

# How to Code

As technology becomes more and more accessible, the need for programmers is always increasing. Coding is a skill learned and perfected over time, but everyone has to start somewhere. There are a variety of languages that are perfect for...

Method 1 of 2:

## Learning a Language



**Don't worry too much about what language you pick.** Many beginning coders struggle with what language to pick when they first start learning. (Because they don't know where to start with their first code). The actual language that you start learning on doesn't make a big difference when it comes to learning about data structures and logic. These are the skills that matter most, and they can be honed with any language.<sup>[1]</sup>

1. When picking a language, just focus on what kind of development you want to start with and pick an introductory language from there. For example, if you want to learn web development, start with HTML5, supplemented with CSS, JavaScript, and PHP. If you want to do desktop applications, start with C++ or another basic programming language.
2. If you make coding a career, you will find that you may never use the language that you first learned to code. Instead, you will be learning languages as you go through documentation and experimentation.



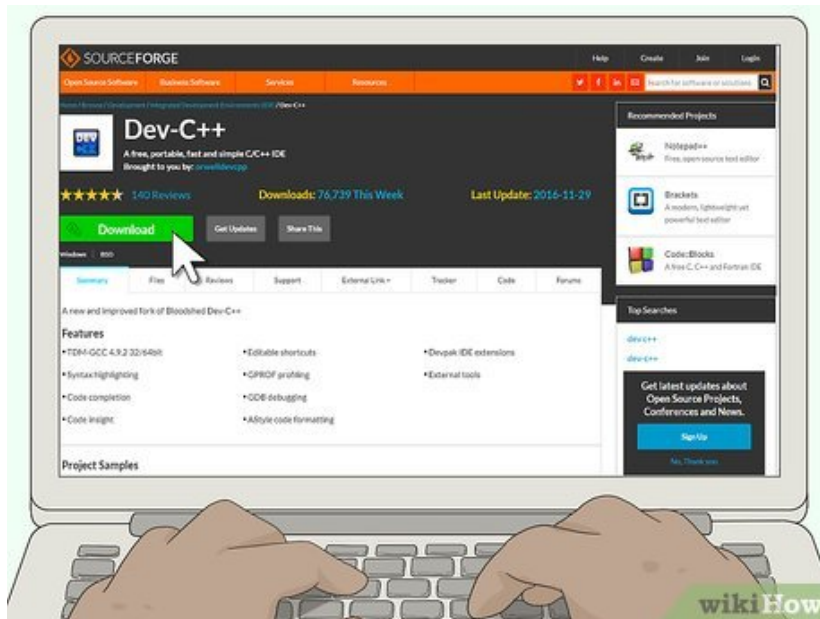
3.



**Download a good text editor.** Many programming languages allow you to use external text editors when you write your code. Find a text editor that will allow you to see indentations and code markup.

1. Popular programs include Notepad++ (Windows), TextWrangler (OS X), or JEdit (any system).

4.



**Download any necessary compilers.** Some programming languages require a compiler in order to run the code that you create. Compilers translate the code you write into a lower-level language that the machine can process. Many compilers are open-source and free to use. Languages that require compilers include:

