

# Python's Next Decade and Us

Let's brainstorm where Python will grow in the 2020s! Who will be using Python in 2030 and why? How will people use Python? Where will Python fit in the global computing community? What can we do to make our ideal Python over the next decade? What TLC will help Python the most?



Introduction and why and how  
I got into Python: 4 minutes





## USING A FUNCTION

A function, sometimes referred to as a subroutine, contains a block of code that performs a specific task, such as displaying a message. Using functions makes it easy to re-use sections of code in a PHP page. For example, you may have a function that displays a warning message when a client enters invalid data into a form. Instead of retyping the section of code that displays the message for each field in the form, you can simply re-use the function.

Functions also allow you to group lines of code into smaller, more manageable sections. This makes it easier to understand and troubleshoot the code.

A function is created using the `function` keyword followed by the name of the function and a set of parentheses. A function name must start with a letter or an underscore (`_`) character. The name should identify the purpose of the function or describe the action the function performs.

It is common to have a function name begin with a capital letter. If the name consists of multiple words, you can

capitalize the first letter of each word to make the name easy to read, such as `AddTotalCost`. Function names are case-insensitive, but you should be consistent in naming functions to help make your scripts easier for other people to understand.

The block of code used in a function is enclosed in braces (`{}`). The block of code will not be executed until the function is called in the script. To call a function, you type the name of the function followed by a set of parentheses where you want to execute the code.

PHP has many useful built-in functions, such as the `print` function. To avoid unnecessary work, you should check whether there is an existing function that suits your needs before defining your own function.

### Apply It

You can call a function from within another function. When calling other functions from within a function, you should be careful not to generate a loop that causes the functions to continuously call each other.

**TYPE THIS:**

```
function WelcomeMessage()
{
    print "Welcome to my Web page!<br>";
}
function DisplayIntroduction()
{
    WelcomeMessage();
    print "My name is Martine.<br>";
}
DisplayIntroduction();
```

**RESULT:**

Welcome to my Web page!  
My name is Martine.

When creating a function, you should keep in mind that any variables created within the function can be accessed only within that function. The function's variables will not be accessible to other functions or other sections of code in the PHP page.

**TYPE THIS:**

```
function SetPageTitle()
{
    $pageTitle = "Martine's Home Page";
}
function DisplayWelcome()
{
    print "Welcome to $pageTitle<br>";
}
DisplayWelcome();
```

**RESULT:**

Welcome to

### USING A FUNCTION

```
Untitled - Notepad
File Edit Search Help
<html>
<head>
<title>Using a Function</title>
</head>
<body>

<?php
function MyMessage()
{
$userName = "Tom";
if ($userName != "")
{
    print "Hello $userName,<br>";
}
else
{
    print "Hello new visitor,<br>";
}
?>
</body>
</html>
```

**CREATE A FUNCTION**

1 To create a function, type `function`.

2 Type a name for the function followed by () .

```
Untitled - Notepad
File Edit Search Help
<html>
<head>
<title>Using a Function</title>
</head>
<body>

<?php
function MyMessage()
{
$userName = "Tom";
if ($userName != "")
{
    print "Hello $userName,<br>";
}
else
{
    print "Hello new visitor,<br>";
}
?>
</body>
</html>
```

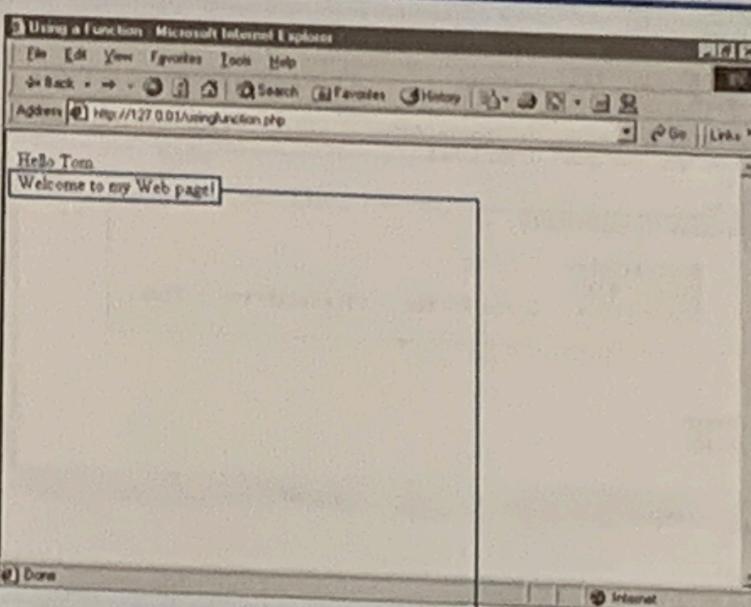
3 Type the code you want to execute when the function is called. Enclose the code in braces.

```
Untitled - Notepad
File Edit Search Help
<html>
<head>
<title>Using a Function</title>
</head>
<body>

<?php
function MyMessage()
{
$userName = "Tom";
if ($userName != "")
{
    print "Hello $userName,<br>";
}
else
{
    print "Hello new visitor,<br>";
}
?>
</body>
</html>
```

