

Python for the C# developer

Presented by JetBrains



by Michael Kennedy

DevelopMentor | [@mkennedy](#) | [michaelckennedy.net](#)

Agenda

- Introduce the basics of the Python language
- Review what is awesome about C# and .NET
- Explore Python's version of each C# / .NET feature

The Python language in 10 minutes

- High-level programming language
- Batteries included (large standard library)
- Interpreted (sometimes JIT compiled)
- Object-oriented (especially Python 3)
- Strongly-typed with dynamic semantics
- Syntax emphasizes readability
- Supports reuse through modules and packages



The 'shape' of a Python program

- Python defines code blocks (known as suites in Python) using whitespace and colons.

```
def somemethod(name):  
    if name == "Michael":  
        print("Hi old friend")  
    else:  
        print("Nice to meet you")  
        print("My name is ... ")
```

```
def main():  
    somemethod()
```

Demo

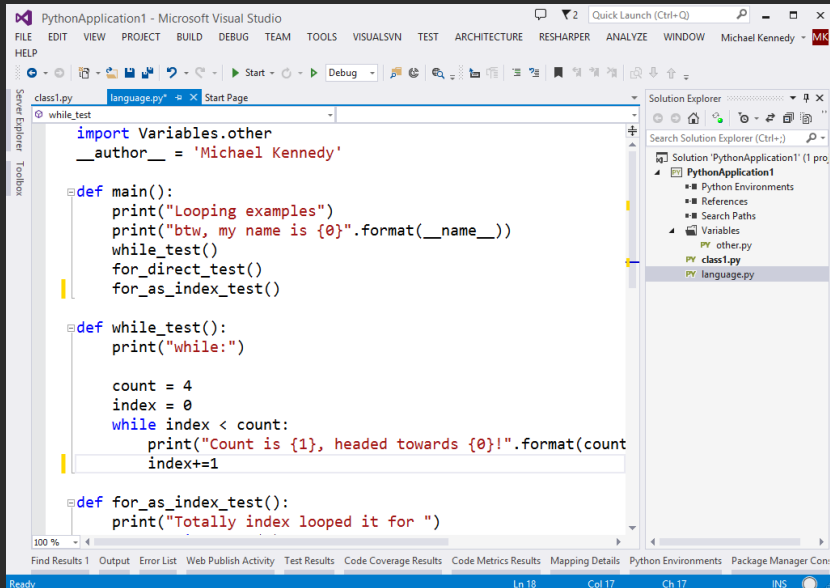
Python language demo

What's awesome about C# and .NET?

System.Object: Everything is an object.	LINQ
IEnumerable + foreach loops	Visual Studio / IDEs
Class properties (int Age {get; set;})	Side-by-side execution (isolation)
Anonymous types	Iterator methods / yield return
Add reference	Anonymous methods / lambdas / closures
NuGET package management	Base class libraries
Entity Framework / ORMs	JIT compilation
Great debugging tools	Resharper and IDE plugins
ASP.NET MVC	GUI designers

IDEs

C#



```
class1.py
import Variables.other
__author__ = 'Michael Kennedy'

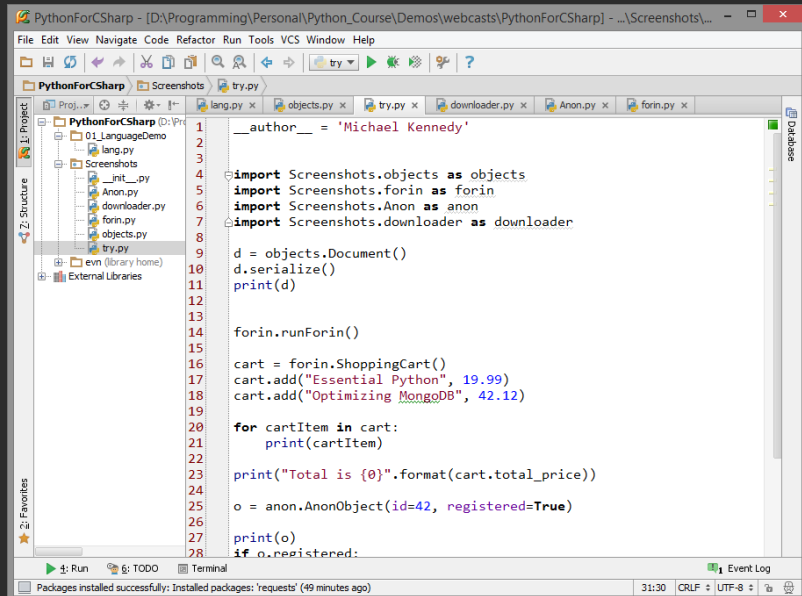
def main():
    print("Looping examples")
    print("btw, my name is {}".format(__name__))
    while_test()
    for_direct_test()
    for_as_index_test()

def while_test():
    print("while:")

    count = 4
    index = 0
    while index < count:
        print("Count is {1}, headed towards {0}!".format(count, index))
        index+=1

def for_as_index_test():
    print("Totally index looped it for ")
```

