Python for the C# developer

Presented by JetBrains

by Michael Kennedy

DevelopMentor | @mkennedy | michaelckennedy.net

DEVELOPMENTOR

Agenda

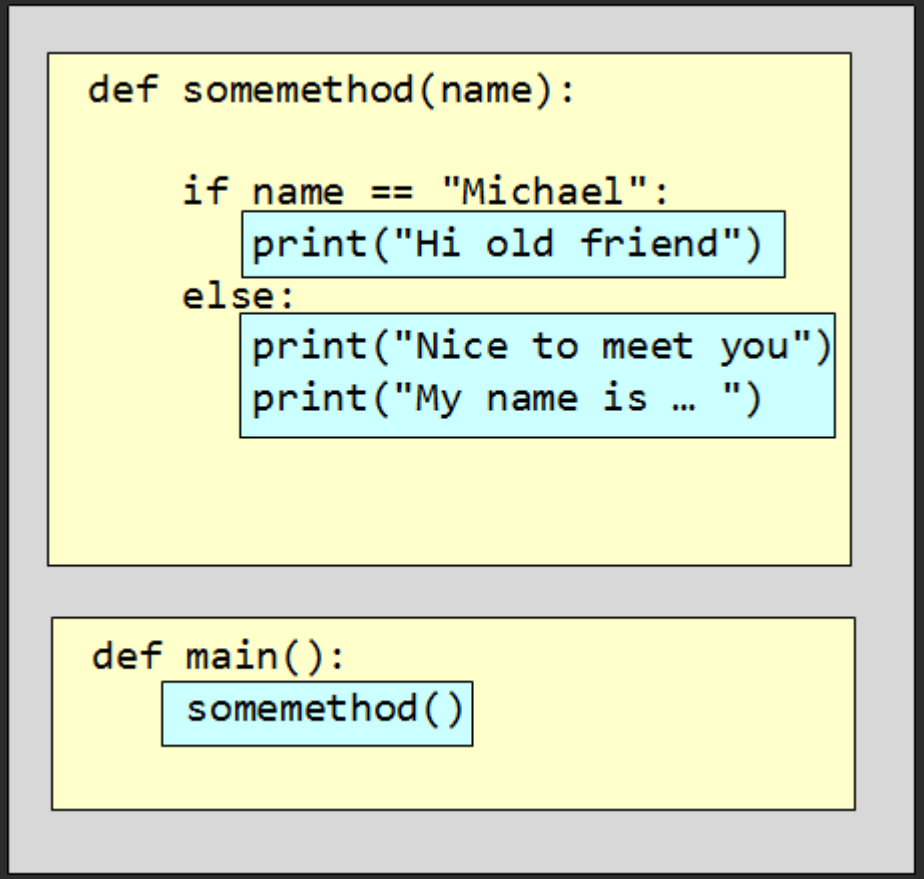
- Introduce the basics of the Python language
- Review what is awesome about C# and .NET
- Explore Python's version of each C# / .NET feature

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.



Demo

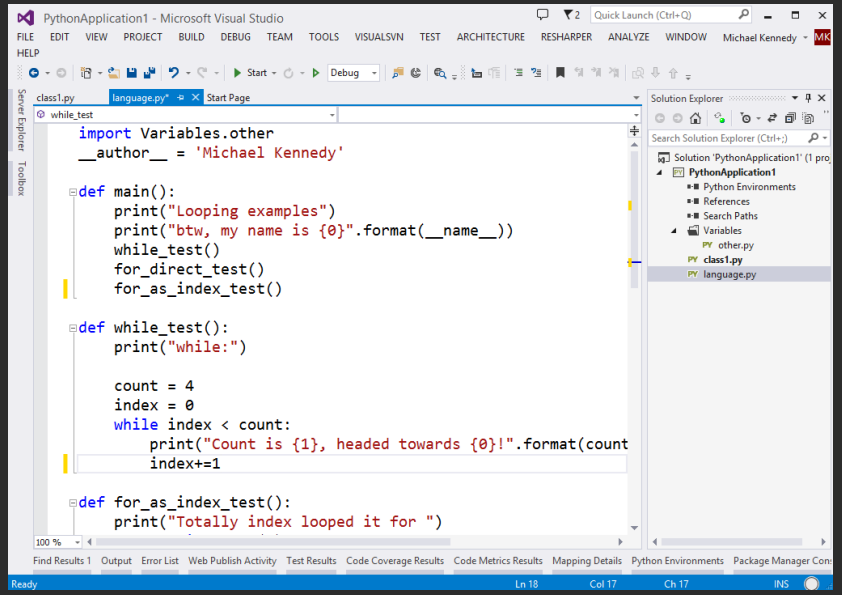
Python language demo

What's awesome about C# and .NET?

