opensource.com
10 year anniversary

Supported by Red Hat

LOG IN SIGN UP

Articles Resources Downloads About Open Organization

By the numbers: Python community trends in 2017/2018

Python Developers Survey generates a wide range of data about the current use and trajectory of the Python programming language.

01 May 2018 | Ewa Jodlowska | 232

Image by : Opensource.com

This article was co-written with Dmitry Filippov.

Python is rapidly growing in popularity and drawing more attention in tech news, including reports about Python being used involve more high school girls in computer engineering and a recommendation that intro-level college computing classes teach the programming language. Additionally, Stack Overflow's 2018 developer survey found Python is the programming language the most people want to learn. Python is seen and used across the web, from simple personal websites to data mining and machine learning at the world's largest banks.

What makes Python so special? Who are Python developers? And why is it so popular? To answer these and many other important questions, JetBrains and the Python Software Foundation (PSF) partnered on a developer survey of people who use Python as a primary or supplementary language. Until now, there hasn't been a Python-specific study to learn how it is being used by diverse developers, what components complement its use, and what makes it one of the most loved languages.

The survey's goal was to identify the latest trends in Python and gather insight on how the Python development world looks today. With that motivation in mind, we set out to determine:

- What's the current Python 3 adoption rate?
- How is Python used with other languages?
- What kinds of projects are Python used for?
- What are the major types of development among Python users?
- What are the most popular technologies and tools?
- Which frameworks and libraries are most common, and how do they complement each other?
- Who are Python developers?

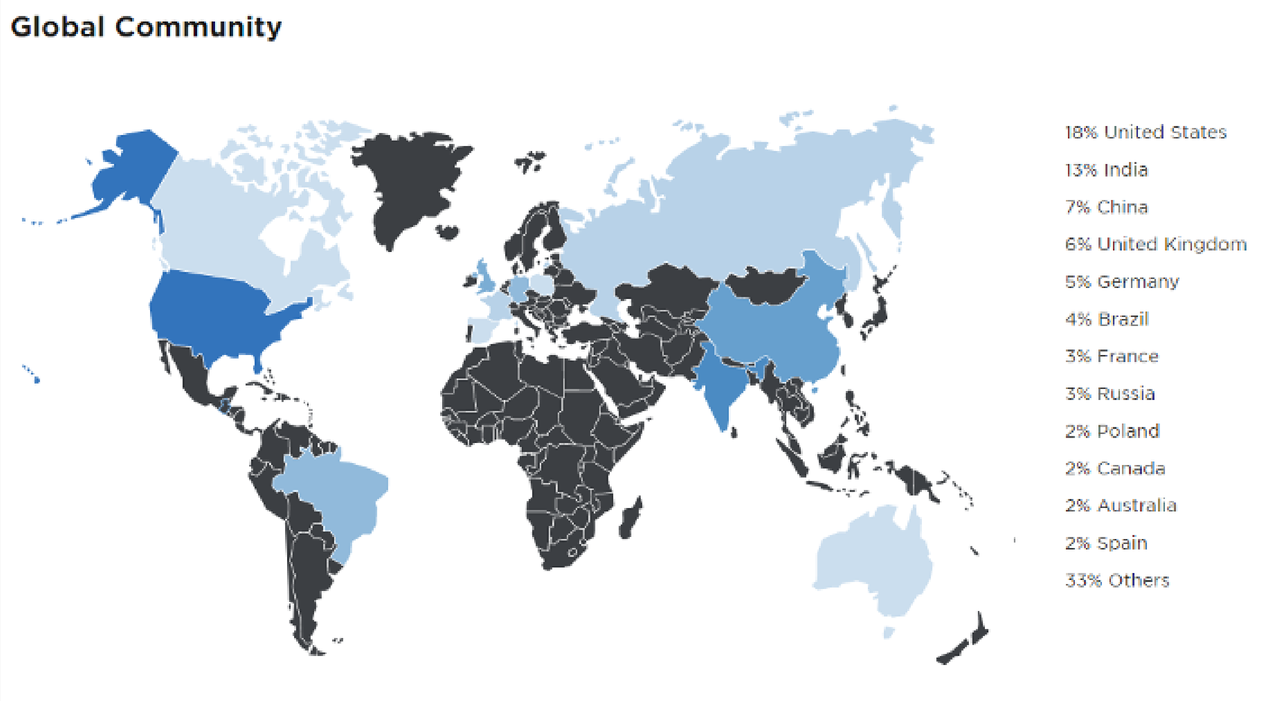
The survey was fielded in October 2017. We'll summarize the results here, and you can dig into more charts and the raw data on the Python Developers Survey 2017 Results website.

Methodology

Before we look at the data and insights, it's important to review the survey's methodology, including how it was distributed and steps taken to eliminate potential bias and ensure it wasn't slanted in favor of any specific tool, technology, library, or country.

We sent the survey to several independent groups, including those subscribed to the PSF mailing lists, blog, Slack, LinkedIn, and Twitter. It was also promoted for several weeks on some of the most visited Python.org pages and publicized via Read the Docs. The majority of responses (62%) came from banners on Python.org; other major sources were the PSF blog and Twitter posts. No product-, service-, or vendor-related channels were used in order to prevent the survey results from slanting in favor of any specific tool or technology.

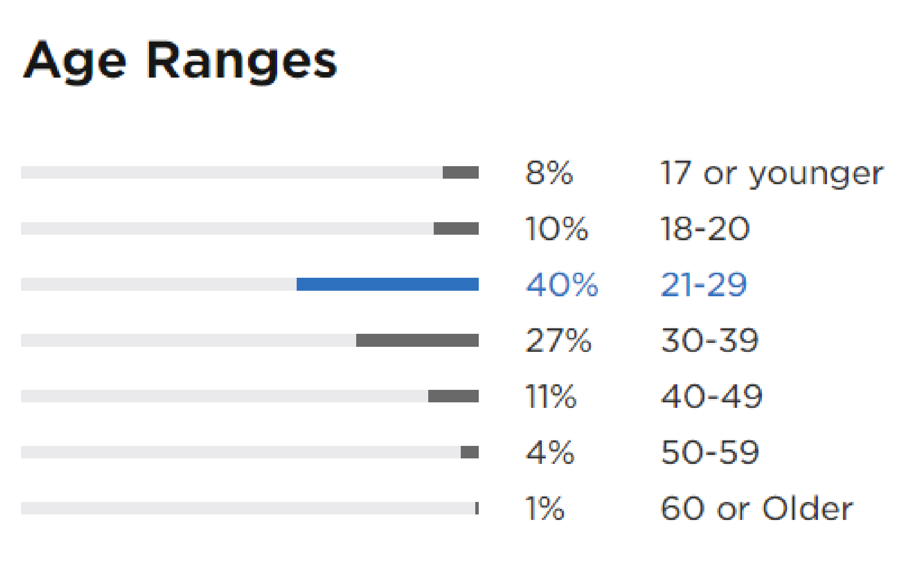
The survey was well received by the community and the response rate was higher than predicted. During the collection period, it received more than 10,000 responses. Partial responses and duplicates were filtered out during the post-survey analysis, leaving 9,532 respondents from 150+ countries included in our analysis dataset. The conservative confidence interval that describes the maximum statistical error for such a large sample size is just 1%, which means all results are statistically significant. The main goal for the survey was achieved: we received precise and reliable data!



All images courtesy of the Python Developers Survey 2017 Results website.