Python

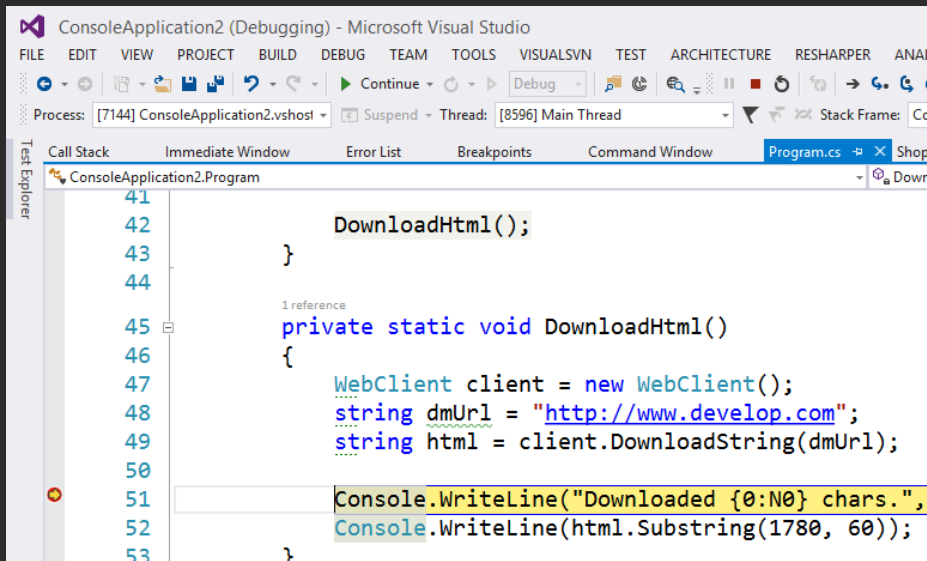


```
try.py
1 __author__ = 'Michael Kennedy'
2
3
4 import Screenshots.objects as objects
5 import Screenshots.forin as forin
6 import Screenshots.anon as anon
7 import Screenshots.downloader as downloader
8
9 d = objects.Document()
10 d.serialize()
11 print(d)
12
13
14 forin.runForin()
15
16 cart = forin.ShoppingCart()
17 cart.add("Essential Python", 19.99)
18 cart.add("Optimizing MongoDB", 42.12)
19
20 for cartItem in cart:
21     print(cartItem)
22
23 print("Total is {}".format(cart.total_price))
24
25 o = anon.AnonObject(id=42, registered=True)
26
27 print(o)
28 if o.registered:
```

