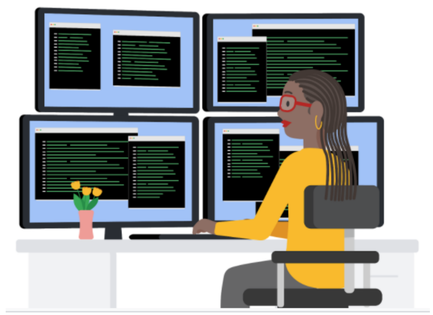
Ways to learn about programming

Writing programming language code can be an exciting and rewarding experience. The programming field has a long history of people helping each other improve their skills and develop best practices. You will focus on the R programming language in this course, but in the future you might choose to pursue additional programming languages based on your interests and professional goals. This reading is a general guide to help you decide which programming languages are best suited for you.



Popular programming languages by profession

Let's go through some potential job titles you might encounter and the most popular programming languages used in those professions. Also included is a list of additional resources for you to explore and learn more about each of the programming languages introduced.

Data analyst

A data analyst collects, transforms, and organizes data to draw conclusions, make predictions, and drive informed decision-making. The most popular programming languages used by data analysts are R and Python.

R offers convenient statistical features for data analysis and is useful for creating advanced data visualizations. Check out these resources to learn more about R:

- The R Project for Statistical Computing: a website for downloading R, documentation, and help
- R Manuals: links to manuals from the R core team, including introduction, administration, and help
- Coding Club R Tutorials: a collection of coding tutorials for R
- R for Beginners: a starting guide to help you work with data, graphics, and statistics in R

Python is a general-purpose language that you can use to create what you need for data analysis. Here are a few resources to begin learning Python:

- The Python Software Foundation (PSF): a website with guides to help you get started as a beginner
- Python Tutorial: a Python 3 tutorial from the PSF site
- Coding Club Python Tutorials: a collection of coding tutorials for Python

Kaggle is an online repository of various datasets that can be used in both R and Python. It's a robust platform that regularly hosts solution-based competitions using data sets in high-interest industries. Learners may also explore a vast trove of data modeling discussions, trending plug-in models, and useful code snippets. Here are some great resources to get started in Kaggle:

- Datasets: explore and download a vast collection of data sets while up-voting your favorite collection.
- Competitions: commit individually or collaborate in a team towards data competitions for the possibility of financial rewards. Even without winning the competitions, this is a great way to network with other analysts.
- Learn: use this resource for an additional perspective on data visualization, linear regression techniques, or time series charting code.

Web designer

A web designer is responsible for the styling and layout of web pages containing text, graphics, and video. Web designers generally use Hypertext Markup Language v5 (HTML5) and Cascading Style Sheets (CSS) to create web pages.

HTML5 provides structure for web pages and is used to connect to hosting platforms. Learn more about HTML5 and CSS using these resources:

- HTML Tutorial: an introduction to HTML with links to HTML5 features, examples, and references
- HTML5 Cheat Sheet: a handy summary of HTML5 tags, attributes, and compatibility with HTML4
- HTML5 and CSS Fundamentals course: a free W3C course on edX; a verified course certificate can be issued for $199

CSS is used for web page design and controls graphic elements (color, layout, and font) and page presentation on multiple devices (large screens, mobile screens, and printers). Check out these cheat sheets for CSS:

- Interactive CSS Cheat Sheet: includes the most common CSS snippets for gradient, background, font-family, border, and much more
- 50 Best HTML & CSS Cheat Sheets: a list of 50 cheat sheets–choose a few that are useful to you

Mobile application developer

A mobile application developer uses programming to create applications used on laptops, mobile phones, and tablets. The most popular programming languages for mobile application developers are Swift, Java, and C#.