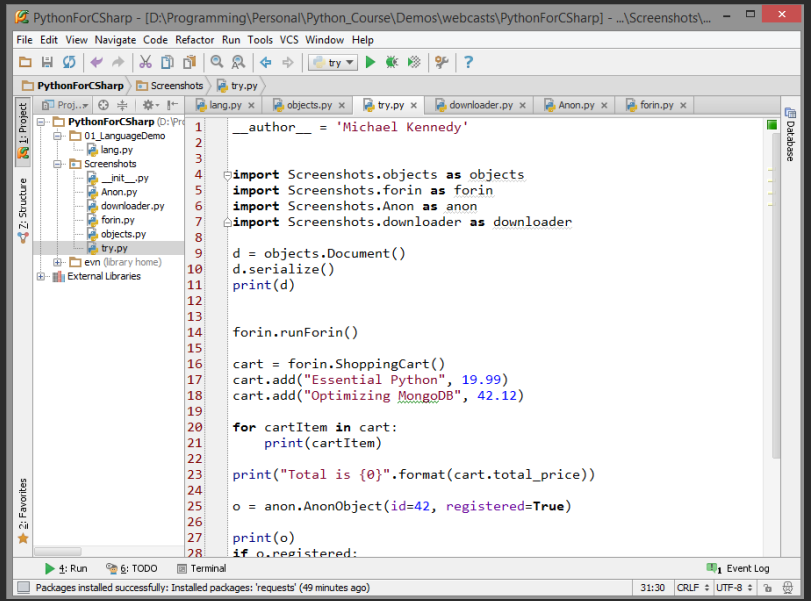
System.Object: Everything is an object.	LINQ
IEnumerable + foreach loops	Visual Studio / IDEs
Class properties (int Age {get; set;})	Side-by-side execution (isolation)
Anonymous types	Iterator methods / yield return
Add reference	Anonymous methods / lambdas / closures
NuGET package management	Base class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