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

Great debuggers

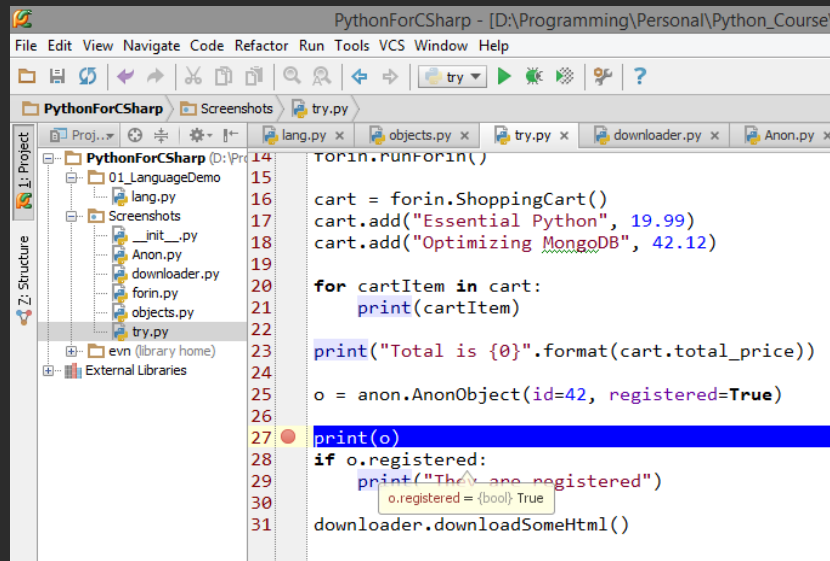
C#



The screenshot shows the Visual Studio IDE with a C# project named 'ConsoleApplication2 (Debugging)'. The 'Program.cs' file is open, and the 'DownloadHtml()' method is being debugged. The code defines a static method that uses a WebClient to download HTML from 'http://www.develop.com'. The console output shows 'Downloaded {0:N0} chars.' and the HTML content. The 'Debug' menu is open, and the 'Continue' button is highlighted.

```
41  
42     DownloadHtml();  
43 }  
44  
45 private static void DownloadHtml()  
46 {  
47     WebClient client = new WebClient();  
48     string dmUrl = "http://www.develop.com";  
49     string html = client.DownloadString(dmUrl);  
50  
51     Console.WriteLine("Downloaded {0:N0} chars.",  
52         Console.WriteLine(html.Substring(1780, 60));  
53 }
```

Python



The screenshot shows the PythonForCSharp IDE with a Python project named 'PythonForCSharp'. The 'try.py' file is open, and the 'forin.runForin()' method is being debugged. The code defines a ShoppingCart class and a forin class. The console output shows 'Total is {0}'. The 'Debug' menu is open, and the 'Continue' button is highlighted.

```
14 forin.runForin()  
15  
16 cart = forin.ShoppingCart()  
17 cart.add("Essential Python", 19.99)  
18 cart.add("Optimizing MongoDB", 42.12)  
19  
20 for cartItem in cart:  
21     print(cartItem)  
22  
23 print("Total is {0}".format(cart.total_price))  
24  
25 o = anon.AnonObject(id=42, registered=True)  
26  
27 print(o)  
28 if o.registered:  
29     print("They are registered")  
30     o.registered = (bool) True  
31     downloader.downloadSomeHtml()
```


Everything is an object

C#

```
class Document : object
{
    public void Serialize()
    {
        // ...
    }

    public override string ToString()
    {
        return "I am a document";
    }
}
```

Python

```
class Document( object ):

    def serialize(self):
        # ...

    def __str__(self):
        return "I am a document."
```

IEnumerable + foreach loops

C#

```
int[] numbers = new[] {1, 2, 3, 4, 5, 6};  
  
foreach (var n in numbers)  
{  
    Console.Write(n + ",");  
}
```

Python

```
numbers = [1, 2, 3, 4, 5, 6]  
  
for n in numbers:  
    print(n, end=', ')
```

IEnumerable + foreach loops

C#

```
class ShoppingCart : IEnumerable<Tuple<string, float>>
{
    List<Tuple<string, float>> cartItems =
        new List<Tuple<string, float>>();

    public void Add(string name, float price)
    {
        cartItems.Add(new Tuple<string, float>(name, price));
    }

    public IEnumerator<Tuple<string, float>>
        GetEnumerator()
    {
        return cartItems.GetEnumerator();
    }

    IEnumerator IEnumerable.GetEnumerator()
    {
        return GetEnumerator();
    }
}
```

Python

```
class ShoppingCart:

    def __init__(self):
        self.items = []

    def add(self, name, price):
        self.items.append( (name, price) )

    def __iter__(self):
        return self.items.__iter__()
```