Swift (for Apple platforms) is an open source scripting language for macOS, iOS, watchOS, and tvOS. Its main goal is to make applications run faster. Browse these resources for more information about Swift:

- Swift.org: an open source community with resources to learn how to use Swift, including videos and sample code
- Swift developer site: an Apple developer website with information for developers who want to use Swift
- Swift development resources: Apple's collection of documentation, sample code, videos, and recommended books

Java (for Android devices) is the official language for Android development. The article I want to develop Android apps - which languages should I learn? explores some other languages used for Android development. Check out these resources for Java:

- Android Studio: a downloadable integrated development environment (IDE) with tools to build apps for Android devices
- Build your first Android app in Java: instructions for installing Android Studio and creating your first app
- Java tutorial for beginners: write a simple app with no previous experience: an overview of how to learn Java, with examples

C# (pronounced C-sharp) is an object-oriented programming language that is widely used to create mobile apps in the .NET open source developer platform. Xamarin extends the .NET platform with a framework for developers to create cross-platform mobile apps for both iOS and Android. Here are a few resources to help you learn C#:

- Microsoft .NET learning materials for C#: includes free courses, tutorials, and videos to learn the programming language C#
- Microsoft Xamarin learning materials: includes free courses, tutorials, and videos to learn about mobile development with Xamarin
- Xamarin Tutorial - build your first iOS or Android app in C#: instructions for building a mobile app that displays the text "Hello World"
- Learn C# from Codecademy: a website with free basic interactive lessons, and additional activities that can be accessed with a monthly subscription

Web application developer

A web application developer designs and develops network applications used across the web. The most popular programming languages used by web application developers are Java, Python, Ruby, and PHP.

Java is widely used to create enterprise web applications that can run on multiple clients. Java's main strength is its "Write Once, Run Anywhere" (WORA) approach.Browse these resources to learn more about Java:

- Oracle Java Tutorials: Java tutorials from Oracle documentation
- Java for Beginners: a free Java course for beginners from the website "Home and Learn"

Python is a general-purpose programming language. Check out the Python resources listed in the data analyst section.

Ruby is a general-purpose, object-oriented programming language used for web application development. Ruby isn't the same as Ruby on Rails, which is an open source web application framework that runs using Ruby. Browse these resources to learn more about Ruby:

- Ruby news: information about the latest Ruby releases and links to other resources
- Ruby documentation: includes guides, tutorials, and reference material to help you learn more about Ruby
- Ruby programmer's guide: a tutorial and reference guide for Ruby
- Learn Ruby from Codecademy: a website with free basic interactive lessons, and additional activities that can be accessed with a monthly subscription

PHP is a scripting language particularly suited for web application development. It was based on Perl, another programming language. PHP is simple, flexible, and relatively easy to learn. Check out these resources to learn more about PHP:

- PHP downloads and documentation: information about the latest PHP releases and links to other resources
- PHP the Right Way: a quick reference for popular PHP coding standards
- Interactive PHP tutorial: a free tutorial that runs PHP code in exercises

Game developer

A game developer is an application developer who specializes in video game creation. Game developers most commonly use the programming languages C# and C++.

C# is an object-oriented programming language that is widely used to create games. Check out the C# resources listed in the mobile application developer section.

C++ is an extension of the C programming language that is also used to create console games, like those for Xbox. Browse more information about C++:

- Microsoft resources for C++: learn how to install the Visual Studio IDE and write C++ code
- Microsoft C++ and C# code samples for gaming: a resource with over 40 C++ and C# code samples for gaming
- Interactive C++ tutorial: a free tutorial that runs C++ code in exercises

Tips for learning programming languages

Here are a few tips to follow when you start learning a new programming language:

- Define a practice project and use the language to help you complete it. This makes the learning process more practical and engaging.
- Keep previous concepts and coding principles in mind. Many of these are transferable between programming languages. So, after you have learned one language, learning a second or third programming language tends to be much easier.
- Create and keep good notes and cheat sheets in whatever format (handwritten or typed) that works best for you.
- Create an online filing system for information that you can easily access while you work in various programming environments.

Mark as completed

Like **Dislike** **Report an issue**