

1

Introduction and Basics

This practical hands-on book introduces the Python programming language for rapid prototyping of mobile device applications. It effectively teaches how to program easily on Nokia smartphones that are based on Symbian OS and the S60 platform. A wide range of smartphone functionalities are covered, including camera, sound, graphics, Bluetooth, Internet, positioning, SMS messaging and many more.

Mobile Python – or, more formally, Python for S60 (see Figure 1.1) – empowers you to do fun and engaging stuff with your mobile phone. You can start programming shortly after getting into this book. Being able to see results quickly on the phone guarantees to bring inspiration and makes programming these gadgets fun!

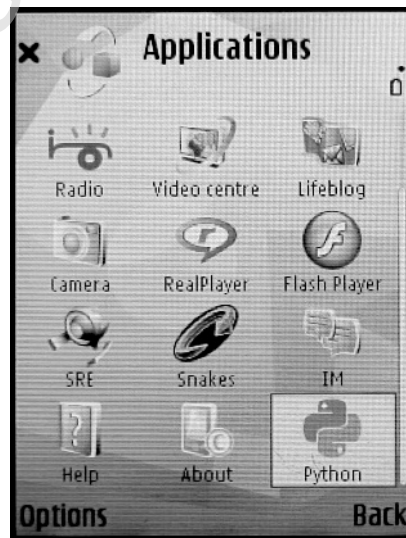


Figure 1.1 Python for S60

Development on the Symbian platform has been time-consuming in the past and it has required in-depth knowledge of C++ or Java. Python for S60 remedies this problem. It is easy to learn and takes only a few days to get into most of its features. Novice programmers, artists and people from creative communities can innovate and contribute applications to the mobile space.

Python for S60 brings the increasingly popular Python programming language to the mobile platform. You can use this book to learn the Python programming language by way of Python for S60 or use your previous Python knowledge to get into mobile programming in no time. Similar to traditional Python, Python for S60 is released under an open-source license, so you will be backed up by an enthusiastic community of talented developers and a large library of extension modules.

Python for S60 allows you to go through a fast iterative design cycle by providing an elegantly simple but powerful platform for your programs, which can be rapidly adapted to real-world requirements. With completely free and open tools you are able to create useful and appealing applications based on your own ideas.

Our message to all you creative and innovative people out there is: use your talent, skills, ideas and energy to inspire the world! May this book help you to do so!

Although the book is written in a style that beginners can cope with, it hopefully also offers experienced programmers new insights and in-depth knowledge. It is not a traditional programming text book with meticulous coverage of every aspect of a programming language. Instead it adopts a light and engaging tone that helps the reader to proceed through the chapters in a practical hands-on manner, steadily increasing knowledge through learning by doing.

The material in this book has been reformulated and refined through dozens of workshops and tutorials offered by the authors at a wide variety of institutions and companies across 17 countries, including MIT, Stanford University, the University of Art and Design, Helsinki, and Yahoo Research, Berkeley. Among the hundreds of applications created by participants in these events are games for social interaction; applications for interacting with large public displays, sensors or robots; and personal organizer applications. This book is intended to broaden the audience for these kinds of applications even further, enabling you to create new mobile applications that may have seemed to be out of your, or anyone's, reach before.

We have used the book's material to build many applications and prototypes for companies such as Nokia. We have also helped several universities around the globe to conduct research projects on robotics, sensor networks, positioning and data collection using Python for S60.

Mobile phones are carried by over two billion people – far more than the 200 million who carry laptop computers. Despite the greater penetration of mobile phone *users*, the number of mobile phone *programmers* is far lower than the number of people who have learned to program personal computers. This book aims to change that.

There is no doubt that the built-in features of mobile phones have empowered a generation of people to connect with others more effectively than ever before. However, these standard applications are just scratching the surface with respect to unleashing the true creative potential of our culture. We can expect to see a user-driven innovation era and user-generated mobile applications are in the near reach. This book is inspired by [von Hippel 2005] and lays out a practical path for how innovation driven by lead users can become a reality on the mobile platform.