IDEs

C#



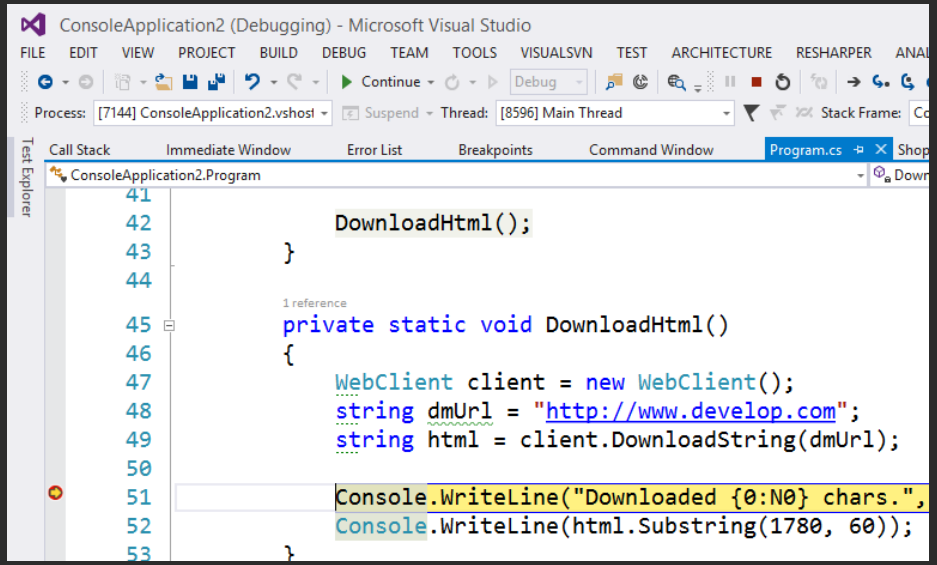
Python



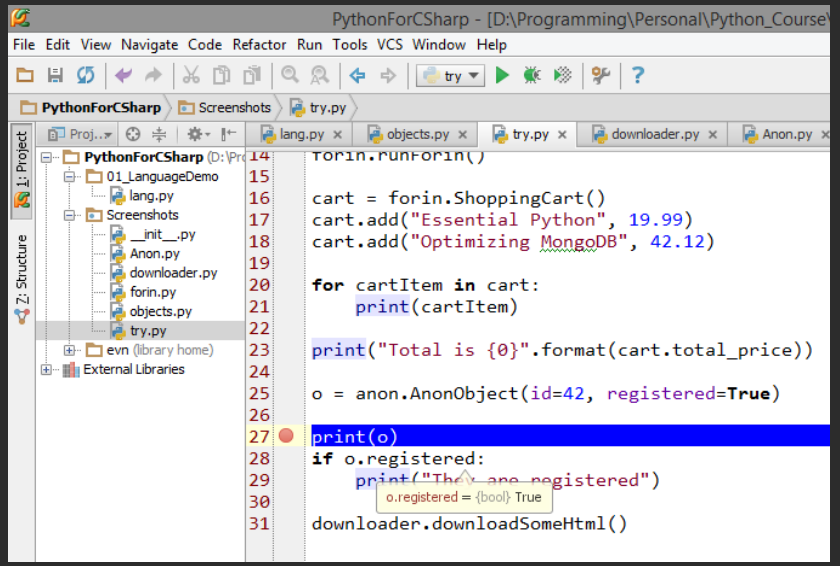
http://www.jetbrains.com/pycharm/

Great debuggers

C#



Python



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

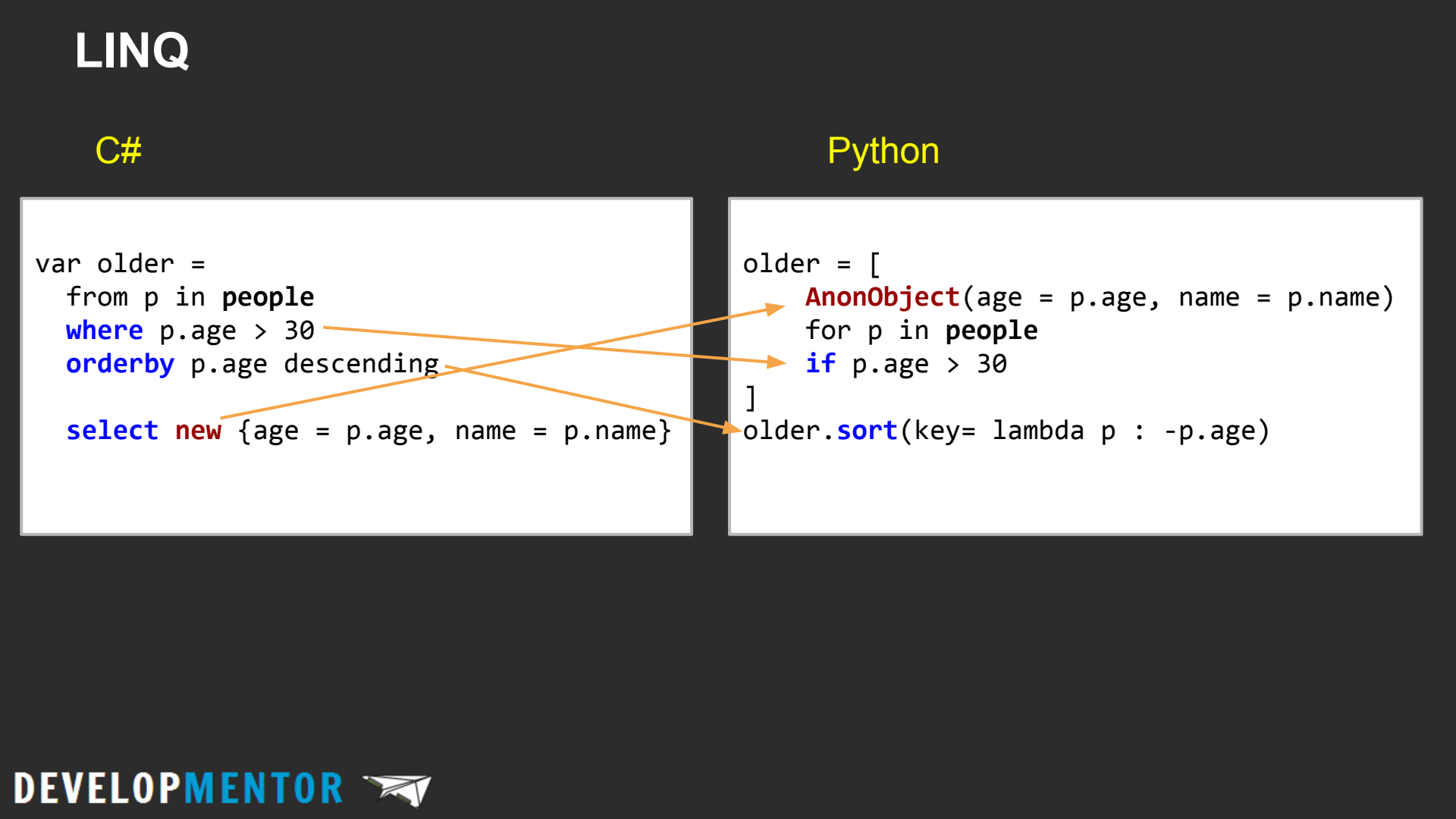
C#

```
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.
```