Python developer profile

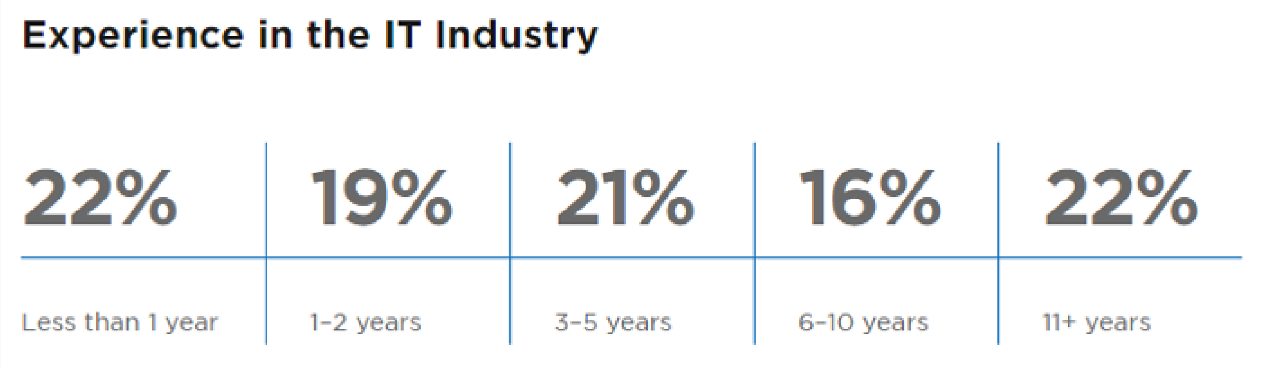
We'll start our journey into the Python world with the most exciting thing about it: The Python community is as diverse as the language and its applications.



Python users range widely in age, but the majority of respondents are in their 20s and a quarter are in their 30s. Interestingly, almost one-fifth of Python users are under age 20. If we compare the Python developers' age ranges with the general developers age ranges identified by Stack Overflow in its latest survey, the distribution looks similar, with the Python survey reporting slightly more developers under the age of 18. This can be explained by the fact that many students use Python in schools and universities, and it's a common first language.

More than half of the respondents work full time as developers, and one in five developers double as data analysts, architects, or team leads. Comparing our data to the Stack Overflow survey, we can see Python developers are less likely than developers in general to be employed full time (52% of Python developers vs. 74% of all developers) and more likely to be self-employed or freelancers (13% of Python developers vs. 9.7% of all developers).

The Python survey respondents report a wide range of experience; 22% have less than one year of experience, the same percentage has more than 11 years of experience, and there's a smooth distribution in the middle. This indicates there's a good balance between newcomers and experienced developers, making Python a sustainable language.

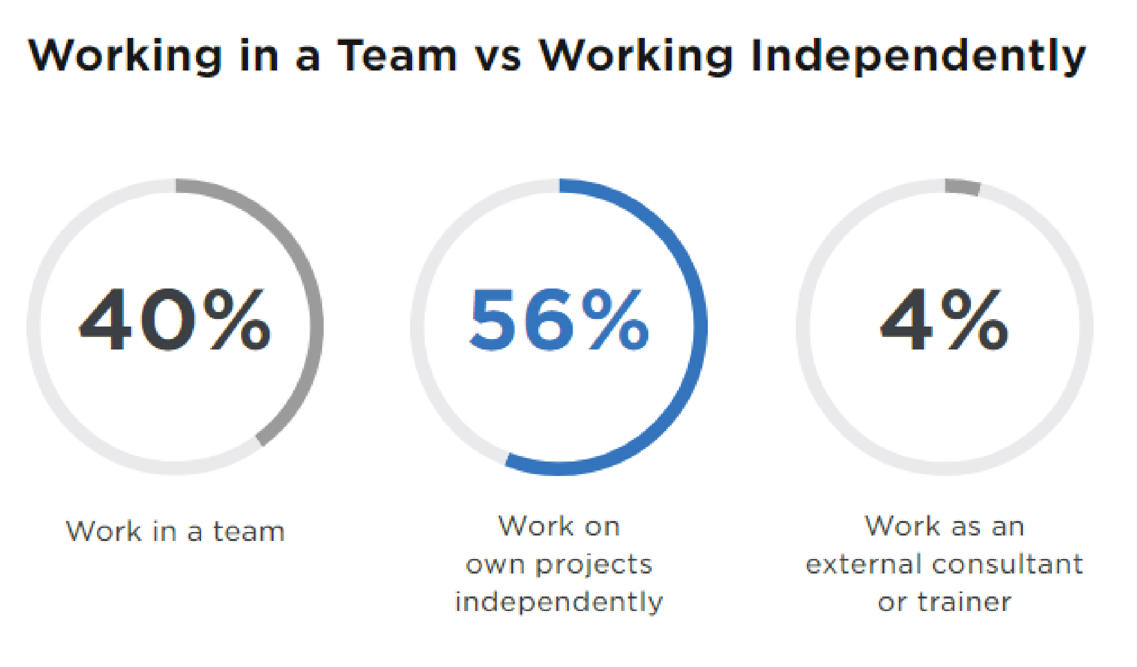


Experience in the IT Industry

22%	19%	21%	16%	22%
Less than 1 year	1-2 years	3-5 years	6-10 years	11+ years

It's notable that Python has a much higher percentage of newcomers than other languages on average. According to Stack Overflow data, 30% of developers have less than two years of professional experience, compared to 41% in the Python developers survey.

Large developer groups are uncommon in Python. In our survey, 56% of Python developers say they work on projects independently and 40% work on a team usually of two to seven people. About half the respondents work on one main project with a handful of side projects, while about a quarter focus on one project at a time.



In 2016, JetBrains hosted a Python developer survey without the PSF's involvement. While JetBrains had a large sample size in 2016, its survey was promoted primarily through its own channels, which naturally attracted a larger share of PyCharm users—about 50% users of PyCharm and 50% of other editors. To avoid inevitable bias, the 2016 survey didn't compare the userbase numbers for different code editors. Even though the 2016 survey results are biased, it is worth comparing some of the 2017 and 2016 survey results. For example, in the 2016 survey, 45% reported working independently on their own projects vs. 51% working on teams. The different ratio in the 2017 survey might be explained by the larger number of newcomers for whom Python is the first language and data scientists who are now on par with the number of web developers using Python.

Python job opportunities

As we wrote above, Python was the most desirable language (i.e., the one survey takers are most interested in learning) in Stack Overflow's 2018 Developer Survey; this is the second consecutive year it's received that ranking. Python users also ranked it third on the list of most loved programming languages. How do these facts relate to the Python job market?