By reading this book, you will become fully equipped to be one of the new lead users. You may gain inspiration and motivation that turns you into an innovator and a contributor to the developer community for mobile applications.

1.1 Why Does Python Make a Difference?

Only skilled and experienced programmers were previously able to build mobile applications using C++ or Java. As a result, many people often gave up early or never really started.

The emergence of Python for S60 offers a crucial turning point, as it brings the Python programming language to the mobile space. This makes mobile development approachable for many new developers that were previously excluded.

Python for S60 can drastically reduce development time; it allows development with completely free and open tools and reuse of open source code modules. This can potentially lower costs and other barriers to entry for first-time developers of mobile applications.

With Python running on Symbian OS, the short development cycle gives a shortcut from the inspiration of an idea to its implementation. It makes rapid prototyping on the mobile platform easy and efficient by wrapping complex, low-level technical details in simple interfaces.

In recent years, the processing power and memory capacity of smartphones have drastically advanced which have made it possible to run an interpreted language such as Python on such devices.

Modern smartphones offer a rich set of features, including WiFi, camera, sound recording and networking that could easily be combined and used for new types of applications. As this book shows, Python for S60 makes accessing these features extremely convenient, letting you focus on your own application idea instead of on the intricacies of the platform.

The mobile space and the Internet are rapidly converging. Client–server solutions can be developed quickly in Python for S60 in combination with a web services back end, such as Django or Ruby on Rails, or using a custom server, which could also be implemented in Python. Being able to use a single agile language, and even some of the same code, on both the client and the server is a great benefit. Chapter 8 deals with advanced topics in networking, such as peer-to-peer communication and turning your mobile phone into a web server. Chapter 9 is dedicated to combining web services, such as Yahoo! Maps and Flickr, with Python for S60.

1.2 How to Use this Book

With a simple text editor and a Nokia smartphone, you can instantly code and test working applications found in this book. You will learn things by running small yet fully working programs on the actual phone to see what they do and then study and modify them on your computer. By experiencing the hands-on coding style and ready-to-use programs, you can soon feel success that keeps you inspired throughout the book.

This book includes over 100 example programs that demonstrate different aspects of the mobile platform. The code for all example programs can be downloaded for free from www.mobilepythonbook.com. Some of the programs are small scripts that show you how to automate tasks, such as sending SMS messages; some are full-scale applications with graphical user interfaces. The examples are designed in such a way that they often build on each other, which makes learning easier through repetition.

Most of the examples demonstrate a specific functionality, so they are short and easy to understand – the median length of examples is just 17 lines of code! However, many of the examples are not just for playing around but are already usable solutions in their own right.

The examples in this book are designed to be combined, modified and enhanced by your own ideas. Throughout the book, examples are cross-referenced with each other, so the book can also be ‘dipped into’ and not necessarily read from cover to cover.

Besides many examples, Chapters 3 to 6 contain some Python language lessons. These lessons, spanning at most a page, introduce you to the basic concepts of the Python programming language. They provide you necessary knowledge on the language, so you can follow examples and extend them by yourself. Even though the lessons cover the basic concepts in Python, they omit many of its interesting and useful features. As the lessons deal with Python in general and not specifically about Python on the mobile platform, you can easily find more information about the topics in many books and Python tutorials on the web (see the References section).

However, you will be surprised how far you get with 14 one-page lessons on Python!

1.3 Who Is this Book For?

Since Python is easy to learn, you do not need to master any advanced computing concepts before touching this book. You only need an understanding of some basic programming principles or a scripting language, such as PHP or JavaScript, to get started with programming in Python for S60.

Because of the steep learning curve of most mobile platforms, the creative community and novice programmers have been excluded from developing their own ideas for applications for mobile phones. We believe that Python for S60 remedies this problem. Therefore, this book is primarily aimed at people who are new to mobile programming, who lack the time and enthusiasm to learn C++ or Java or who cannot afford to spend weeks or months on development. Rapid prototyping with Python for S60 gives them a fast entry ticket. At the same time, many experienced developers find Python a refreshingly agile alternative and may enjoy the additional sense of elegance and freedom that it provides.

