



python.net

Namrata M. Goldar¹, Vijay M. Rakhade², Lowlesh N. Yadav³

Student, Computer Science & Engineering, Shri Sai College of Engineering and Technology, Bhadrawati, India¹

Assistant Professor, Computer Science & Engineering, Shri Sai College of Engineering and Technology,
Bhadrawati, India²

Head of Department, Computer Science & Engineering, Shri Sai College of Engineering and Technology,
Bhadrawati, India³

Abstract:

Python:

We are aware that websites like YouTube and Dropbox make use of Python Programming in their source code? Python is a commonly used language that anyone can easily understand and apply. We can make nearly anything using Python. Most systems today (Mac, Linux, UNIX, etc.) have Python installed as a default setting since it is an open-source language and free language. Upon reading this book, we are going to become fluent in this awesome code language and see it applied to a variety of examples. No type of declaration of methodology, parameters, functions, or variables (like in other languages) is found in Python making its code concise and easy. As I said earlier, we can use language in everything if we want to build a website, make a game, or even create a search engine. The large plus of using Python is, an obvious compiler is not necessary since it's an entirely interpreted language

.Net:

When we hear the name .NET, it gives a feeling that it is something to do only with internet or networked applications. Level although it is correct that .NET provides a hard basis for increasing such requests it is likely to make a lot of new forms of demands. The following list determination give you an idea of various types of applications that we can develop. NET.

- ASP.NET Web applications: These comprise data-driven and dynamic browser-based programmers. Apps that use Windows Forms are the standard rich client applications.
- Console applications: These are the standard DOS types of programmers, such as batch scripts.
- Component Libraries: These are collections of parts that frequently include some business logic.
- Windows Custom Controls: You can create your own Windows controls, just like with conventional ActiveX controls.
- Web Custom Controls: By extending the concept of custom controls to web applications, code reuse and modularization are made possible.
- Web services: These are features that can be accessed via "web callable" protocols like HTTP, XML, and SOAP.

Keywords: Methods, Python Net installation

I. INTRODUCTION

The Python.NET package offers relatively seamless integration between Python programmers and the .NET Common Language Runtime (CLR) as well as a robust application scripting tool for .NET developers. It enables interaction between Python code and the CLR and can be used to integrate Python into .NET applications.

Python programmers may integrate with the .NET 4.0+ Common Language Runtime (CLR) on Windows and the Mono runtime on Linux and OS X thanks to the Python.NET package. For .NET developers, Python for .NET offers a potent application scripting tool. This package allows you to use .NET services and components developed in any language that targets the CLR (C#, VB.NET, F#, C++/CLI) to script .NET programs or create whole applications entirely in Python.

The Python runtime and API are integrated with Microsoft .NET Common Language Runtime (CLR) application code using the Python net (or Python.NET) package. As an alternative, you might utilize IronPython, which is existing.

The Python Package Index (PyPI) offers NET native Python net, which has the following features:



- CLR namespaces for .NET 4.0 and later are considered similar to Python packages.
- Combine Python applications with C#, VB.NET, F#, and C++/CLI-written .NET services and components.

Additionally crucial are several things Python net DOES NOT support that .NET developers using Visual Studio should be aware of, such as:

Produce managed code, or code that will only run when the CLR is in control, by converting Python into a .NET CLR language.

II. METHODS

A. Python net Installation Prerequisites

Python net is compatible with Python versions 2.7 and 3.5-3.8 from python.org.

You can define the type of Python you have connected to your PC by running:

B. python --version

The output should be similar to:

C. Python 3.8.2

If you do not have a recent version of Python installed, you can [install Python from Active State](#).

You can be checked if the pip is previously connected and up-to-date by incoming the following command:

D. pip --version

The output should be similar to:

E. pip 20.0.2 from C:\Python38\lib\site-packages\pip (python 3.8)

III. PYTHON NET INSTALLATION

How to install Python net on Linux, macOS, and Windows.

Pip Install Python net on Linux:

Python net and all of its dependencies can be installed on Linux using the following set of commands:

```
sudo apt-get install clang
```

```
sudo apt-get install libglib2.0-dev
```

```
sudo apt install git
```

```
sudo apt install NuGet
```

```
mono /usr/lib/nuget/nuget.exe config -set http_proxy=http://my.proxy.address:port
```

```
/usr/bin/python3 -m pip install -U pycparser --user=/usr/bin/python3 -m pip install -U
```

```
git+https://github.com/pythonnet/pythonnet
```

```
--user --egg
```

Depending on whether you are consuming a proxy or not, and also whether Git and the Nuget set are now installed, you can take out definite commands.

Pip Install Python net on macOS

Python net can be installed on macOS by doing the following:

Use Homebrew to install mono .NET:

```
brew install mono
```

Update the following environment variables:



```
export PATH=$PATH:/Library/Frameworks/Mono.framework/Versions/Current/bin
export PKG_CONFIG_PATH=/Library/Frameworks/Mono.framework/Versions/Current/lib/pkgconfig
```

Use pip to install Pycparser and Pythonnet:

```
pip install pycparser
pip install pythonnet
Pip Install Pythonnet on Windows
```

The simplest way to install Python net on Windows is to install the pre-compile Python net. why use the following command:

Build and Install Pythonnet From Source

1. Download the tar.gz source code archive from <https://pypi.org/project/pythonnet/#files>
2. Extract the archive and comment out line 424: `subprocess.check_call(cmd, shell=use_shell)`
3. Install Pythonnet, which has been packaged with distutils, using the following command:

python setup.py install

When you jump up Python in this manual (or else add this manual to sys. path), you can only **export the car** and jump by .NET assemblies.