**CALL A FUNCTION**

4 To call a function, type the function name followed by () .



5 Display the PHP page in a Web browser.

The Web browser displays the result of the function.

Done  
6/17/02



## Web Development

- [Administration \(34\)](#)
- [Books \(4\)](#)
- [BrainDump \(15\)](#)
- [DHTML \(5\)](#)
- [Flash \(9\)](#)
- [Java \(17\)](#)
- [JavaScript \(7\)](#)
- [Multimedia \(5\)](#)
- [MySQL \(23\)](#)
- [Perl \(26\)](#)
- [PHP \(90\)](#)
- [Practices \(14\)](#)
- [Python \(14\)](#)
- [Reviews \(4\)](#)
- [Style Sheets \(2\)](#)

## [Democracy, The PHP Way](#)

Contributed by [Vikram Vaswani](#)

Article Rating: / 2

2001-04-16

Poor      Best [Rate](#)



### JUMP TO:

- [Democracy, The PHP Way](#)
- [The Plan](#)
- [Design View](#)
- [Start Me Up](#)
- [Vote Now, Or Forever Hold Your Peace](#)
- [The Number Game](#)
- [Down Memory Lane](#)
- [One Picture, A Thousand Words](#)
- [Cookie-Cutter Code](#)
- [Adding More...](#)

advertisement

## [Democracy, The PHP Way \( Page 1 of 10 \)](#)

In your travels across the Web, you've probably seen (maybe even participated in) online polls, quick measurements of what visitors to a site think of the hot-button issues of the day. And back when portals where a Good Thing, online polls appeared on each and every one of them; they were - and still are - a simple and effective method of promoting a sense of community amongst the visitors to a Web site, and of generating demographic data on a site's visitors.

## # Dev Shed

- » [XForms Basics, Part 2 in: XML](#)
- » [XForms Basics in: XML](#)
- » [Python UnZipped in: Python](#)

[More Open Source Help](#)