We think the following groups of people will benefit from this book:

- **Lead users and ‘prototypers’**
If you want to gain knowledge and practical skill for quickly programming working prototypes of innovative mobile applications, you may find Python for S60 your toolkit of choice. It is open source, so you will not be hindered by closed, proprietary platforms which severely restrict your freedom to experiment. If you are an enthusiastic mobile phone user who has many ideas on new ways of using your phone, we show you how to realize your own novel concepts in practice.
- **Mobile artists and mobile interaction designers**
Python for S60 will open the door for you to the world of programmable mobile phones. As a creative media artist or interaction designer you might be less constrained by conventional thinking than a typical software engineer – and this is your opportunity. By using Python for S60 to combine several smartphone features, for instance, camera, sound recording, SMS and Bluetooth, you can explore new frontiers of art and design. If you are a designer who has worked with ActionScript for Flash or Director’s Lingo and want to start creating mobile applications, you will find Python for S60 already familiar to you in many respects.
- **Web developers**
If you have worked with PHP or JavaScript but haven’t used Python before, start today! Python for S60 allows you to quickly write mobile client applications that can be part of your website or service. You can also create novel mashups that combine information from a web service with that from your physical surroundings.

- **Experienced mobile application developers**
Converting to Python for S60 makes development feel light, happy and productive while retaining your old powers. Whenever a colleague complains that Python is slow or it misses feature X, sit down and write a C++ extension for Python in a few hours. This way, you get the best of both worlds.
- **Researchers**
Python for S60 is a perfect platform for doing various kinds of research. It is the easiest way to collect rich empirical data with mobile phones and you can prototype novel applications quickly. Since it is open source and easily extendable in C++, you can even perform some demanding computations on the device. Moreover, you can get started right away, since Nokia smartphones are ubiquitous, off-the-shelf products and Python for S60 is freely available.
- **Teachers and students**
If you are teaching introductory programming classes in Python, this book might serve as a source of motivation and inspiration for your students. The smartphone is a rich and ready-to-use platform with many built-in functionalities such as networking, camera, graphics, image handling, GUI design, Bluetooth, telephony and more, so a plethora of concepts can be demonstrated and experimented on it. Nowadays, many students are motivated by the mobile phone, which is an integral, personal part of their lives, rather than by a PC. Being able to easily 'pimp up' the mobile phone and learn programming at the same time is a strong incentive even for younger students.
- **Python community**
If you are one of the hundreds of thousands of Python programmers and want to enter the mobile space using completely free and open tools as well as open-source code modules, Python for S60 is an ideal path for you to take.

1.4 What Are Symbian OS, S60 and Python for S60?

Symbian OS is an operating system designed for mobile devices. It includes associated libraries, user interface frameworks and reference implementations of common tools. As a descendant of Psion's EPOC, it runs exclusively on ARM processors. Symbian OS APIs are publicly available and anyone can develop software for Symbian OS.

S60 is a software platform for mobile phones based on Symbian OS. It is Nokia's user interface framework that runs on all Nokia S60 devices on top of Symbian OS. S60 is one of the leading smartphone platforms in the world. It is developed by Nokia, which licenses it to other

manufacturers including Lenovo, LG Electronics, Panasonic and Samsung. S60 consists of a suite of libraries and standard applications based on Symbian OS APIs.

Python is a dynamic object-oriented, open-source, computer-programming language. It can be used for many kinds of software development, for instance, to create stand-alone programs, scalable server software or small scripts – Python’s roles are virtually unlimited. Python was created by Guido van Rossum and is distributed under an OSI-approved, open-source license that makes it free to use, even for commercial products.