Properties

C#

```
class ShoppingCart
{
    public float TotalPrice
    {
        get
        {
            float total = 0;
            foreach (var item in cartItems)
            {
                total += item.Item2;
            }

            return total;
        }
    }
}

Console.WriteLine("Total price: {0}", cart.TotalPrice);
```

Python

```
class ShoppingCart:

    @property
    def total_price(self):
        total = 0.0
        for item in self.items:
            total += item[1]

        return total

print("Total is {0}". \
      format(cart.total_price))
```

Anonymous objects

C#

```
var o = new
{
    Id = 2,
    Registered = true
};

Console.WriteLine(o);
// { Id = 2, Registered = True }

if (o.Registered)
{
    Console.WriteLine(
        "They are registered...");
}
```

Python

```
class AnonObject(dict):
    __getattr__ = dict.get
    __setattr__ = dict.__setitem__
```

```
o = AnonObject(id=42, registered=True)

print(o)
# {'registered': True, 'id': 42}

if o.registered:
    print("They are registered...")
```

Lambda expressions

C#

```
private static IEnumerable<int>
FindNumbers(Predicate<int> predicate)
{
    for (int i = 0; i < 100; i++)
    {
        if (predicate(i))
            yield return i;
    }
}

IEnumerable<int> nums =
    FindNumbers(n => n % 11 == 0)

// [0, 11, 22, 33, 44, 55, 66, 77, 88, 99]
```

Python

```
def numFilter(predicate):
    for i in range(100):
        if predicate(i):
            yield i

nums = numFilter(lambda n : n % 11 == 0)

# [0, 11, 22, 33, 44, 55, 66, 77, 88, 99]
```

LINQ

C#

```
var older =  
    from p in people  
    where p.age > 30  
    orderby p.age descending  
    select new {age = p.age, name = p.name}
```

Python

```
older = [  
    AnonObject(age = p.age, name = p.name)  
    for p in people  
    if p.age > 30  
]  
older.sort(key= lambda p : -p.age)
```

NuGET package management

C#

```
PM>Install-Package mongocsharpdriver
```

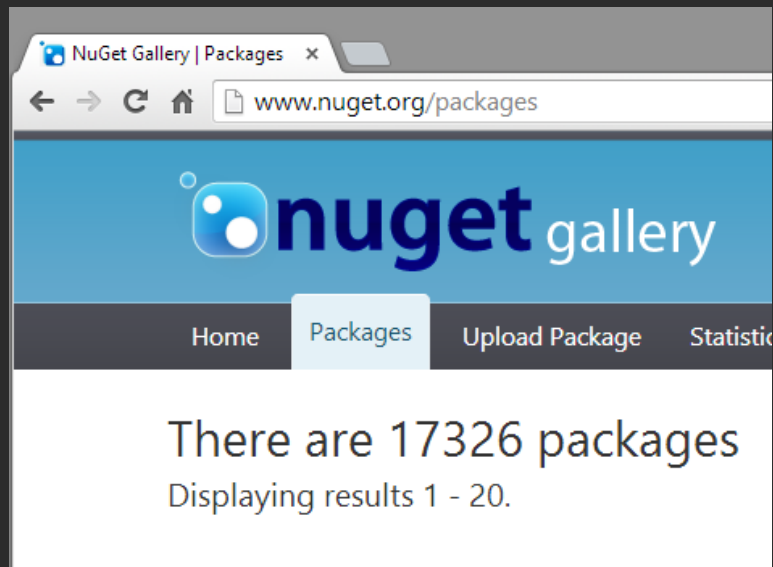
```
Installing 'mongocsharpdriver 1.8.3'.  
Successfully installed 'mongocsharpdriver 1.8.3'.  
Adding 'mongocsharpdriver 1.8.3' to YourApp.  
Successfully added 'mongocsharpdriver 1.8.3' to  
YourApp.
```

Python

```
c:\>pip install pymongo
```

```
Downloading/unpacking pymongo  
  Running setup.py egg_info for package pymongo  
  
Installing collected packages: pymongo  
Running setup.py install for pymongo  
Fixing build\lib.win-amd64-3.3\bson\binary.py  
  ...  
Successfully installed pymongo  
Cleaning up...
```


