**What can Python be used for besides data science?**

The better question is what can't it be used for? Here are some key places where you may see Python:

* **Web Development** – Developers, engineers, and data scientists use Python for web scraping or creating a mock-up an app.
* **Automating Reports**– Analysts or product managers who need to make the same [**Excel**](https://www.coursereport.com/blog/how-to-use-microsoft-excel-like-a-data-analyst) report every single week can use Python to help create reports and save time.
* **Finance and Business**– Used for reporting, predictive models, and academic research.
* **Simulations**– As a postdoctoral fellow at Ohio State University, my colleagues used Python to create simulations to study various different behaviors with a computer.
* **Why do you think Python has recently overtaken R in popularity among data scientists?**
* There are a couple of reasons I think Python has taken off. Python is a general purpose language, used by data scientists and developers, which makes it easy to collaborate across your organization through its simple syntax. People choose to use Python so that they can communicate with other people. The other reason is rooted in academic research and statistical models. I would say that R has better statistical packages than Python, but Python has deep learning, structured ways to do machine learning, and can deal with larger amounts of data. As people shift more to deep learning, the bias has been shifting toward Python.
* **What is Data Science?**
* Data science is the field of study that combines domain expertise, programming skills, and knowledge of mathematics and statistics to extract meaningful insights from data. Data science practitioners apply [machine learning](https://www.datarobot.com/wiki/machine-learning/) [algorithms](https://www.datarobot.com/wiki/algorithm/) to numbers, text, images, video, audio, and more to produce [artificial intelligence (AI)](https://www.datarobot.com/wiki/artificial-intelligence/) systems to perform tasks that ordinarily require human intelligence. In turn, these systems generate [insights](https://www.datarobot.com/wiki/insights/) which analysts and business users can translate into tangible business value.

## Why Data Science is Important?

More and more companies are coming to realize the importance of data science, AI, and machine learning. Regardless of industry or size, organizations that wish to remain competitive in the age of [big data](https://www.datarobot.com/wiki/big-data/) need to efficiently develop and implement data science capabilities or risk being left behind.

## Why Learn Python For Data Science?

Before we explore how to learn Python for data science, we should briefly answer why you should learn Python in the first place.

In short, understanding Python is one of the valuable skills needed for a data science career.

Though it hasn’t always been, Python is the programming language of choice for data science. Here’s a brief history:

* In 2016, it [overtook R on Kaggle](https://blog.kaggle.com/2017/01/05/your-year-on-kaggle-most-memorable-community-stats-from-2016/), the premier platform for data science competitions.
* In 2017, it [overtook R on KDNuggets’s annual poll](https://www.kdnuggets.com/2017/08/python-overtakes-r-leader-analytics-data-science.html) of data scientists’ most used tools.
* In 2018, 66% of data scientists reported [using Python daily](https://www.kdnuggets.com/2018/05/poll-tools-analytics-data-science-machine-learning-results.html), making it the number one language for analytics professionals.

Data science experts expect this trend to continue with increasing development in the Python ecosystem. And while your journey to learn Python programming may be just beginning, it’s nice to know that employment opportunities are abundant (and growing) as well.

According to Indeed, the [average salary for a Data Scientist](https://www.indeed.com/career/data-scientist/salaries) is $121,583.

The good news? That number is only expected to increase, as demand for data scientists [is expected to keep growing](https://quanthub.com/data-scientist-shortage-2020/). In 2020, there are **three times** as many job postings in data science as job searches for data science, according to Quanthub. That means the demand for data scientitsts is vastly outstripping the supply.

So, the future is bright for data science, and Python is just one piece of the proverbial pie. Fortunately, learning Python and other programming fundamentals is as attainable as ever. We’ll show you how in five simple steps.

But remember – just because the steps are simple doesn’t mean you won’t have to put in the work. If you apply yourself and dedicate meaningful time to learning Python, you have the potential to not only pick up a new skill, but potentially bring your career to a new level.

## How to Learn Python for Data Science

#### Click to View Our How to Learn Python Infographic

First, you’ll want to find the right course to help you learn Python programming. [Dataquest’s courses](https://www.dataquest.io/) are specifically designed for you to learn Python for data science at your own pace, challenging you to write real code and use real data in our interactive, in-browser interface.

In addition to learning Python in a course setting, your journey to becoming a data scientist should also include soft skills. Plus, there are some complimentary technical skills we recommend you learn along the way.