Python is often used for prototyping and teaching introductory programming classes. It can be learned in a few days and offers strong support for integration with other languages and tools. Python comes with an extensive standard library, thus its slogan is ‘Python – batteries included’.

Python runs on most common and legacy platforms, for example, Windows, Mac OS X, Linux/Unix, OS/2, Amiga and Palm OS. It also runs on Nokia S60 2nd and 3rd Edition mobile phones – that is where this book comes in. Python has also been ported to the Java and .NET virtual machines. It is an interpreted programming language that combines remarkable power with clear syntax; it has modules, classes, exceptions, high-level dynamic data types and dynamic typing.

Python for S60 brings the Python programming language to the S60 platform. Python for S60 is based on Python version 2.2.2. It supports many of the Python Standard Library modules but also includes several modules specific to the mobile platform, for example, native GUI elements, Bluetooth, networking, GSM location information, SMS messaging, access to the camera, and more. The full range is described in detail throughout this book. Nokia makes Python bindings for Symbian OS APIs that are publicly provided on S60 devices. All examples in this book were made using Python for S60 version 1.4.0.

1.5 Python Terminology in this Book

In this book, the term ‘Python’ may refer to three different concepts (see Figure 1.2).

Figure 1.2(a) shows the Python programming language, which is the same both on a PC and a phone, although the PC cannot access the phone’s functionalities.

Figure 1.2(b) shows Python for S60, which runs, or interprets, the Python language on the S60 smartphone platform and provides interfaces to the phone’s functionalities.

Figure 1.2(c) shows a Python interpreter that is used to run Python on a PC. In some examples of this book, you need that as well.

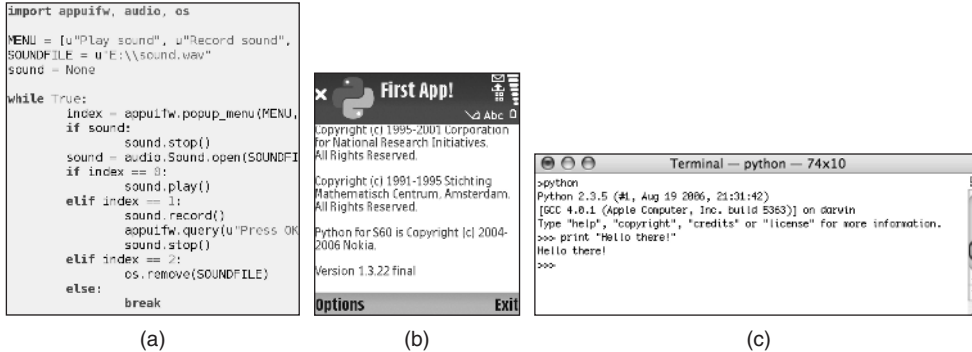


Figure 1.2 Three meanings of ‘Python’: (a) Python code, (b) Python for S60 and (c) Python interpreter on a PC

In practice, almost always in this book we are talking about Python for S60 and thus we use the abbreviation ‘PyS60’ for clarity. The Python language lessons talk about Python the programming language, so they apply on a PC as well. On rare occasions we need to run the Python interpreter on a PC – in these cases we state this clearly in the text.

1.6 Democratizing Innovation on the Mobile Platform

Eric von Hippel, Professor and Head of the Innovation and Entrepreneurship Group at the MIT Sloan School of Management, discusses in his book ([von Hippel 2005]) the phenomenon that users can generate innovation if a toolkit – based on a platform product – is provided, that allows them to create user-developed modifications that suit their own needs. He calls this ‘distributed innovation’ by ‘lead users’. Lead users have the following characteristics:

- They are ahead of most users in their population with respect to an important market trend and so are experiencing needs today that will later be experienced by many other users.
- They know and understand their own needs well.
- They are close to ‘real situations’, so the products they develop will appeal to others too.
- They may innovate if they want something that is not available on the market.

Although his empirical data is collected from other fields, his arguments around his lead-user theory matches well with what we practically experience from outcomes of workshops with creative students learning

Python for S60. We see that they innovate instantly, creating unusual and novel applications based on their own ideas, fulfilling their own needs and enabling them to share their innovations.