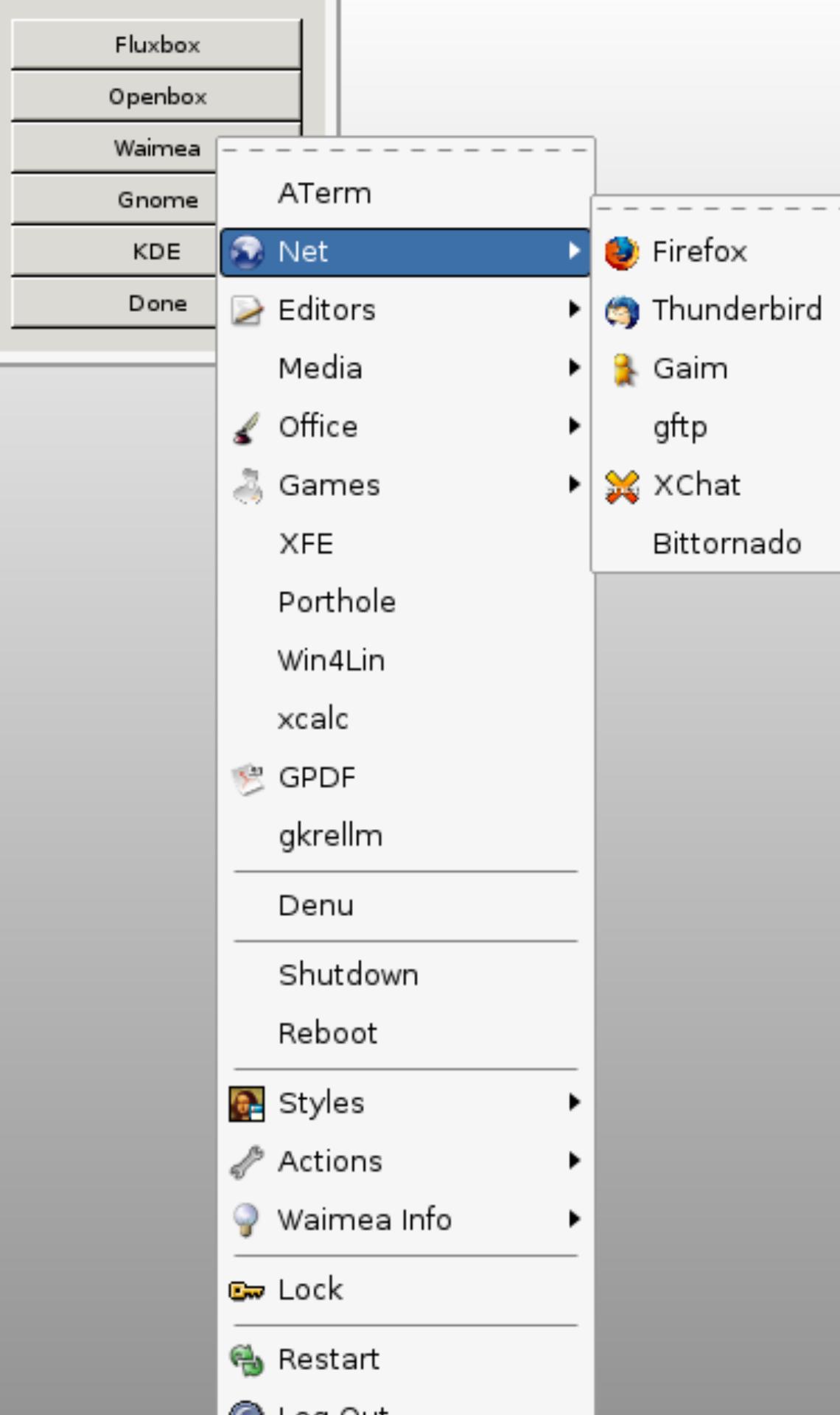