### **Step 1: Learn Python Fundamentals**

Everyone starts somewhere. This first step is where you’ll learn Python programming basics. You’ll also want an introduction to data science.

One of the important tools you should start using early in your journey is [Jupyter Notebook](https://www.dataquest.io/blog/jupyter-notebook-tips-tricks-shortcuts/), which comes prepackaged with Python libraries to help you learn these two things.

**Kickstart your learning by**: Joining a community

By joining a community, you’ll put yourself around like-minded people and increase your opportunities for employment. According to the Society for Human Resource Management, employee referrals account for 30% of all hires.

Create a [Kaggle](https://www.kaggle.com/) account, join a local Meetup group, and participate in [Dataquest’s learner community](https://community.dataquest.io/) with current students and alums.

**Related skills**: Try the Command Line Interface

The [Command Line Interface (CLI) lets you run scripts more quickly](https://www.dataquest.io/blog/data-cleaning-command-line/), allowing you to test programs faster and work with more data.

### **Step 2: Practice Mini Python Projects**

We truly believe in hands-on learning. You may be surprised by how soon you’ll be ready to [build small Python projects](https://www.dataquest.io/blog/top-20-python-ai-and-machine-learning-open-source-projects/). We've already put together a great [guide to Python projects for beginners](https://www.dataquest.io/blog/python-projects-for-beginners/), which includes ideas like:

* [Tracking and Analyzing Your Personal Amazon.com Spending Habits](https://www.dataquest.io/blog/how-much-spent-amazon-data-analysis/) — A fun project that'll help you practice Python and pandas basics while also giving you some real insight into your personal finance.
* [Analyze Data from a Survey](https://www.dataquest.io/blog/how-to-analyze-survey-data-python-beginner/) — Find public survey data or use survey data from your own work in this beginner project that'll teach you to drill down into answers to mine insights.
* Try one of our [Guided Projects](https://www.dataquest.io/data-science-projects/) — Interactive Python projects for every skill level that use real data and offer guidance while still challenging you to apply your skills in new ways.

But that's just the tip of the iceberg, really. You can try programming things like calculators for an online game, or a program that fetches the weather from Google in your city. You can also build simple games and apps to help you familiarize yourself with working with Python.

Building mini projects like these will help you learn Python. programming projects like these are standard for all languages, and a great way to solidify your understanding of the basics.

You should start to build your experience with APIs and begin [web scraping](https://www.dataquest.io/blog/web-scraping-tutorial-python/). Beyond helping you learn Python programming, web scraping will be useful for you in gathering data later.

**Kickstart your learning by**: Reading

Enhance your coursework and find answers to the Python programming challenges you encounter. Read guidebooks, blog posts, and even other people’s open source code to learn Python and data science best practices – and get new ideas.

[*Automate The Boring Stuff With Python*](https://automatetheboringstuff.com/) by Al Sweigart is an excellent and entertaining resource. But we've put together an entire list of [data science ebooks that are totally free](https://www.dataquest.io/blog/free-books-learn-data-science/) for you to check out, too. Highlights include:

* [The Data Science Handbook](https://www.thedatasciencehandbook.com/) — A great collection of interviews with working data scientists that'll give you a better idea of what real data science work is like and how you can succeed in the field.
* [Python Data Science Handbook](https://jakevdp.github.io/PythonDataScienceHandbook/) — A helfpul guide that's also available in convenient [Jupyter Notebook format on Github](https://github.com/jakevdp/PythonDataScienceHandbook" \t "_blank) so you can dive in and run all the sample code for yourself.
* [Elements of Statistical Learning](https://web.stanford.edu/~hastie/ElemStatLearn/) — A massive and recently-updated statisics textbook that can serve as a great reference as you're learning Python to make sure your work is statistically valid.

**Related skills**: Work with databases using SQL

SQL is used to talk to databases to alter, edit, and reorganize information. SQL is a staple in the data science community, and we've written a whole article about[why you need to learn SQL if you want a job in data](https://www.dataquest.io/blog/why-sql-is-the-most-important-language-to-learn/).

### **Step 3: Learn Python Data Science Libraries**

Unlike some other programming languages, in Python, there is generally a best way of doing something. The three best and most important Python libraries for data science are NumPy, Pandas, and Matplotlib.