Can Python for S60 democratize innovation on the mobile platform? We have the vision of a big garden full of beautiful flowers, each representing a novel mobile application created by a lead user fitting his or her own needs. Will you help to grow these flowers? Distributed innovation on the mobile platform is possible and perhaps this book can be a starting point to help trigger it.

Let's look now at some of the topics and arguments that [von Hippel 2005] states and how they map to Python for S60.

1.6.1 User-Centered Innovation Process

[von Hippel 2005] explains that users that innovate can develop exactly what they want, rather than relying on manufacturers to act as their (often imperfect) agents. It may be that the needs of local user communities differ and so local lead users really may be the world's lead users with respect to their particular needs. Further, [von Hippel 2005] argues that users generally have a more accurate and more detailed model of their needs than manufacturers. The information assets of some particular users are close to what is required to develop a particular innovation. Users tend to develop innovations that are functionally novel, requiring a great deal of user-need information and use-context information for their development.

Using Python for S60, users can program applications based on their interests and own ideas – even integrating local cultural aspects. Users with few programming skills can innovate, iterating through new ideas rapidly.

1.6.2 Motivation of Lead Users

[von Hippel 2005] also states that, for individual user–innovators, enjoyment and learning of the innovation process can be important.

To program with Python for S60 is often described by people as fun since it generates reward and motivation through a seamless process of iterative development and design with instant coding, modifying and testing on the real phone in the real mobile network. People can easily learn how to start coding their own mobile application ideas with hardly any learning curve.

1.6.3 Sharing of Innovations

According to [von Hippel 2005], users often achieve widespread diffusion: they often 'freely reveal' what they have developed. Individual users

can benefit from innovations developed and shared by others. Freely revealing users may benefit from enhancement of their reputation from positive network effects because of increased diffusion of their innovation.

Coding Python for S60 modules and making them public for sharing with others is becoming a common practice. The projection is for huge potential if many lead users come on board to contribute.

1.6.4 Development of Products by Lead Users

[von Hippel 2005] states that studies have shown that many of the innovations reported by lead users are judged to be commercially attractive or have actually been commercialized.

Python for S60 allows lead users and creative minds to prove the concept of their own ideas, ideas that fulfill some real needs and can potentially be shared with others. In our experience from giving mobile phone programming workshops around the world, when we ask people what they would like to do with their phone, almost everyone comes up with a unusual idea.

1.6.5 Toolkit

Further, [von Hippel 2005] says that the ability of users ‘to innovate is improving radically and rapidly through improved access to easy-to-use tools and steadily richer innovation commons. Companies learn to supply proprietary platform products that offer user–innovators a framework upon which to develop and use their improvements. . . . kits and design tools . . . can serve as platforms upon which to develop and operate user-developed modifications.’

With Python for S60, a toolkit provides the S60 platform with many open APIs and a rich set of features and phone functionality into which creative users can tap. With a hands-on tutorial, such as MobiLenin, lots of starting code and the help of this book, people can program their own ideas quickly and in a powerful manner.

1.7 The Process of Rapid Prototyping with Python S60

Python for S60 can be seen as an ideal prototyping tool. Turning an idea or concept into code for a working software prototype can be done in weeks, if not days. Many fully functioning code examples, such as the ones found in this book, can be used as a springboard to get started with the rapid prototyping process. Rapid prototyping with Python for S60 may:

- save much development time
- save many development costs
- allow a developer team to turn several ideas into prototypes within budget and time limits, instead of building just one.

1.8 Summary

Python for S60 is all about having one's head in the clouds, one's hands in mud and one's feet on the ground. In other words, it allows you to come up with ideas and implement and test them in a straightforward and pragmatic manner. Dive in, develop, share and enjoy!

In the next chapter we take the first practical steps. We start by installing the Python for S60 interpreter on your phone and then create our first PyS60 program.