NuGET package management



18,594 packages



39,590 packages

Iterator methods / yield return

C# code

```
private static IEnumerable<int>
    FibonacciGenerator()
{
    int current = 1;
    int next = 1;

    yield return current;
    while (true)
    {
        int temp = current + next;
        current = next;
        next = temp;
        yield return current;
    }
}
```

Python code

```
def fibinacci_generator():
    current, nxt = 1, 1
    yield current

    while True:
        current, nxt = nxt, current + nxt
        yield current
```

ASP.NET MVC

C#



Python



Pyramid™



Flask

web development,
one drop at a time

Entity Framework

C#



Python



JIT Compilation

C#

Python

JIT compilation via CLR

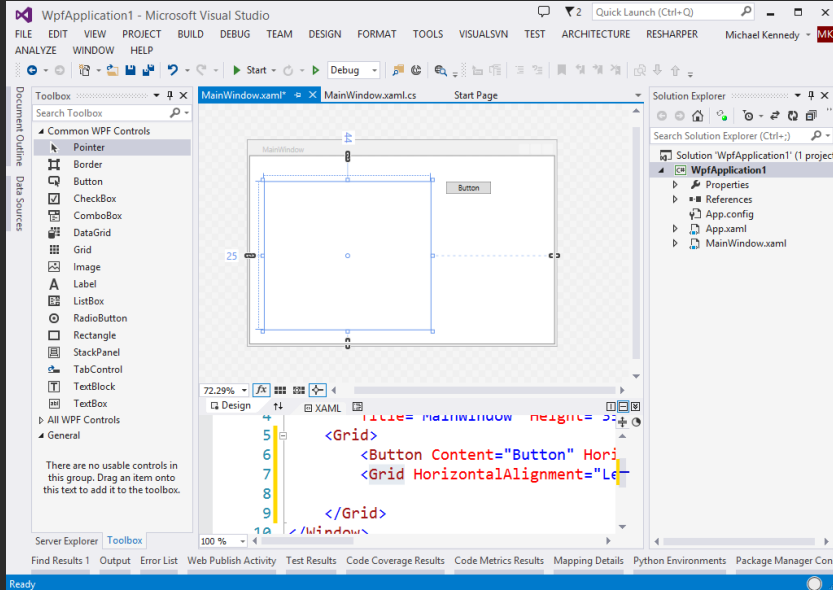


IronPython

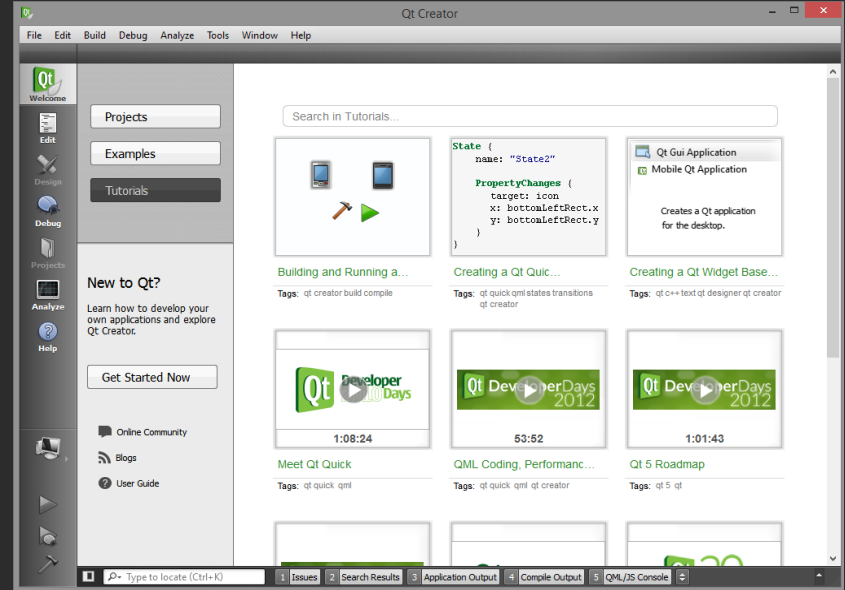


GUI Designer

C#



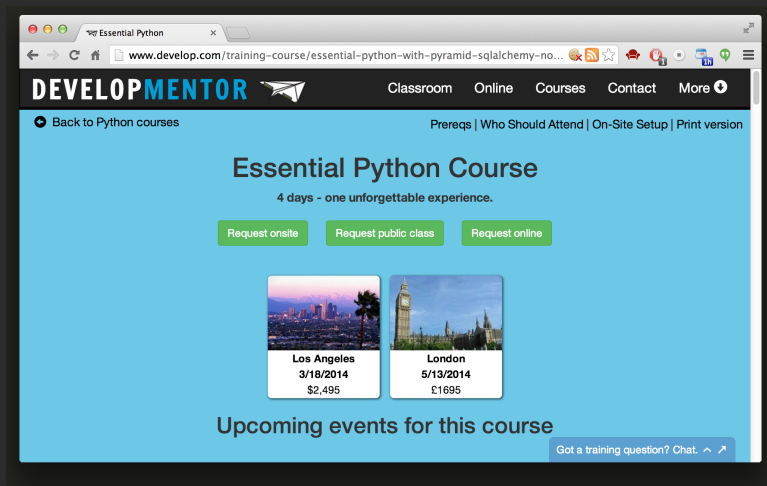
Python



Summary: Python for the C# developer

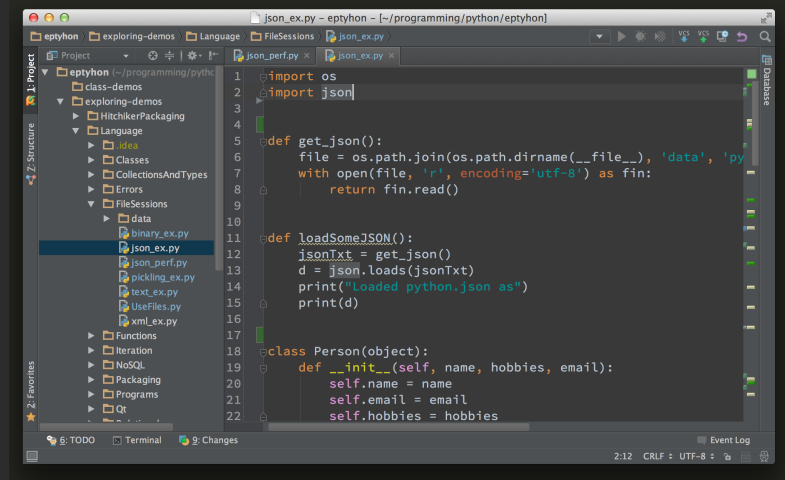
- Python language is simple, concise, and readable
- Many parts of C# and .NET are awesome
- Python often has equivalent features
 - sometimes nicer
 - sometimes less nice
- Python has a very capable IDE / Debugger in PyCharm

Thanks for coming, you should