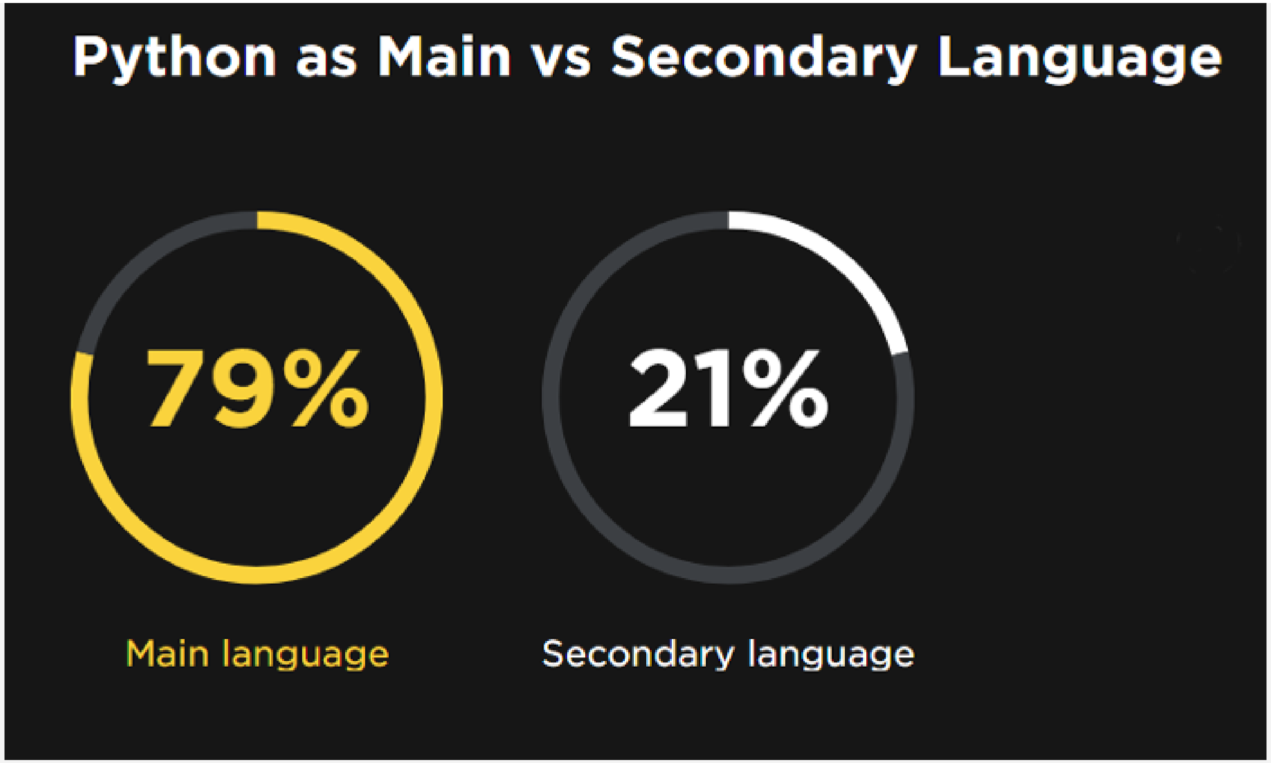
Even though the Python Developer Survey didn't specifically address job popularity, we can gather from other resources that Python developers are in demand for a wide range of jobs all over the world. These include machine learning, databases, data analysis, cloud infrastructure, design, site reliability/testing, web scraping, security, mobile development, APIs, and more. Based on the jobs listed on jobs.python.org, opportunities skew toward international demand, especially in the UK region.

Searching for "Python developer" on any major job site will yield thousands of job opportunities. In March 2018, the average salary (based on Indeed data) of a U.S. Python developer in the U.S. was $115,835. The high average salary indicates that many companies are competing to hire knowledgeable Python developers and further supports the idea that Python is an in-demand skill.

Python also has a stake in the best job in America—data science (according to Glassdoor.com in April 2018). Approximately one in five data science jobs involve Python, including the NumPy, pandas, and Matplotlib libraries.

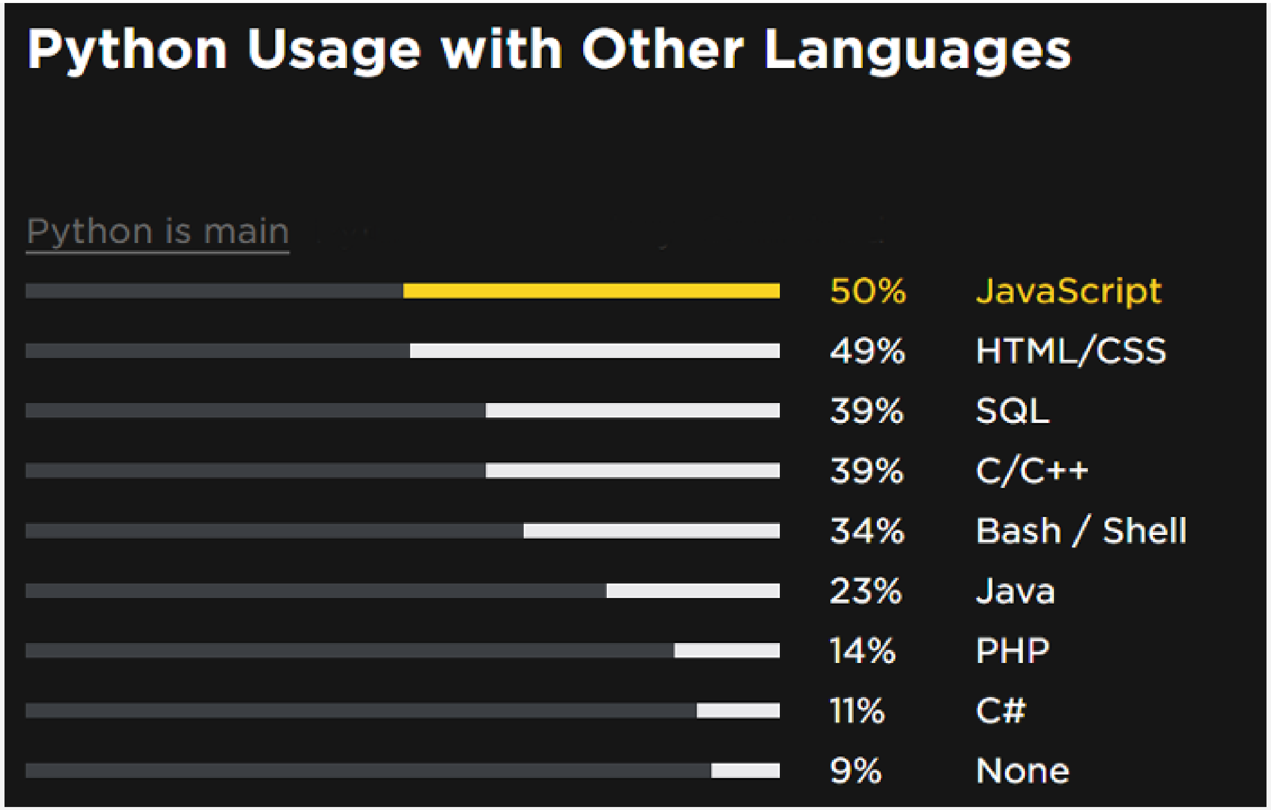
General Python usage

Almost four out of five Python developers say it's their main language, a four percent increase from 75% in JetBrains' 2016 survey.



Various research shows the number of Python developers, as well as the share of developers using Python as their main language, is growing steadily year by year. In Stack Overflow's survey, Python popularity has increased from 32% in early 2017 to 38.8% at the end of the year. This can be explained by the rapidly growing popularity of Python for data science, as this segment of users is growing much faster than the others.