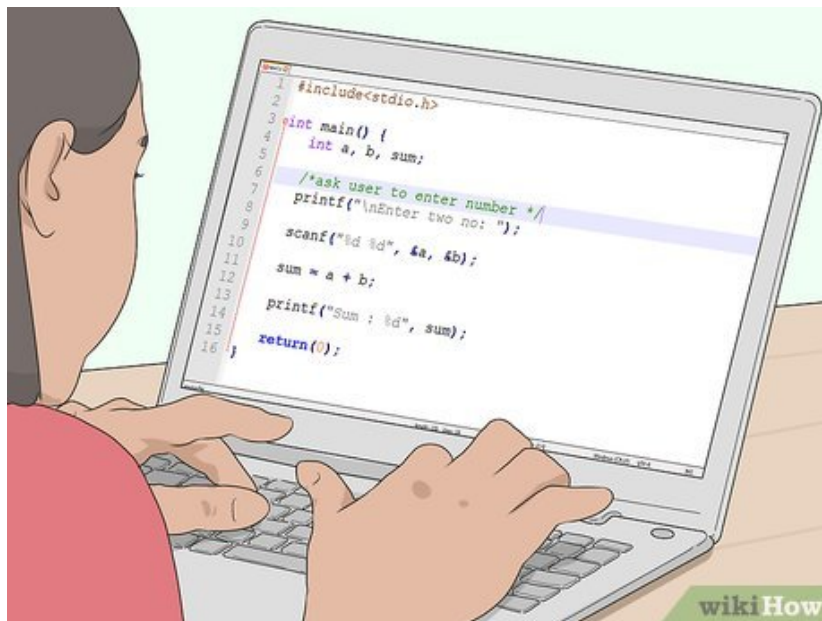
1. C
2. C++
3. C#
4. Java
5. BASIC
6. Fortran

5.



**Start your first project.** Pick a good introductory project that will help you get your feet wet. There are a variety of suggestions and tutorials online, but some places to start are basic websites for HTML, basic database and form functions with PHP, or simple programs with any of the compiler languages.

6.



**Comment all of your code.** All programming languages have a comment feature that allows you to enter text that is ignored by the compiler. This allows you to add comments to your code. These comments are crucial, both for letting others know how your code works and for reminding yourself what your code does.

1. You can also use the comment function to quickly remove code from your program for testing purposes. Put comment tags around the code you want to exclude and then remove the comment tags to return the code.

7.



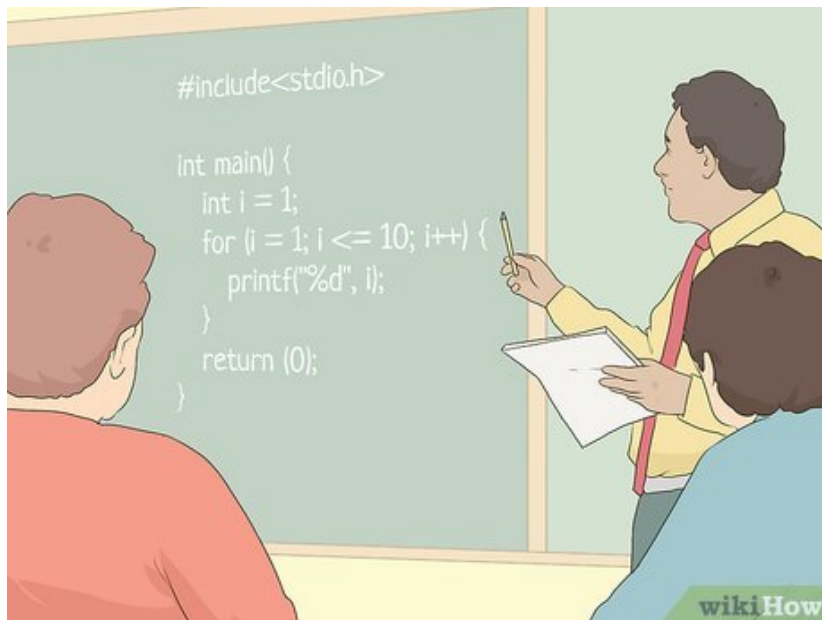
**Pick apart other programs or web projects.** When you're learning as you go, don't be ashamed to look things up and see how others have tackled the same tasks. Take time to understand why the code is doing what it does.

1. See this guide for details on how to view the source code of websites.

Method 2 of 2:

## Expanding Your Knowledge

1.



**Take classes.** Community colleges, tech schools and online programs offer certifications and classes that will help you get a job as well as teaching you programming. An advanced degree like Computer Science is not always necessary, but could help to get a full time programming job.<sup>[2]</sup>



3.