The screenshot shows the DevelopMentor website with the 'Essential Python Course' prominently displayed. The page includes navigation links like 'Classroom', 'Online', 'Courses', 'Contact', and 'More'. Below the course title, it states '4 days - one unforgettable experience.' and provides buttons to 'Request onsite', 'Request public class', and 'Request online'. Two event listings are shown: one for Los Angeles on 3/18/2014 for \$2,495, and another for London on 5/13/2014 for £1695. At the bottom, there is a link to 'Got a training question? Chat.'.

Check out my Python course
from DevelopMentor



The screenshot shows the PyCharm IDE interface. On the left, the 'Project' tool window displays a file tree for a project named 'eptyhon'. The tree includes folders like 'class-demos', 'exploring-demos', 'Language', 'Classes', 'CollectionsAndTypes', 'Errors', 'FileSessions', 'data', 'binary_ex.py', 'json_ex.py', 'json_perf.py', 'pickling_ex.py', 'text_ex.py', 'UseFiles.py', and 'xml_ex.py'. The 'json_ex.py' file is selected. The main editor window shows the code for 'json_ex.py', which includes imports for 'os' and 'json', a 'get_json()' function, a 'loadSomeJSON()' function, and a 'Person' class with an '__init__' method.

Download and try PyCharm

Follow me online
[@mkennedy](https://twitter.com/mkennedy)
<http://blog.michaelckennedy.net>