Python is often combined with other languages:

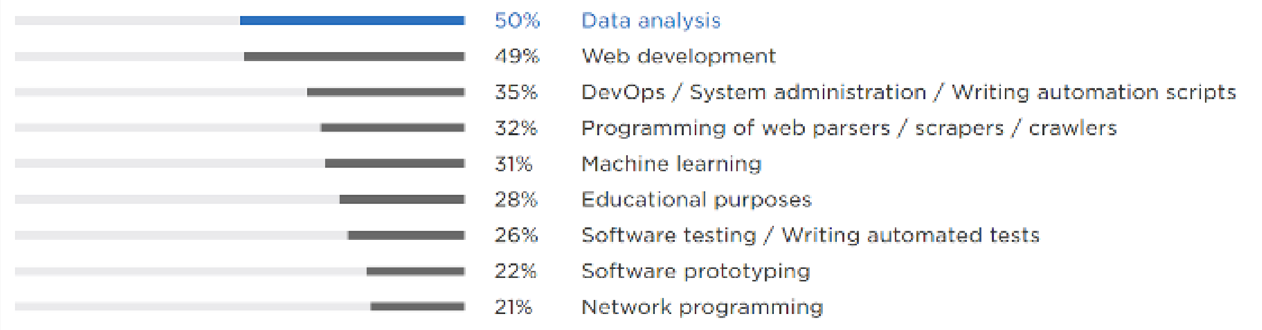


Half of all developers using Python as their main language also use JavaScript. JavaScript is used by 79% of web developers but only 39% of those involved in data analysis or machine learning.

The breakdown is slightly different for those who use Python as their secondary language, as fewer of them use JavaScript (46%) and many more use C/C++ (42%), Java (41%), and C# (24%).

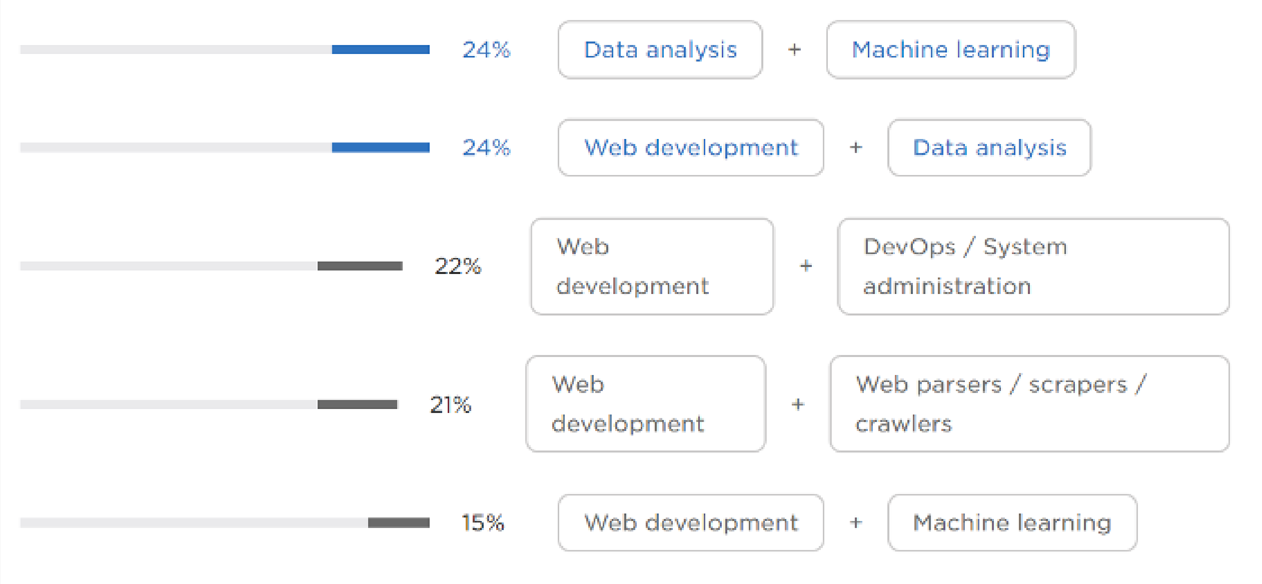
Types of Python development

To identify the most popular types of Python development and their intersections, we asked two similar questions: "What do you use Python for?" to which respondents could select multiple answers, and "What do you use Python for the most?" with only one possible answer. The results showed scientific development is now as popular as web development: half the respondents work on data science projects and half do web development.



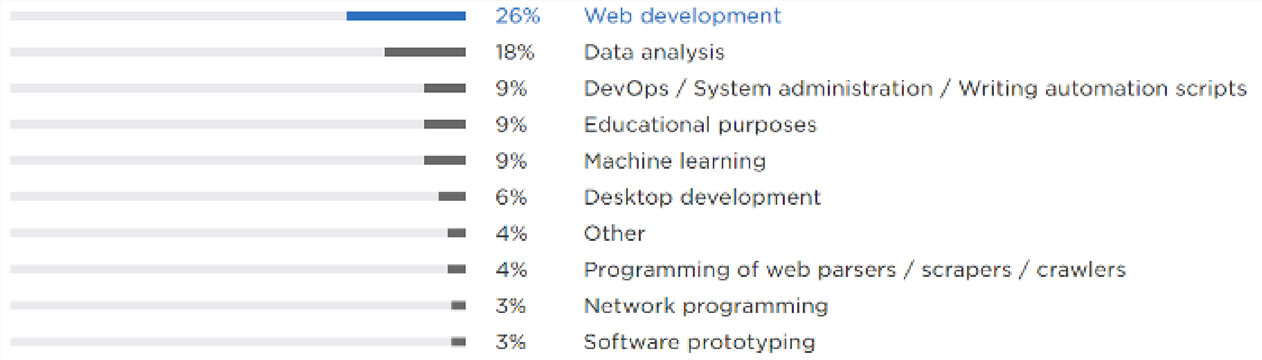
Responses differ depending on whether the respondent uses Python as their main or secondary language: 54% of those who use Python as their main language are involved in web development compared to 33% of those who say Python is their secondary language. The difference is less significant for data analysis, machine learning, and other types of development.

Many Python developers wear multiple hats. The roles they combine most often are:

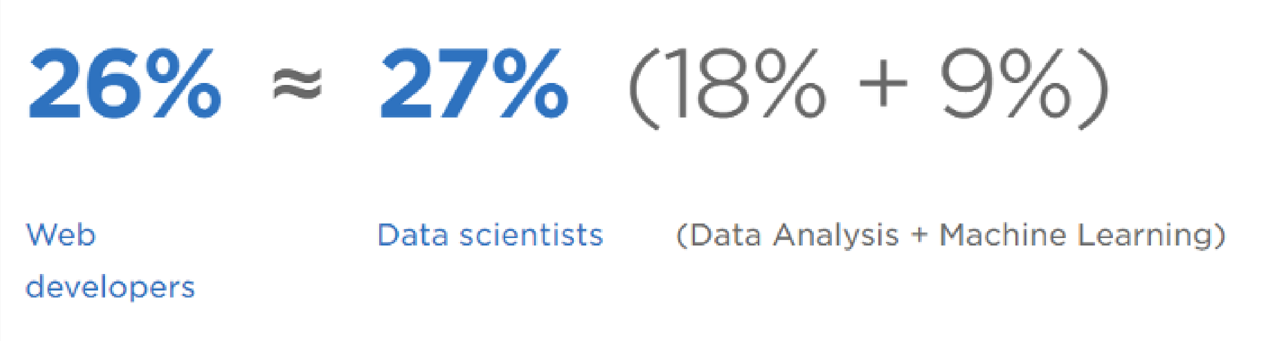


The intersection of data analysis and machine learning was expected, but the overlaps between web development and data analysis/machine learning are noteworthy.