We've put together a helpful guide to the [15 most important Python libraries for data science](https://www.dataquest.io/blog/15-python-libraries-for-data-science/), but here are a few that are really critical for any data work in Python:

* NumPy —  A library that makes a variety of mathematical and statistical operations easier; it is also the basis for many features of the pandas library.
* pandas — A Python library created specifically to facilitate working with data, this is the bread and butter of a lot of Python data science work.
* Matplotlib — A visualization library that makes it quick and easy to generate charts from your data.
* scikit-learn — The most popular library for machine learning work in Python.

NumPy and Pandas are great for exploring and playing with data. Matplotlib is a data visualization library that makes graphs like you’d find in Excel or Google Sheets.

**Kickstart your learning by**: Asking questions

You don’t know what you don’t know!

Python has a rich community of experts who are eager to help you learn Python. Resources like Quora, Stack Overflow, and [Dataquest’s learner community](https://community.dataquest.io/) are full of people excited to share their knowledge and help you learn Python programming. We also have an FAQ for each mission to help with questions you encounter throughout your programming courses with Dataquest.

**Related skills**: Use Git for version control

Git is a popular tool that helps you keep track of changes made to your code, which makes it much easier to correct mistakes, experiment, and collaborate with others.

### **Step 4: Build a Data Science Portfolio as you Learn Python**

For aspiring data scientists, [a portfolio is a must](https://www.dataquest.io/blog/build-a-data-science-portfolio/).

These projects should include work with several different datasets and should leave readers with interesting insights that you’ve gleaned. Some types of projects to consider:

* Data Cleaning Project — Any project that involves dirty or "unstructured" data that you clean up and analyze will impress potential employers, since most real-world data is going to require cleaning.
* Data Visualization Project — Making attractive, easy-to-read visualizations is both a programming and a design challenge, but if you can do it right, your analysis will be considerably more impactful. Having great-looking charts in a project will make your portfolio stand out.
* Machine Learning Project — If you aspire to work as a data scientist, you definitely will need a project that shows off your ML chops (and you may want a few different machine learning projects, with each focused on your use of [a different popular algorithm](https://www.dataquest.io/blog/top-10-machine-learning-algorithms-for-beginners/)).

Your analysis should be presented clearly and visually; ideally in a format like a Jupyter Notebook so that technical folks can read your code, but non-technical people can also follow along with your charts and written explanations.

Your portfolio doesn’t necessarily need a particular theme. Find datasets that interest you, then come up with a way to put them together. However, if you aspire to work at a particular company or industry, showcasing projects relevant to that industry in your portfolio is a good idea.

Displaying projects like these gives fellow data scientists an opportunity to potentially collaborate with you, and shows future employers that you’ve truly taken the time to learn Python and other important programming skills.

One of the nice things about data science is that your portfolio doubles as a resume while highlighting the skills you’ve learned, like Python programming.

**Kickstart your learning by**: Communicating, collaborating, and focusing on technical competence

During this time, you’ll want to make sure you’re cultivating those soft skills required to work with others, making sure you really understand the inner workings of the tools you’re using.

**Related skills**: Learn beginner and intermediate statistics

While learning Python for data science, you’ll also want to get a solid background in statistics. Understanding statistics will give you the mindset you need to focus on the right things, so you’ll find valuable insights (and real solutions) rather than just executing code.

### **Step 5: Apply Advanced Data Science Techniques**

Finally, aim to sharpen your skills. Your data science journey will be full of constant learning, but there are advanced courses you can complete to ensure you’ve covered all the bases.

You’ll want to be comfortable with regression, classification, and k-means clustering models. You can also step into machine learning – bootstrapping models and creating neural networks using scikit-learn.

At this point, programming projects can include creating models using live data feeds. Machine learning models of this kind adjust their predictions over time.

**Remember to**: Keep learning!

Data science is an ever-growing field that spans numerous industries.

At the rate that demand is increasing, there are exponential opportunities to learn. Continue reading, collaborating, and conversing with others, and you’re sure to maintain interest and a competitive edge over time.

### **Python Libraries**

It's hard to talk about Python without talking about libraries. A library is a collection of saved code that someone else has written for you. You can import various bits of code so that you don't have to do everything on your own!

**A few libraries that are perfect for beginners:**

* **Random** – This is used to generate random numbers, which can be interesting. You could build your own game using this.
* **Math** – This one gives you access to all kinds of math functions like square root, cos, sine, and more.
* **Collections** – This will help you interface with your computer or collections, which gives you actual access to additional data structure types within Python.