IV. RESULTS

A. Calling .NET code from Python

Python.NET lets CLR namespaces to be preserved really as Python packages.

```
import clr
from System import String
from System.Collections import *
```

To load an assembly, use the `AddReference` function in the `car` module:

```
import clr
clr.AddReference("System.Windows.Forms")
from System.Windows.Forms import Form
```

By default, Mono will be used on Linux and macOS, and .NET Framework on Windows. For particulars on the loading of changed runtimes, please mention to the documents.

B. .NET Cor

If .NET Core is connected in a defaulting place or the `dotnet CLI` instrument is on the `PATH`, loading it in its place of the avoidance (`Mono/.NET Outline`) runtime just wants set each the surroundings moveable `PYTHONNET_RUNTIME=coreclr` or work python net. load clearly:

```
from python net import load
load("coreclr")
import clr
```

C. Embedding Python in .NET

- Data-driven and dynamic browser-based programs are included in ASP.NET Web apps.
- The most common rich client apps are those that use Windows Forms.
- Console applications: These are your typical DOS programs, like batch scripts.
- Component libraries are groupings of components that typically contain some business logic.
- Windows Custom Controls: Similar to traditional ActiveX controls, Windows controls can be customized by you.



- Web Custom Controls: Modularization and code reuse are made possible by applying the concept of custom controls to web applications.

Web services are functionalities that may be accessed by "webs callable" protocols like HTTP, XML, and SOAP.

Example:

```
static void Main(string[] args)
{
    PythonEngine.Initialize();
    using (Py. GIL())
    {
        dynamic np = Py.Import(<s>"numpy"</s>);
        Console.WriteLine(np.cos(np.pi * 2));
        dynamic sin = np.sin;
        Console.WriteLine(sin(5));
        double c = (double)(np.cos(5) + sin(5));
        Console.WriteLine(c);
        dynamic a = np.array(new List<float> { 1, 2, 3 });
        Console.WriteLine(a.dtype);
        dynamic b = np.array(new List<float> { 6, 5, 4 }, dtype: np.int32);
        Console.WriteLine(b.dtype);
        Console.WriteLine(a * b);
        Console.ReadKey();
    }
}
```

OUTPUT:

```
1.0
-0.958924274663
-0.6752620892
float64
int32
[ 6. 10. 12.]
```

V. DISCUSSION

A strong application scripting tool is provided for .NET developers by the Python.NET package, which allows Python programmers relatively seamless interaction with the .NET Common Language Runtime (CLR). It enables CLR interaction between Python code and is also capable of integrating Python into .NET applications.

This package doesn't implement Python as a first-class CLR language; more specifically, it doesn't translate Python code into managed code (IL). The CPython engine and a .NET runtime are combined instead.

With this method, Python code can still execute at native Python execution speeds while still using CLR services and existing Python code, and C-API extensions.



VI. RESOURCES

Information on installation, FAQ, troubleshooting, debugging, and projects using python net can be found in the Wiki:
<https://github.com/pythonnet/pythonnet/wiki>

Mailing list

<https://mail.python.org/mailman/listinfo/pythondotnet>

Chat<https://gitter.im/pythonnet/pythonnet>

VII. CONCLUSION

In this research paper, I explain how to integrate a Python script within a .NET application and redirect the Python's scripts' output to a .NET dialog form. The whole mixing is clear to the operator since they take no hint that Python is truly burden the back-end work. With the help of the Python.NET package, Python programmers may integrate their code with the .NET 4.0+ Common Language Runtime (CLR) on Windows and the Mono runtime on Linux and OS X. Python for .NET offers .NET developers a potent application scripting tool. You may construct full programmers in Python or script .NET applications using this package.

REFERENCES

- [1] "Download .NET Framework 4.8 Offline Installer". Microsoft. Archived from the new on August 15, 2019. Repossessed on August 15, 2019.
- [2] "Microsoft develops on the sector by open source". Opensource.com. November 19, 2014. Repossessed January 2, 2020.
- [3] ".NET Framework & Windows OS versions". docs.microsoft.com. Retrieved November 21, 2020.
- [4] "Standard ECMA-335: Common Language Infrastructure (CLI)". ECMA-international.org (6 ed.). ECMA. June 2012. Archived from the new on June 29, 2013. Repossessed August 31, 2005.
- [5] "Standard ECMA-334: C# Language Specification". ECMA-international.org (4 ed.). ECMA. June 2006. Archived from the news on October 31, 2010. Repossessed August 31, 2005.
- [6] "ISO/IEC 23271:2012 Data skill – Shared Semantic Infrastructure". iso.org (3 ed.). International Organization for Standardization. February 13, 2012.
- [7] "ISO/IEC 23270:2006 – Documents equipment– Program writing argots – C#". iso.org (2 ed.). International Organization for Standardization. January 26, 2012. Archived from the news on December 6, 2010. Repossessed April 1, 2008.
- [8] "Microsoft's Empty Promise". Free Software Foundation. 16 July 2009. Archived from the news on August 19, 2009.