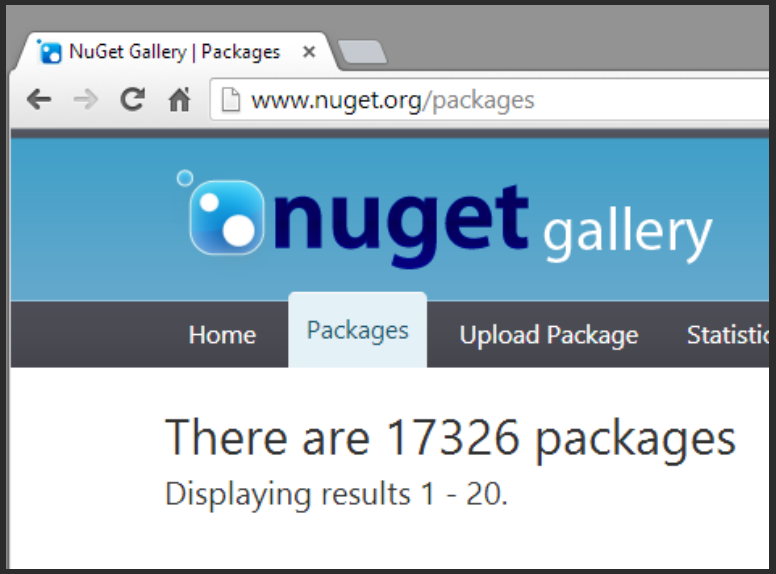
Python

```
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

Entity Framework

C#

Python

JIT Compilation

C#

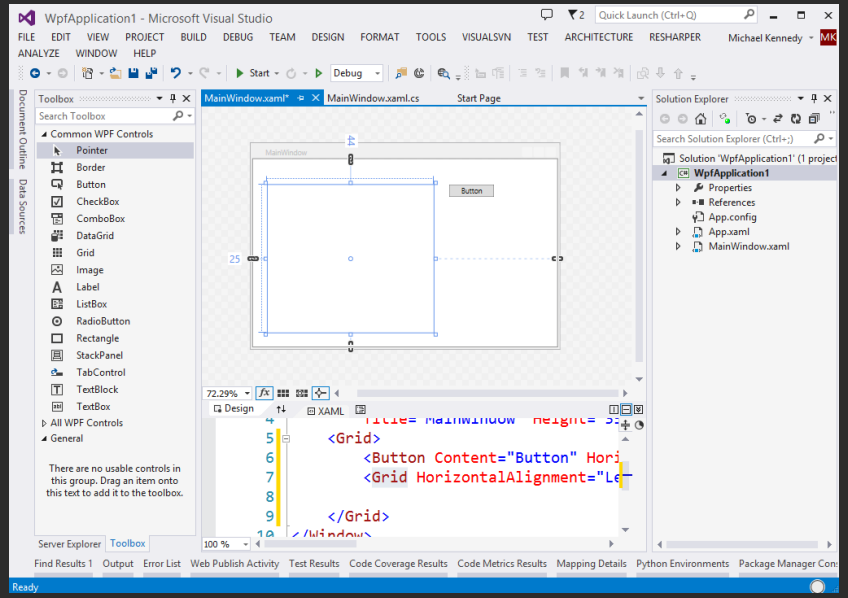
Python

JIT compilation via CLR

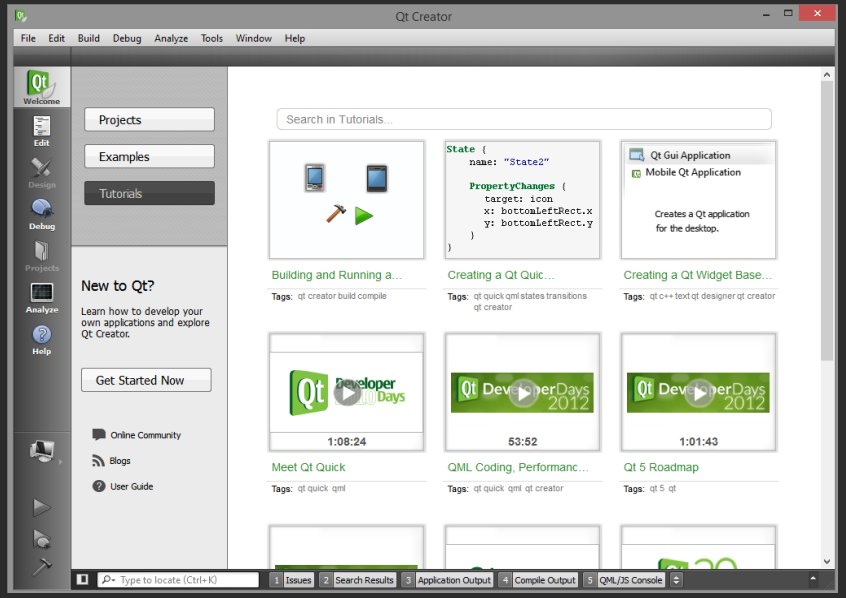
IronPython

GUI Designer

C#



Python