When asked about the primary type of development they do, 26% of Python users say web development, outpacing data analysis (reported by 18%), by a wide margin.



However, if we combine the 18% who do data analysis with the nine percent whose primary role is machine learning, we learn 27% are primarily doing scientific development. That means there are as many web developers as there are data scientists using Python.



26% ≈ 27% (18% + 9%)

Web developers

Data scientists

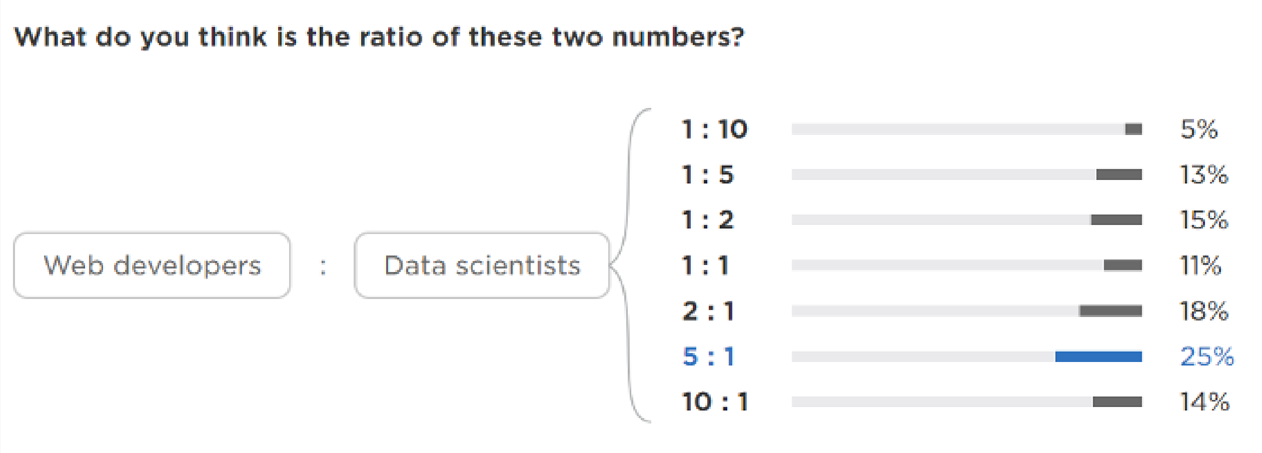
(Data Analysis + Machine Learning)

Interestingly enough, when JetBrains did its Python developers survey in 2016, 38% of respondents identified as web developers and only 21% as scientific developers. That could be evidence of rapid growth of data scientists among Python developers.

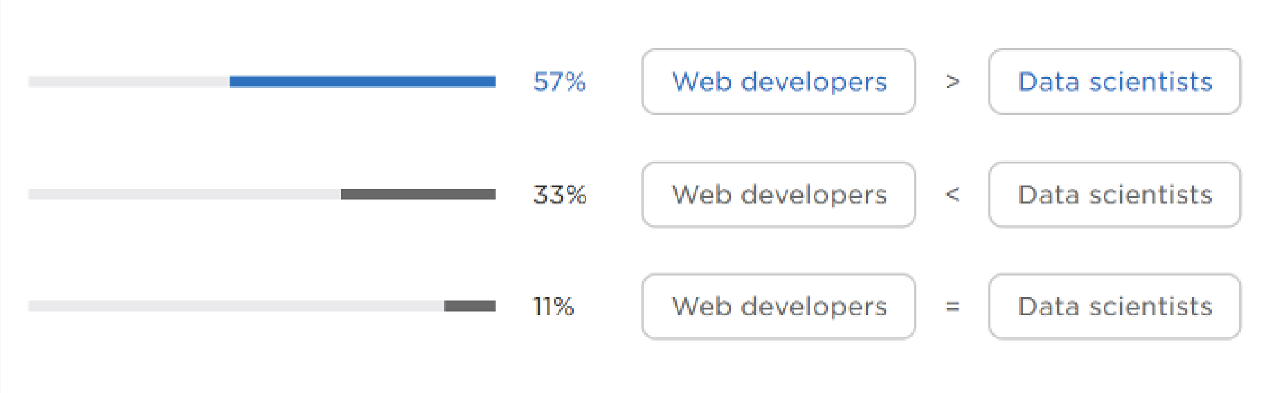
Comparing the trends among those using Python as their main language vs. a secondary language, web development has the biggest gap (29% main vs 15% secondary). The difference is much smaller for data analysis and machine learning. Conversely, more DevOps and system administrators are using Python as a secondary language (13%) than a main language (8%).

Is data science taking Python by storm?

The most intriguing question in the Python developers survey concerned the ratio of web developers to data scientists in the Python world. We asked respondents to estimate the ratio between web developers and data scientists using Python. Respondents could answer based on their own experience, beliefs, and general "gut feeling." Because we explicitly asked about types of development, and the sample size is large enough to be statistically significant, we could compare the community's perception against reality.


What do you think is the ratio of these two numbers?

More respondents (57%) thought web developers were more common than data scientists, while only 33% assumed the opposite.



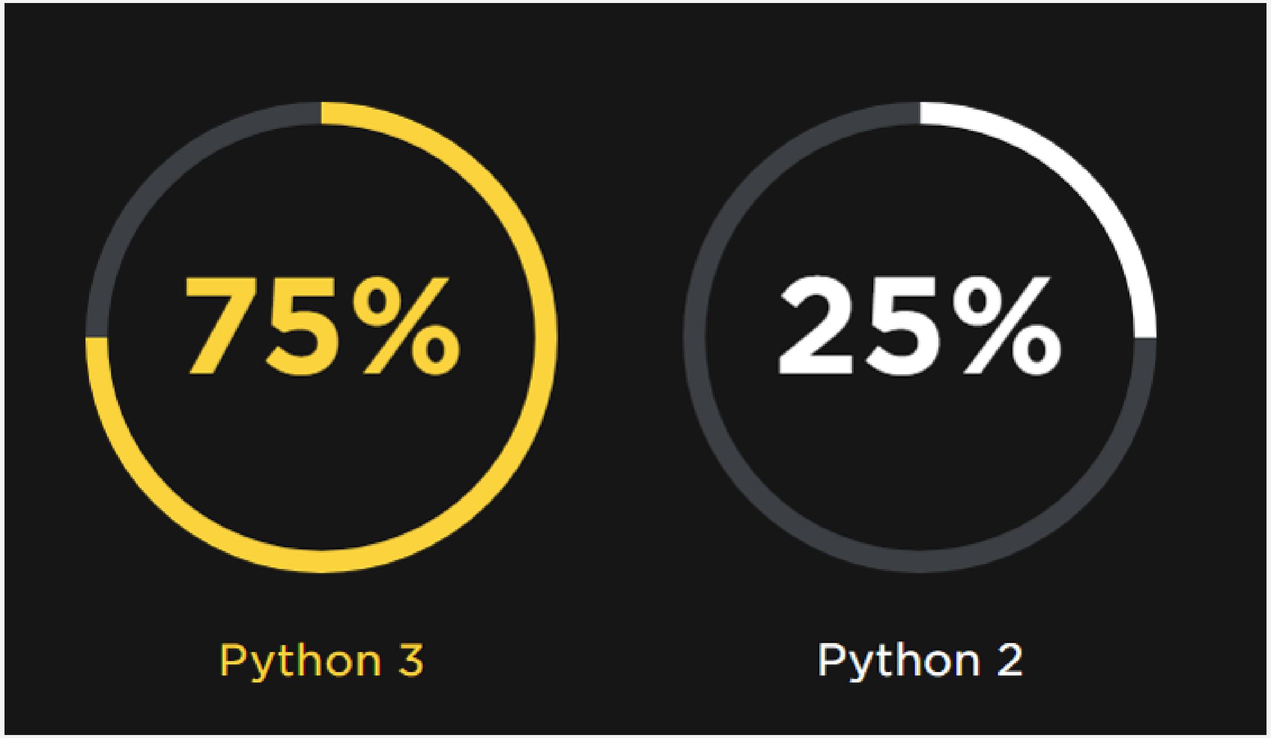
Based on the data reported in the previous section (with 26% reporting web development and 27% reporting scientific development as their primary activity), the true ratio is 1:1.