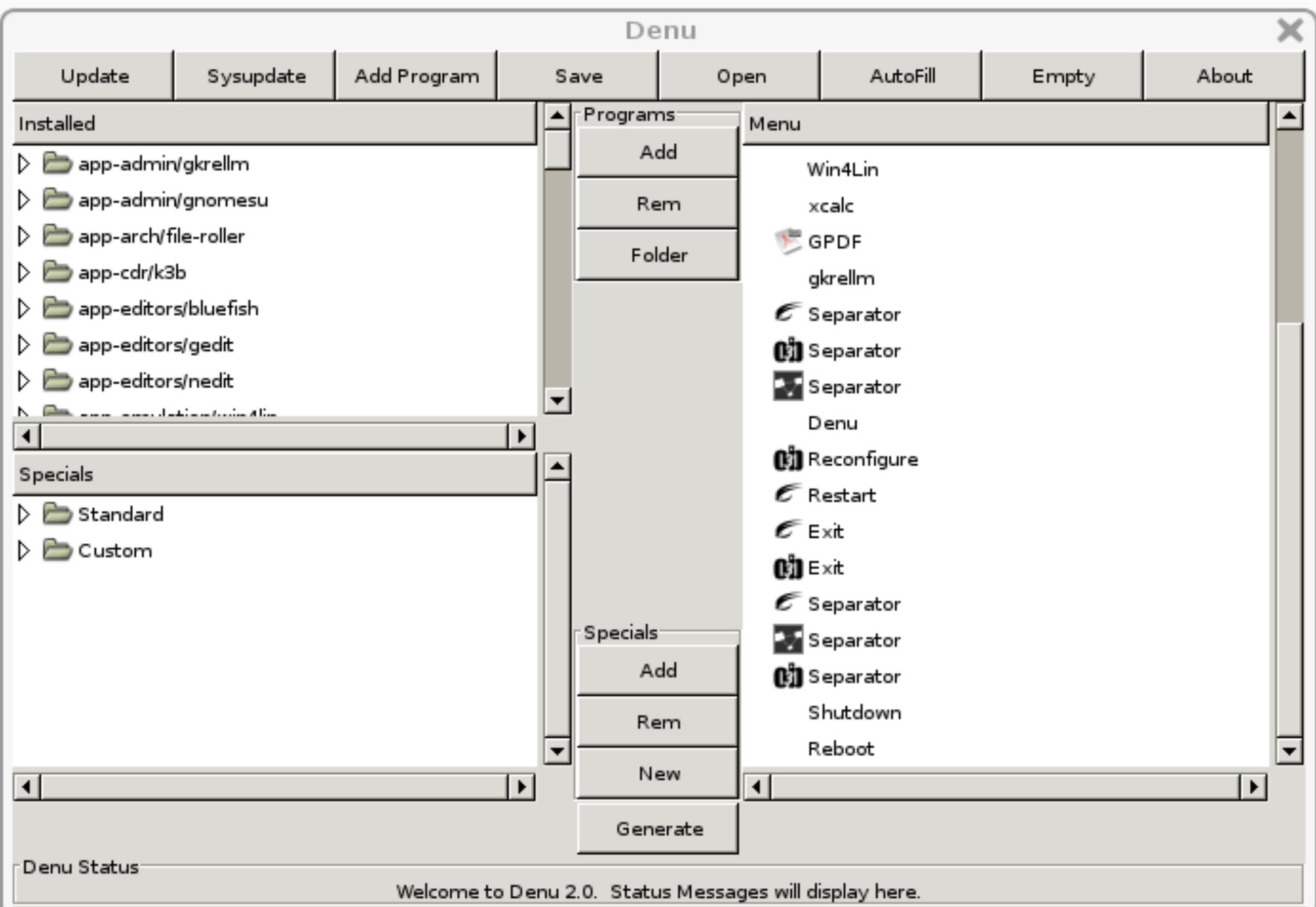
## ASP Free

