Contents

What you will need

You need the following to join in the fun:

- **PC**. I'll assume that you have a Windows desktop or laptop machine. Specifically you're using Windows XP as your operating system (OS). Don't worry if you don't know what OS you're using. If you do not know what you're using, then you're most probably using Windows.
- **Internet access**. Fast internet access is good. If you have a dialup connection, then you will need to be patient.

Resources

There are many ways to learn Python programming. Since you're reading this page, I'm assuming that you're using my notes. So I assume you have access to all of them. The notes are designed to be covered in 8 weeks.

If you're learning this in our Columbia College Programming Club (CCPC), then besides the notes you'll have the following resources:

- During club meetings you will work in **small groups**. Learning from each other helps to speed up your learning process.
- There are also **mentors** in the club that will help you learn. These mentors are Computer Science students in Columbia College.
- The club maintains a **message board** where you can interact outside the club meetings.

All the notes are free. Furthermore, I have selected reading material and references so that you may deepen your mastery of the Python programming. The extra references are freely available on the web. All these can be accessed through our club's web site.

What else you need before the fun begins

OK. So you have the equipment, you have the notes, and you're a member of CCPC. You're all fired up to have fun. But wait. There's something else ...

I need to talk about **you**.

Programming is like building something. While a electrical or computer engineer builds hardware (and more!), a software engineer builds software. Both hardware and software "thingies" that they build are like machines.

There are two main obstacles on your way to become a software engineer (and this is somewhat similar to the hardware engineers). First you will need to learn to use the tools. Next you need to know how to design a solution. You can think of your tools as things to help you transform your design to a real machine.

For writing programs in Python, the main tool will be a programming language called Python.

Learning a programming language is no different from learning any other human language. Certain words have special meanings. And you basically use these words to control the PC. (There are details ... but we can ignore them for the time being).

Designing a solution more or less means planning out what you want to do. This is much harder than learning the tools.

But back to learning the language of Python. Much of learning the language is actually very simple. The only thing you need to do in order to know the language well is to **practice** using the language. Like learning any other language, this takes time. Much of remembering the words or grammar of a language involves repetitive usage. Therefore you need to **persevere** and **be patient**. I guarantee you that your patience will be greatly rewarded.

Remember to make full use of the resources available.

Practice here means whenever you see a **computer program**, you **type** it, **run** it, and **understand** why it behaves that way. Don't just stare at the program.

Persevere here means don't give up! Programming is partly like learning to use a new language. So the first few weeks will be difficult. After that, it get easier and easier.

Read the notes and if necessary the references. Practice using the Python language to **solve problems** given in the notes. **Ask** questions during club meetings. If you need help outside the club meetings, use the message board. Let's not only have fun, but have fun together by learning together. Your club members will be really grateful if you can answer their questions on the message board.

If you are determined to learn to program, then you will. Someone will be around to help you.

OK. Enough talk. Let the fun begins ...