Only one in nine respondents guessed this correctly; most underestimated the number of data science Python users. Web development is generally perceived as the major application for Python. While this was true a couple of years ago, the number of Python data scientists is growing rapidly and is already on par with the number of web developers.

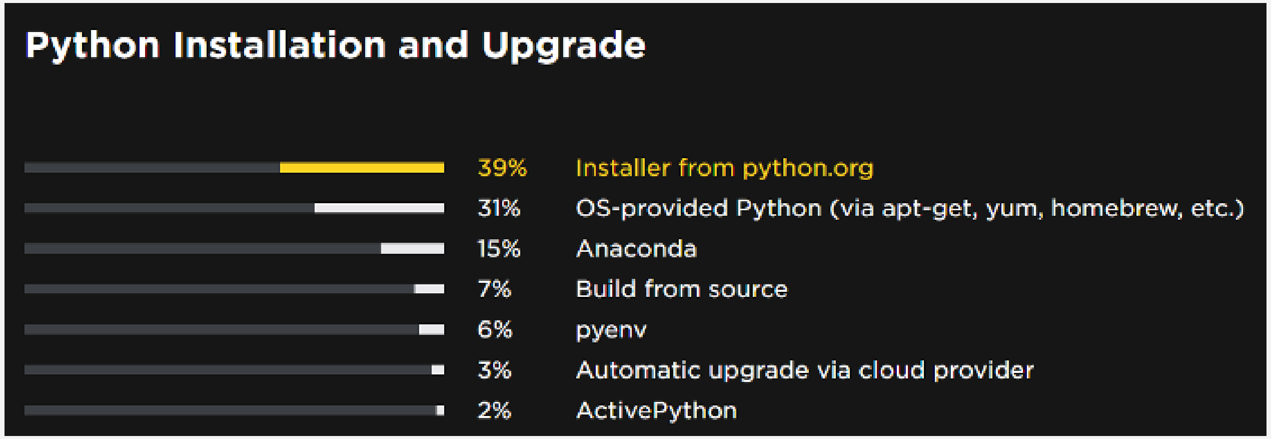
Python 2 vs. Python 3

We asked, "Which version of Python do you use the most?" Python 3 is a strong leader at 75%, and Python 2 is used as the main interpreter by only 25%. Python 3 is growing rapidly; in the 2016 survey, 60% were using Python 2 and 40% Python 3. Python 2 use is declining as it's not actively developed, doesn't get new features, and will no longer be maintained after 2020.

It's notable that 70% of web developers are using Python 3, compared to 77% of data analysts and 83% of machine learning specialists. This may be because many web developers still have to maintain legacy code while transitioning to Python 3, and many data analysts and machine learning specialists have joined the Python ecosystem more recently and went straight to Python 3.



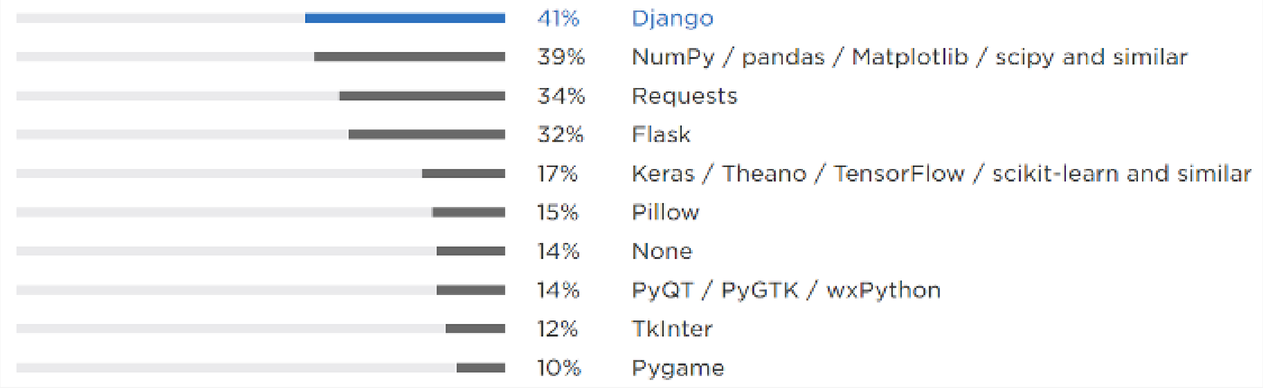
We also asked developers how they install and update their Python installations.



Seventy percent install Python from python.org or with operating system-provided package managers like APT and Homebrew.

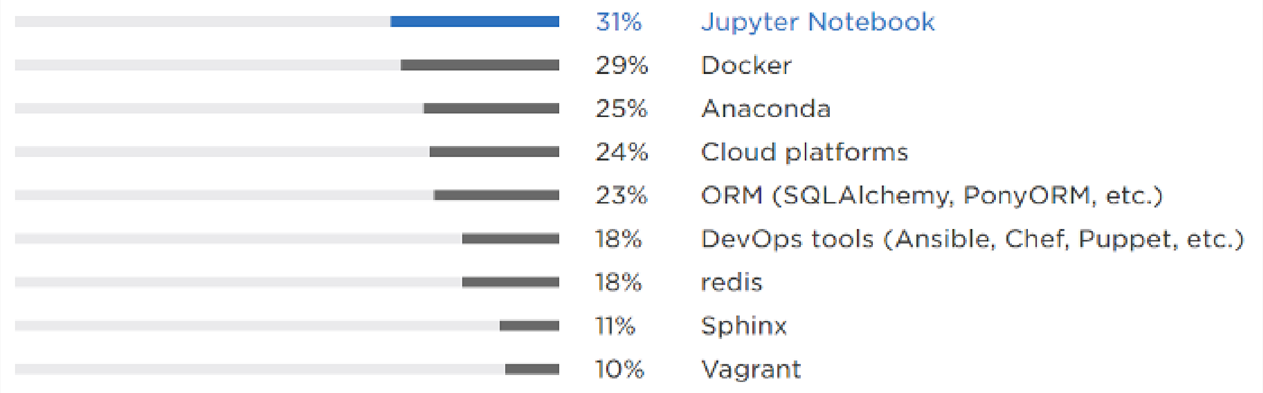
Python frameworks, libraries, and technologies

Django is the most popular framework; it's used by 41% of Python developers.