### Hardware Reviews!

- » [Automatic Row Height in DataGrids in: C#](#)
- » [Migrating from ASP to ASP.NET in: ASP.NET](#)
- » [What is the .NET Framework? in: .NET](#)

[More ASP Help](#)

## Dev Articles

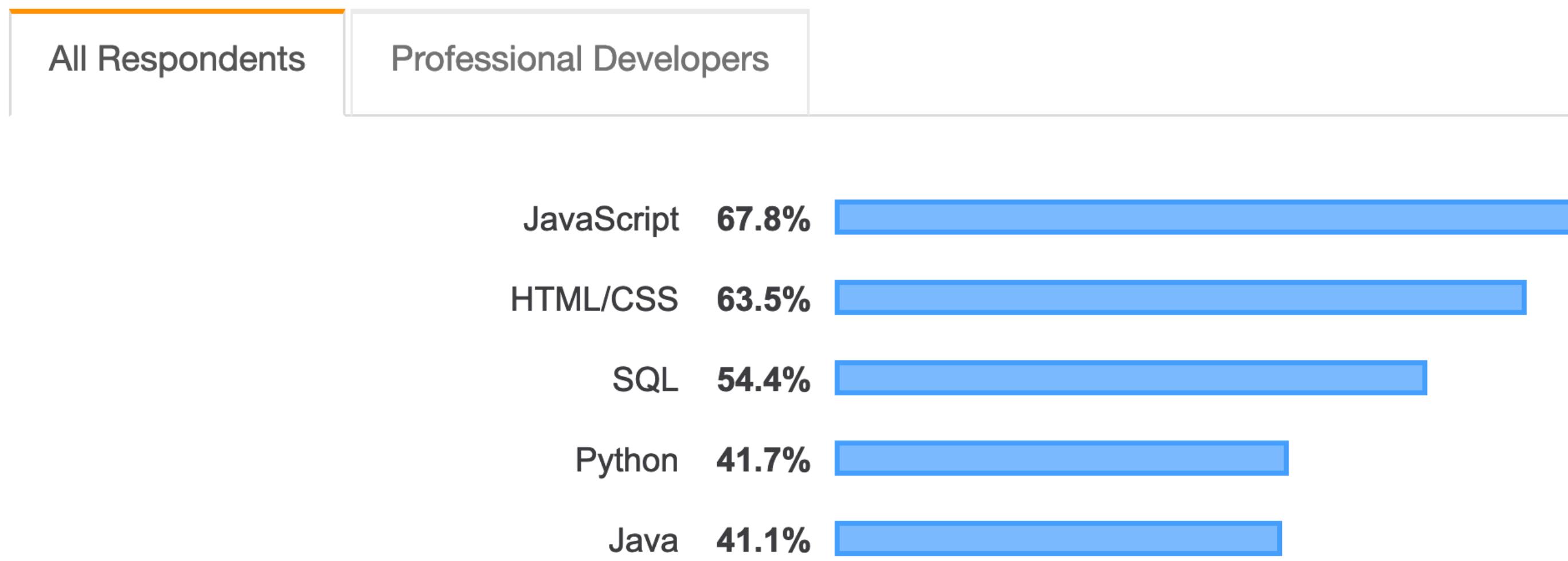


# **What about you?**

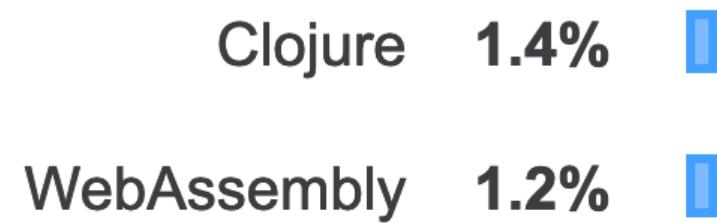


## Most Popular Technologies

### Programming, Scripting, and Markup Languages



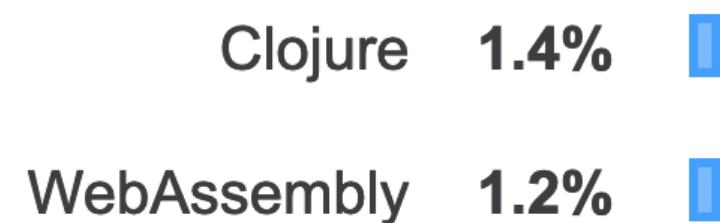
Ok, lets talk stats. Python is #4 in use according to StackOverflow 2019 and #2 in most loved! Does that matter? <https://insights.stackoverflow.com/survey/2019#technology>



87,354 responses; select all that apply

For the seventh year in a row, **JavaScript** is the most commonly used programming language, but **Python** has risen in the ranks *again*. This year, Python just edged out Java in overall ranking, much like it surpassed C# last year and PHP the year before. Python is the fastest-growing major programming language today.

[https://  
insights.stackoverflow.com/  
survey/2019#technology](https://insights.stackoverflow.com/survey/2019#technology)



87,354 responses; select all that apply

For the seventh year in a row, **JavaScript** is the most commonly used programming language, but **Python** has risen in the ranks *again*. This year, Python just edged out Java in overall ranking, much like it surpassed C# last year and PHP the year before. Python is the fastest-growing major programming language today.

[https://  
insights.stackoverflow.com/  
survey/2019#technology](https://insights.stackoverflow.com/survey/2019#technology)

# **87,354 responses**

**NOPE!** We shouldn't be in this to be number 1. We're all here because Python has empowered us to do something with computers. Computers are one of the most important tools human kind has every created. Being able to use them is the most important thing. What language it is doesn't matter.

**7,794,799,000 people**

[https://population.un.org/  
wpp/DataQuery/ for 2020](https://population.un.org/wpp/DataQuery/)

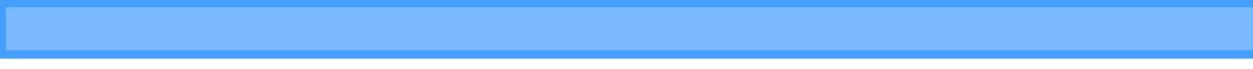


## Most Popular Technologies

### Programming, Scripting, and Markup Languages

All Respondents

Professional Developers

Not Coders 99.99% 

Coders 0.001% 

Here is the fixed chart.