Once you have a handle on the fundamentals, our Metis bootcamp students learn:

* **Pandas** – For data wrangling and data manipulation because it allows a user to read data in, change it, look for missing values, read data out.
* **NumPy** – For fast computation because it speeds up all of the different calculations that you're doing. Pandas actually uses NumPy under the hood for some of its calculations!
* **Scikit-Learn** – For machine learning because it has all of the algorithms you'll want to use for regression, classification, and unsupervised learning. When you’re deep in the Immersive Data Science Bootcamp, you’ll be leveraging Scikit-Learn pretty heavily.
* **Matplotlib and Seaborn**– For data visualizations. The most common ones will both be able to help you produce some nice visuals.

### **Python with Jupyter Notebook**

**Jupyter Notebook is an Integrated Development Environment (IDE)**, and it’s critical in the learning space for two reasons:

* It helps you understand what your code is doing instantaneously. You'll be writing small blocks of code in cells and then executing that code immediately. This gives you instant feedback and shows you errors in your code, shows which functions you might need to change, and more. It allows you to learn more quickly and experiment more conveniently.
* You can also write in Jupyter Notebooks with text. You can include a message to yourself and you can even add images! This function is helpful for organizing your thoughts, remembering what you need to fix or change later, making a note about what a certain code block is doing, and recording steps you’re trying to follow. As an instructor, I can include an image of a code block for my students.

Jupyter Notebook is great for building projects, structuring homework, and collaborative projects. The annotation feature is amazing because students can record their thought process and you can use this in a real-world work environment too!

### **Python Resources for Complete Beginners**

Metis offers a [**Python for Beginners**](https://www.thisismetis.com/courses/python-for-beginners?utm_source=coursereport&utm_medium=sponsoredpost&utm_content=2020q3pythonfords) course and it’s written for people who have never seen Python before. The course starts with, “What is Python?” and then goes through the various different fundamentals in more depth. Metis also offers a free Intro to Python video from their Demystifying Data Science 2019 conference.

Once you've got the Python fundamentals down, try Metis’ [**Beginner Python & Math for Data Science**](https://www.thisismetis.com/courses/beginner-python-and-math-for-data-science?utm_source=coursereport&utm_medium=sponsoredpost&utm_content=2020q3pythonfords) course. This is a great course for people who are serious about a data science career but not quite ready to take the bootcamp. This course will help you brush up on both Python and math as it pertains to data science. From there, you can enroll in Metis’ Immersive [**Data Science Bootcamp**](https://www.thisismetis.com/data-science-bootcamps?utm_source=coursereport&utm_medium=sponsoredpost&utm_content=2020q3pythonfords), which covers machine learning and the visualizations.

In order to understand Python, you have practice. The more you practice, the better you're going to get! Two Python practice learning resources are:

* [**Check.io**](https://checkio.org/?utm_source=coursereport&utm_medium=blogpost), which is a gamified way to learn Python. You'll be completing challenges and progressing along this game board.
* [**Coding Bat**](https://codingbat.com/python?utm_source=coursereport&utm_medium=blogpost), which has a ton of different practice problems. If you are looking to practice, practice, practice, that's another great spot.

I’ve also launched a [**YouTube series**](https://www.youtube.com/channel/UCirb0k3PnuQnRjh8tTJHJuA?utm_source=coursereport&utm_medium=blogpost)! My new series is an intro to [**Seaborn**](https://seaborn.pydata.org/?utm_source=coursereport&utm_medium=blogpost), which is a visualization package. If you are at the level where you know a little bit of Python and are ready to start visualizing data, that could be a helpful resource as well.

**If you complete Metis’ Python for Beginners course would you be ready to apply to Metis’ Data Science Bootcamp?**

The Python for Beginners course will really launch your journey because it gets you comfortable with programming in general. Then, we cover data types, which you have to have down before you can move on. Next, we go through each of those three core foundations: the conditionals, the loops, and the functions.

**What is your advice for a complete beginner who is beginning to learn Python?**

Sometimes beginners can get frustrated because they want to automatically be excellent at Python. It's going to take practice. It's about every single day getting a little bit better. Maybe you're not solving everything from the beginning, but know that you are getting incrementally better. As long as you're willing to put in the work to get a little bit better every day, you're going to be off to a great start with Python.