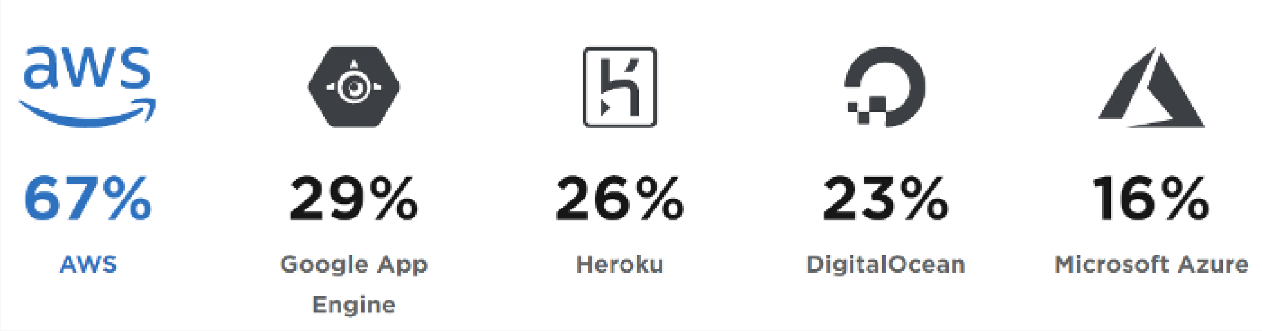
Scientific packages such as NumPy, pandas, and Matplotlib, combined, are in close second at 39%. Other popular frameworks and libraries are Requests, Flask, Keras/Theano/TensorFlow/Scikit-learn, and similar. TensorFlow and Django are on StackOverflow's list of the most loved and wanted technologies.

Django was selected as the top framework by 76% of web developers compared to only 31% of those working in data science. Curiously, 29% of web developers are using scientific libraries. This corroborates a strong overlap between the two roles.



When asked about technologies used in addition to Python, Jupyter Notebook ranked highest at 31%, which makes sense given the large number of developers involved in data science. Docker trails by only two percent at 29%. Breaking this down by roles, Docker is used by 47% of those mostly involved in web development and only 23% of data scientists. Similarly, cloud platforms such as Amazon Web Services, Google App Engine, Heroku, and others are used twice as often by web developers than by data scientists.

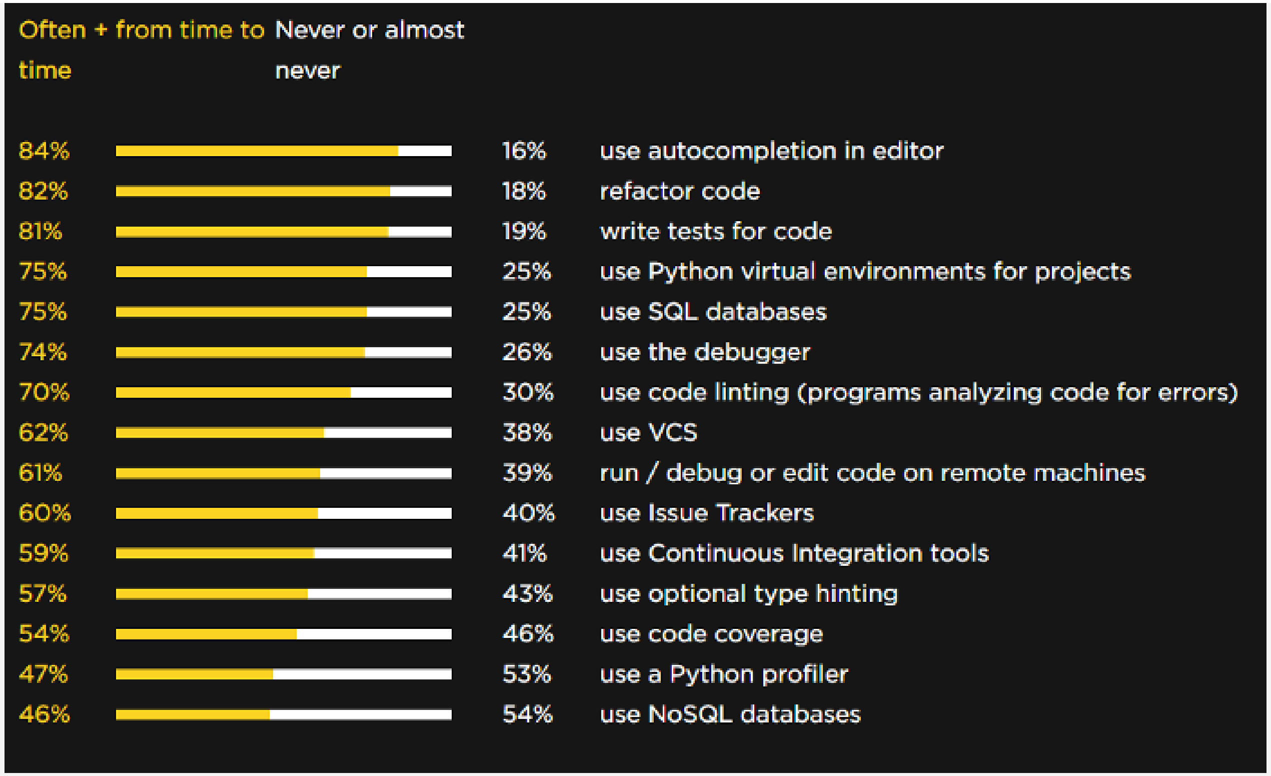
Comparing 2016's survey results with 2017's, Django and Flask, the two most popular web frameworks, have lost share. (Django was used by 51% in 2016 and 41% in 2017; Flask by 40% in 2016 and 32% in 2017). At the same time, use of scientific library and technology frameworks have grown. In 2016, Anaconda, NumPy, and Matplotlib (combined) were used by 36%; at the end of 2017, NumPy, pandas, Matplotlib, SciPy, and similar were used by 39%, and Anaconda was used by 25%.



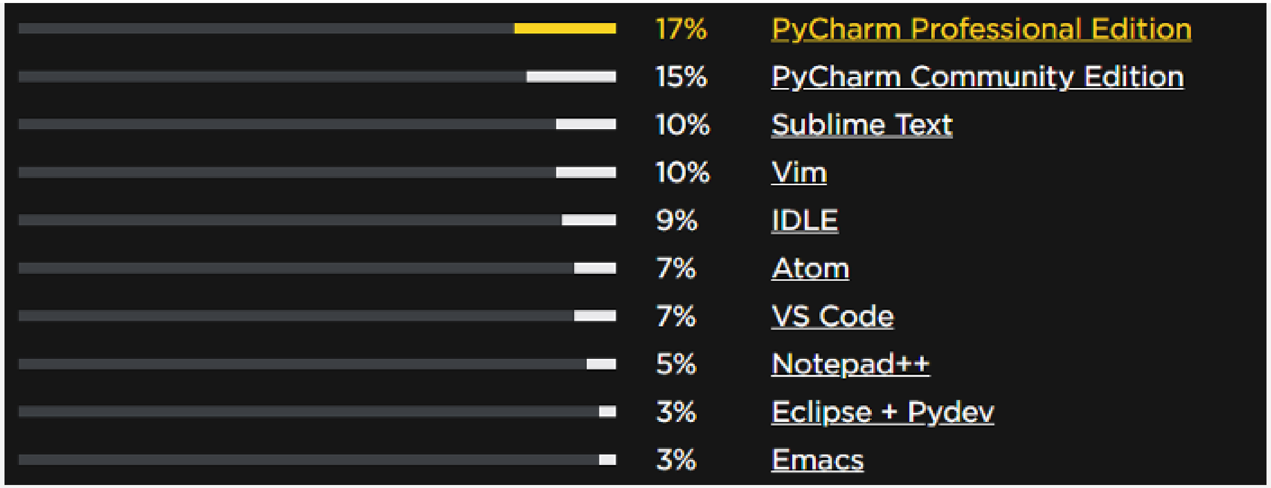
When asked which cloud platforms they use, cloud-using respondents ranked AWS first at 67%. Google App Engine, Heroku, and DigitalOcean are used far less often. AWS also rates high on the list of most loved and wanted platforms in Stack Overflow's survey. We were limited on the number of cloud platforms we could list as potential answers; the "other" category collected 13% of responses, including Linode, PythonAnywhere, OpenShift, and OpenStack.