**23,900,000 developers**

OK, I can hear the rebuttals coming that 87k is too low. Well an estimate of the number of developers worldwide pegged the number at 23.9 million. So I've adjusted the chart for it.

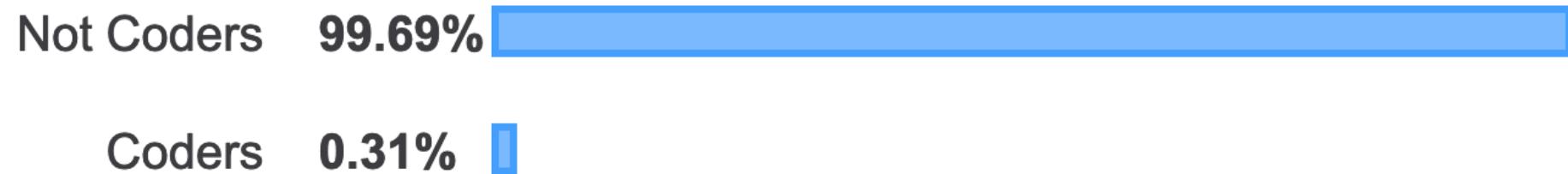


## Most Popular Technologies

### Programming, Scripting, and Markup Languages

All Respondents

Professional Developers



Out 7.6 billion people on Earth  
only an estimated 23.9 million  
(with an m) are developers. That  
is 0.3% of the world's population.  
[https://evansdata.com/reports/  
viewRelease.php?reportID=9](https://evansdata.com/reports/viewRelease.php?reportID=9)

**261,000,000**   
**vs**  
**1,524,838,000** 

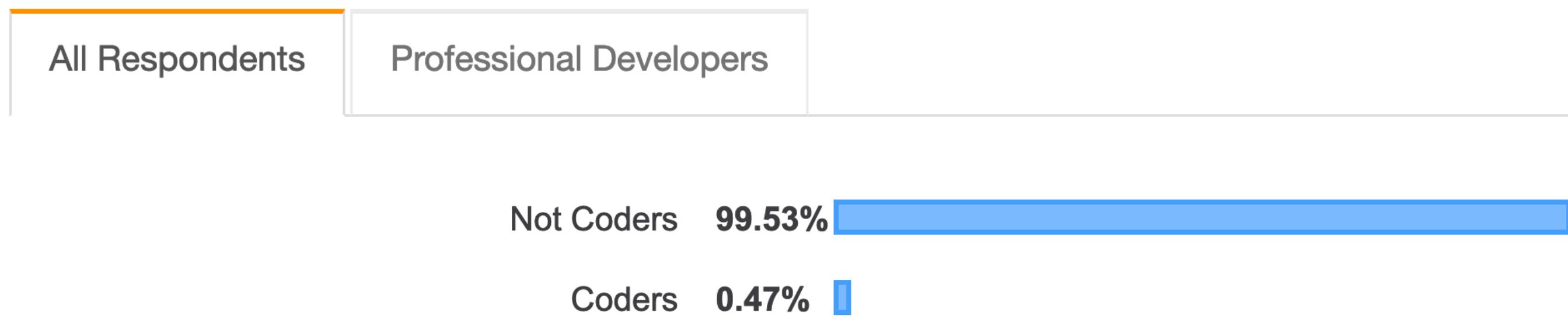
<https://www.gartner.com/en/newsroom/press-releases/2020-01-13-gartner-says-worldwide-pc-shipments-grew-2-point-3-percent-in-4q19-and-point-6-percent-for-the-year>

^ <https://www.gartner.com/en/newsroom/press-releases/2020-01-28-gartner-says-worldwide-smartphone-sales-will-grow-3-->



## Most Popular Technologies

### Programming, Scripting, and Markup Languages



Oh, and some of those folks don't even have a computer. Let's just count folks who do. At the end of 2018 there were 5.1 billion people with mobile services. <https://www.gsmaintelligence.com/research/?file=b9a6e6202ee1d5f787cfabb95d3639c5&download> This final adjusted version shows that 0.47% of all folks with computer access, 23 million-ish, are developers. Let's not be afraid of other languages. They are doing many interesting things that we can learn from. Plus we're all on the same side. 7 minutes



# Rust

[GET STARTED](#)[Version 1.41.0](#)

A language empowering everyone to build reliable and efficient software.

