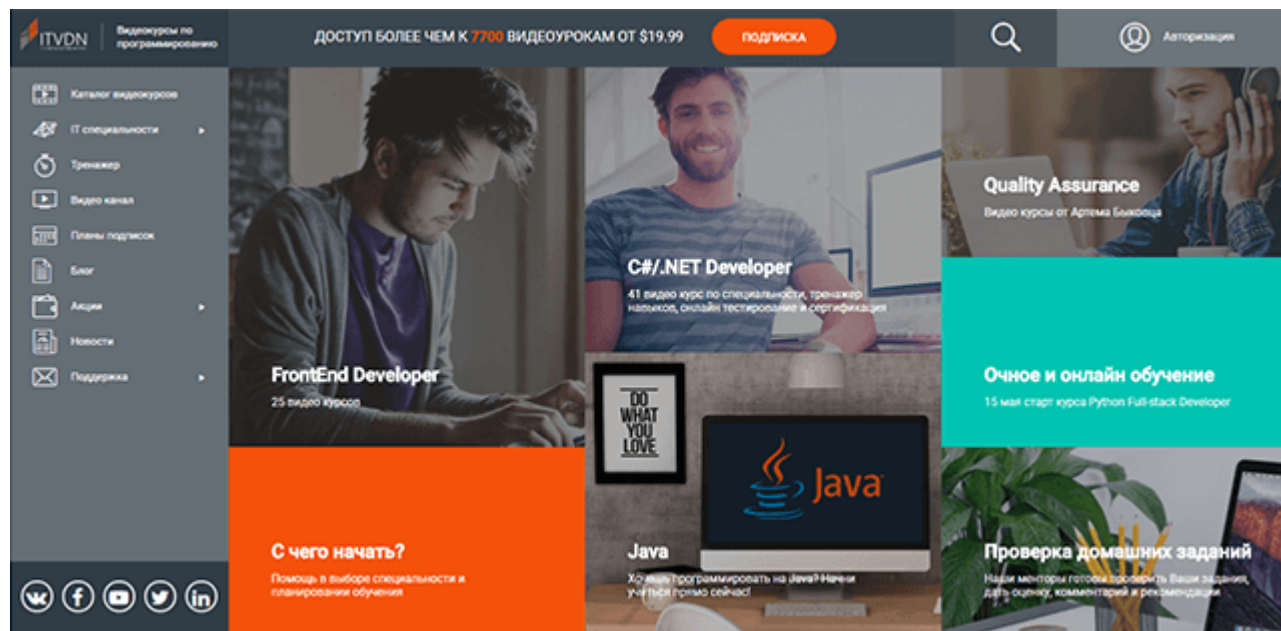
# Python Starter

Введение в Python

Практическая часть

# Смотрите наши уроки в видео формате

ITVDN.com



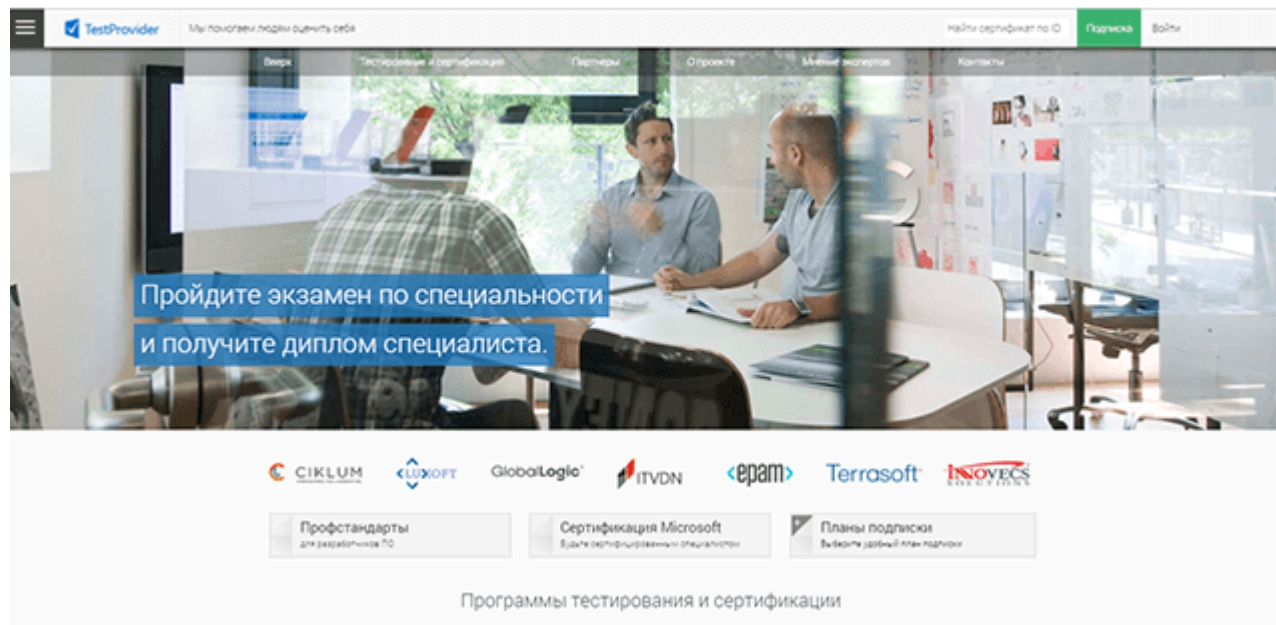
Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://ITVDN.com) для закрепления пройденного материала.

Курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics и другими высококвалифицированными разработчиками.



# Проверка знаний

TestProvider.com



TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



# Python Starter

Q&A

# Информационный видеосервис для разработчиков программного обеспечения