**Learn more languages.** Once you've got a good grasp on your initial language, you can start to branch out. Find another language that complements the one you know, or pick a language for a specific job that interests you. Complementary languages, such as HTML and CSS, are often the easiest to learn.

1. Java is one of the most popular languages, and there are often lots of opportunities for Java developers. Java can run on a huge variety of systems and has innumerable applications. Java is used for Android applications, one of the fastest growing markets.
2. C++ is highly recommended for budding video game developers. Learning how to code in Unity (a widely used, low-cost game engine) and UDK (the code for the popular Unreal engine) can help open some doors, though they aren't as useful outside of the video game industry.
3. If you want to make iPhone apps, Xcode and Objective-C are going to be your primary tools. You'll also need a Mac, since Xcode can only compile on a Mac.
4. Python is a server scripting language that is one of the easier languages to learn. Python is used for web services such as Pinterest and Instagram, and is simple enough to learn the basics in just a few days.

4.



**Be patient.** You will often be faced with challenges while programming, especially when it comes to hunting for bugs or implementing a new idea. You'll have to learn to be satisfied with achieving small results rather than solving a whole puzzle at once. Patience leads to more effective code, which will lead to better-performing programs and happier colleagues.

5.



**Learn to work with others.** If you have multiple people working on a project, you get multiple viewpoints on how to go about it. Working on teams is virtually unavoidable in the business world, so unless you're planning to develop everything independently, be prepared to work with others.

6.



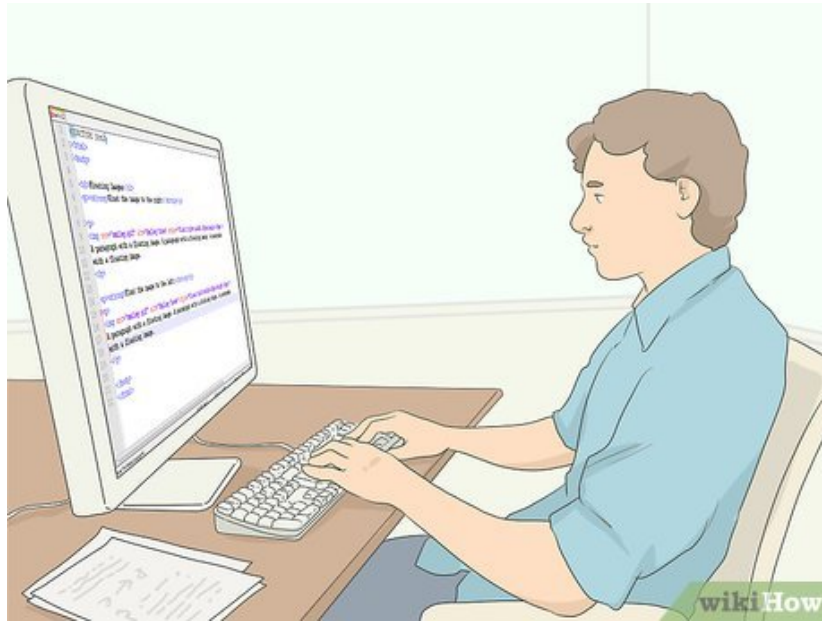
**Get a job where you can practice your coding skills.** Work as a volunteer designing websites or writing desktop applications. A part time job with a small company can lead to opportunities to write code for websites or simple applications.

7.



**Connect with other programmers.** There are countless communities and gatherings of developers that can help support and inspire you. Look up local programming conventions, participate in a hack-a-thon or game jam (timed events with a common theme), and sign up on some programming forums to start expanding your exposure and network.

8.



**Practice, practice, practice.** It is estimated that it takes about 15,000 hours before you can be considered an expert in computer programming.<sup>[3]</sup> This is years of applied practice. True mastery of programming will come only after you put the time in to practice and become proficient.

1. Try to spend time programming every day, even when you're not working. Programming in your free time can lead to breakthroughs and new ideas.

You finished reading the article "**How to Code**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.