## Why Rust?

### Performance

Rust is blazingly fast and memory-efficient: with no runtime or garbage collector, it can power performance-critical services, run on embedded devices, and easily integrate with other

### Reliability

Rust's rich type system and ownership model guarantee memory-safety and thread-safety — enable you to eliminate many classes of bugs at compile-time.

### Productivity

Rust has great documentation, a friendly compiler with useful error messages, and top-notch tooling — an integrated package manager and build tool, smart multi-editor support with auto-

# Rust's memory model.



## Rust's Community Automation

6/28/2016



[talks.edunham.net/codeconfLA2016](http://talks.edunham.net/codeconfLA2016)

@qedunham

codeconfLA@edunham.net

**[https://youtu.be/407nwX6\\_\\_70](https://youtu.be/407nwX6__70)**

Rust community.

# Languages

The React documentation is available in the following languages:

---

Arabic

[العربية](#)

[Contribute](#)

Azerbaijani

[Azərbaycanca](#)

[Contribute](#)

English

[English](#)

[Contribute](#)

---

Spanish

[Español](#)

French

[Français](#)

Italian

[Italiano](#)

React's docs being translated.  
<https://reactjs.org/languages/>

## The Lorenz Differential Equations

Before we start, we import some preliminary libraries. We will also import (below) the accompanying `lorenz.py` file, which contains the actual solver and plotting routine.

```
[ ]: %matplotlib inline  
from ipywidgets import interactive, fixed
```

We explore the Lorenz system of differential equations:

$$\begin{aligned}\dot{x} &= \sigma(y - x) \\ \dot{y} &= \rho x - y - xz \\ \dot{z} &= -\beta z + xy\end{aligned}$$

Let's change ( $\sigma$ ,  $\beta$ ,  $\rho$ ) with ipywidgets and examine the trajectories.

```
[ ]: from lorenz import solve_lorenz  
w=interactive(solve_lorenz,sigma=(0.0,50.0),rho=(0.0,50.0))  
w
```

For the default set of parameters, we see the trajectories swirling around two points, called attractors.

The object returned by `interactive` is a `Widget` object and it has attributes that contain the current result and arguments:

Collab and Jupyter are doing very cool things. Bring prose-first, interactive computing to many programming languages including Python.



Donate



Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

```
# Python 3: List comprehensions
>>> fruits = ['Banana', 'Apple', 'Lime']
>>> loud_fruits = [fruit.upper() for
fruit in fruits]
>>> print(loud_fruits)
['BANANA', 'APPLE', 'LIME']
```

```
# List and the enumerate function
>>> list(enumerate(fruits))
[(0, 'Banana'), (1, 'Apple'), (2,
'Lime')]
```

## Compound Data Types

Lists (known as arrays in other languages) are one of the compound data types that Python understands. Lists can be indexed, sliced and manipulated with other built-in functions. [More about lists in Python 3](#)

1 2 3 4 5

# What should we do?

# **What will Python be in 10 years?**

Today Python can, automate tasks, serve webpages, process tons of data and make an led blink. How many of those 5.1 billion phone users want to do these things? What will we be able to do with Python a decade from now?

**TLC**

**TLC 4 minutes**

# Teach

Help empower others by teaching them Python. There is no better way to discover the bumps and bruises taken when learning something new than to teach it to someone.

# Listen

Listen to others about what they need from the computers they use everyday. Think of someone you know who doesn't program. What tasks could they automate with computers? What would it take to use Python? Help other's outside of your field who also use Python. Data scientists talk to web devs and vice versa.

# Contribute

There is no such thing as too easy. Each little bump in the road can throw someone off.

Smooth over every bump that folks hit when learning Python even if it means writing docs or another programming language.

Computer languages are almost all English, do whatever it takes to bring computing to another's spoken language.

# Computing's Python's Next Decade and Us Everyone