Tools and features for Python development

When we asked about the use of development practices, tools, and features in Python development, the top spots were occupied by code autocompletion, code refactoring, writing unit tests, and using virtual environments for Python projects. NoSQL databases, Python profilers, and code coverage tools were among the most rarely used. These results are very similar to 2016's data:



To identify the most popular editors and IDEs, we asked two questions: "What editors/IDEs have you considered for use in your Python development?" with multiple answers allowed, and a single-answer question, "What is the main editor you use for your current Python development?" Based on more than 8,000 replies to this question, PyCharm is the most popular tool, followed by Sublime, Vim, IDLE, Atom, and VS Code.



Note: We took a number of steps to eliminate bias and ensure the survey was not slanted in favor of any specific tool. To learn more about the survey methodology and the channels used to distribute the survey, please refer to the raw data section of the survey results website.

Web developers have slightly different editor preferences from data scientists. Web developers highly prefer PyCharm Professional Edition, Sublime text, and Vim, while data scientists clearly prefer PyCharm Community Edition, Jupyter Notebook, and Spyder.

When we asked, "What editor(s)/IDE(s) have you considered for use in your Python development?" we learned web developers most often consider Sublime Text (47%), followed by Vim (39%) and Atom (32%). Scientific developers most often consider Jupyter Notebook (42%), followed by PyCharm Community Edition (39%), Sublime Text (31%), and Vim (26%).

We also found that most developers use their editor daily, and about one in five use their editor weekly.

Survey raw data

Consistent with the open source philosophy of Python and its community, we have made the raw data available to the public and we welcome additional analyses and conclusions. We intend to repeat the survey in 2018 and future years. Our goal is to keep the survey similar year to year so longitudinal data analysis can be done.

Before dissecting the raw data, please note the following: data are anonymized, with no personal information or geolocation details. Moreover, all open-ended fields have been pruned to prevent identification of any individual respondent by their verbatim comments. To help others better understand the logic of the survey, we are sharing the dataset, the survey questions, and all the survey logic in English. We used different ordering methods for answer options (alphabetic, randomize, direct). The order in which the answer options were used is specified in each question.

We'll be glad to learn about your findings! Please share them on Twitter or other social media mentioning @jetbrains and @ThePSF with the #pythondevsurvey2017 hashtag. We're also open to suggestions and feedback so we can improve the survey next time. Feel free to open issues here with any comments or questions.

Key takeaways

The 2017 Python developer survey was a benchmark for the community. Some of the more important takeaways include:

- The adoption rate of Python 3 is already at 75% and quickly growing.
- Currently there are as many Python developers working in data science as in web development, but the rapid growth of Python among data scientists indicates parity may change quickly.
- For four out of five developers using Python, it is their main language.

- Django, NumPy, pandas, and Matplotlib are the most popular frameworks and libraries used by Python developers. Jupyter Notebook and Docker are the most popular technologies used with Python. AWS is the most popular cloud platform.
- PyCharm, Sublime, VIM, Atom, and VS Code are the most commonly used Python editors.
- Half of those who use Python as their primary language also use JavaScript and HTML/CSS. Python is also used often with SQL, Bash/Shell, C/C++, and Java.

We hope the survey findings will clarify the current state of the Python developer community, see the big picture, and answer some questions.

To learn more about Python trends and what employers are looking for, attend the authors' talk By the Numbers: Python Community Trends in 2017/2018r at PyCon Cleveland 2018.

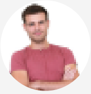
Topics : Python PyCon Programming

About the author

Ewa Jodlowska - Ewa is the Executive Director of the Python Software Foundation.

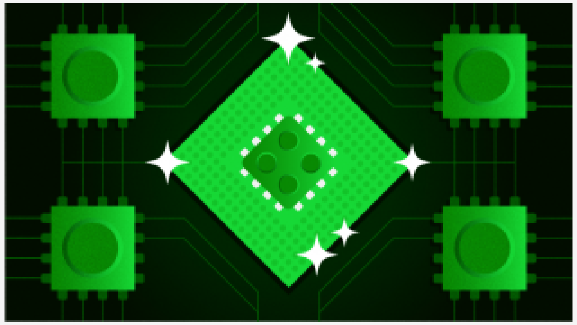
• More about me

Contributors

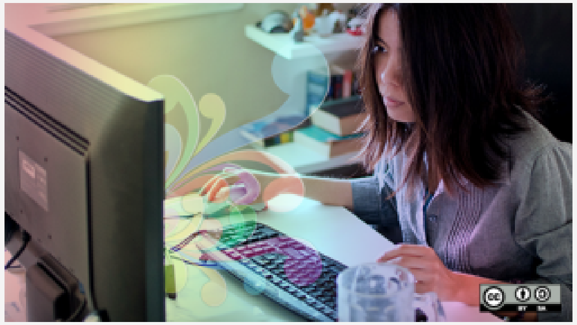


Dmitry Filippov

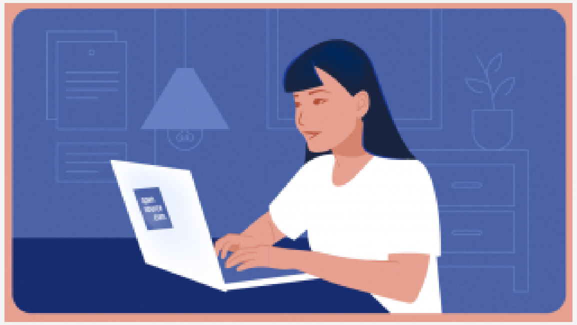
Recommended reading



Program IoT systems using Python with this VSCode plugin for RTOS

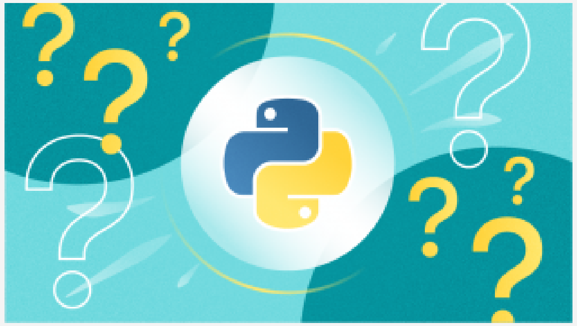


10 ReactJS tools to boost your web development skills



Use intent parsers for your open source home automation project

How to prepare to write your first Mycroft AI skill using Python



4 essential tools to set up your Python environment for success

My favorite 10 Node.js projects

Share your sysadmin tricks

Send us home-grown sysadmin scripts. Submit an article proposal today.

Learn more

opensource.com