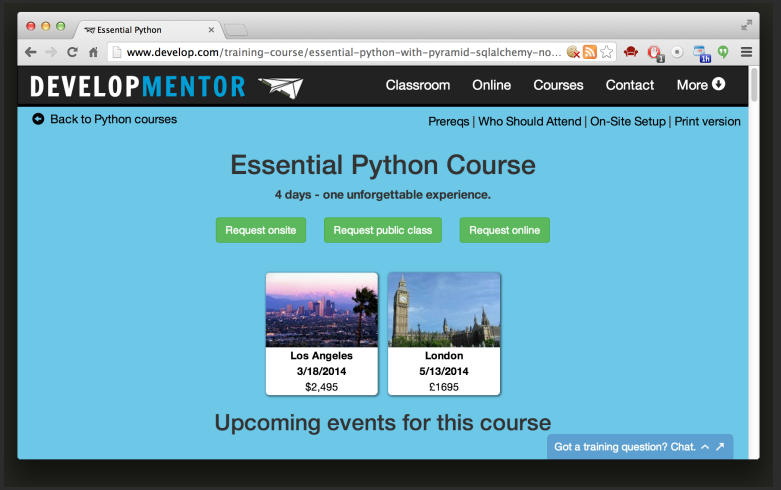


DEVELOPMENTOR

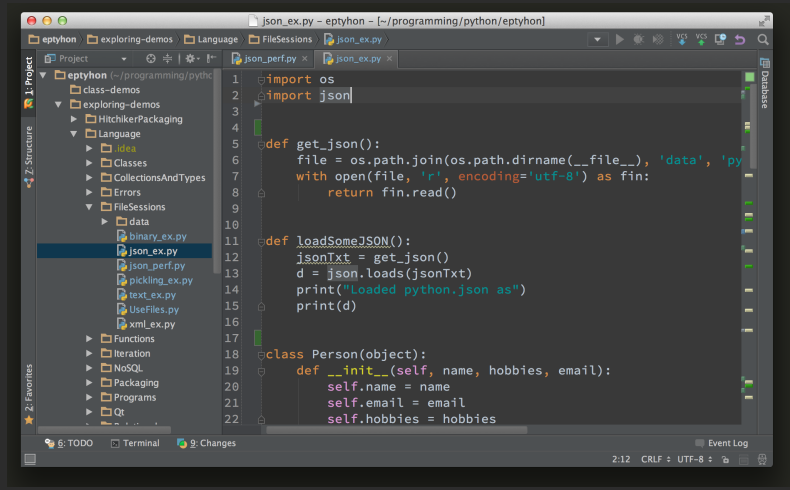
Summary: Python for the C# developer

- 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



Check out my Python course
from DevelopMentor



Download and try PyCharm

Follow me online
@mkennedy
http://blog.michaelckennedy.net

DEVELOPMENTOR