An\_Introduction\_Python\_Computer\_Programming\_c01

Yue Zhang

An Introduction to Python

and Computer Programming

p12

Chapter 1

An Introduction to Python and Computer

Programming

1.1 Introduction

Computers play a very important role in the digital world, and in our daily lives. They are central to smart TVs, phones, homes cars, metros, airplanes, robots, factories and many other digital systems. We use computers through these systems every day.

—-------------------------------------

Las **computadoras juegan un papel muy importante en el mundo digital** y en nuestra vida diaria. Son fundamentales para televisores inteligentes, teléfonos, casas, automóviles, metros, aviones, robots, fábricas y muchos otros sistemas digitales. Usamos computadoras a través de estos sistemas todos los días.

—--------------------------------------

At work, computers have become an indispensible tool to more and more people. Fund managers use computers for analytical trading; architects use computers to make and verify building design; linguists use computers to study and annotate text corpora; natural scientists use computers to simulate large experiments in physics and chemistry; supermarkets use computers to plan and schedule the most efficient home delivery times and routes. As a result, fundamental knowledge on effectively utilizing computers is useful for all industries and disciplines.

—--------------------------------------------------

En el trabajo, los ordenadores se han convertido en una herramienta indispensable para cada vez más personas. Los administradores de fondos usan computadoras para operaciones analíticas; los arquitectos usan computadoras para hacer y verificar el diseño de edificios; los lingüistas usan computadoras para estudiar y anotar corpus de texto; los científicos naturales usan computadoras para simular grandes experimentos en física y química; los supermercados usan computadoras para planificar y programar los tiempos y rutas de entrega a domicilio más eficientes. Como resultado, **el conocimiento fundamental sobre el uso efectivo de las computadoras es útil para todas las industrias y disciplinas**.

—--------------------------------------------------

Computers integrate hardware and software. Hardware refers to physical components, such as chips, keyboards and monitors, while software refers to operating systems and application programs, such as Office, Firefox and iTunes. Software operates on hardware to provide useful functionalities to users, such as text editing, Internet browsing and gaming. On the other hand, hardware provides fundamental support to software, by offering a set of basic instructions, which software combines to achieve complex functionality.

—--------------------------------------------------

**Las computadoras integran hardware y software**. El hardware se refiere a componentes físicos, como chips, teclados y monitores, mientras que el software se refiere a sistemas operativos y programas de aplicación, como Office, Firefox e iTunes. El software opera en el hardware para proporcionar funcionalidades útiles a los usuarios, como la edición de texto, la navegación por Internet y los juegos. Por otro lado, el hardware proporciona un soporte fundamental al software, al ofrecer un conjunto de instrucciones básicas, que el software combina para lograr una funcionalidad compleja.

—--------------------------------------------------

Programming is the design of software using a set of basic instructions. Programming languages abstract hardware instructions into basic statements, which are much easier to understand and handle. Using a programming language, one can write complex software, achieving tailored functionalities by combination of basic statements. This is the ultimate way of utilizing a computer, and is necessary when existing software fails to satisfy the requirement, such as the need to compute novel equations, or simulate specific experiments in an engineering discipline.

—--------------------------------------------------

**La programación es el diseño de software utilizando un conjunto de instrucciones básicas.** Los lenguajes de programación resumen las instrucciones de hardware en declaraciones básicas, que son mucho más fáciles de entender y manejar. Usando un lenguaje de programación, uno puede escribir software complejo, logrando funcionalidades personalizadas mediante la combinación de declaraciones básicas. Esta es la forma definitiva de utilizar una computadora y es necesaria cuando el software existente no cumple con el requisito, como la necesidad de calcular ecuaciones novedosas o simular experimentos específicos en una disciplina de ingeniería.

—--------------------------------------------------

1.1.1 Python and Computer Programming

Python was invented in the 1990s, named after the comedy show Monte Python’s Flying Circus. Due to its simplicity and elegance, it gained popularity among programmers quickly, and was adopted by large companies, such as Google Inc. Compared with other programming languages, such as Java and C++, Python is particularly useful for learning computer programming.

—---------------------------------------

**Python se inventó en la década de 1990** y lleva el nombre del programa de comedia Monte Python's Flying Circus. Debido a su simplicidad y elegancia, rápidamente ganó popularidad entre los programadores y fue adoptado por grandes empresas, como Google Inc. Comparado con otros lenguajes de programación, como Java y C++, Python es particularmente útil para aprender a programar computadoras.

—----------------------------------------

Figure 1.1 shows a comparison between Python, Java and C++, three popular programming languages. Among the three, C++ was invented the earliest. Its statements were the closest to hardware instructions. As a result, C++ programs run very fast, and can be optimized in many ways. However, mastering C++ requires knowledge in computer hardware. Java was invented decades after C++. It offers higher levels of abstraction compared to C++, making it easier to program. As a trade-off, Java programs run slower than C++. Python takes one step further in abstracting away hardware details, offering a conceptually very simple system for statement execution. As a result, Python is the easiest to learn and fully master, but Python programs are typically the slowest to run.

—-----------------------------------------------

La Figura 1.1 muestra una comparación entre **Python, Java y C++**, tres lenguajes de programación populares. Entre los tres, C++ fue el primero en inventarse. Sus declaraciones eran las más cercanas a las instrucciones de hardware. Como resultado, los programas de C++ se ejecutan muy rápido y se pueden optimizar de muchas maneras. Sin embargo, dominar C++ requiere conocimientos en hardware informático. Java se inventó décadas después de C++. Ofrece mayores niveles de abstracción en comparación con C++, lo que facilita la programación. Como compensación, los programas de Java se ejecutan más lentamente que C++. Python va un paso más allá al abstraer los detalles del hardware, ofreciendo un sistema conceptualmente muy simple para la ejecución de sentencias. Como resultado, Python es el más fácil de aprender y dominar por completo, pero los programas de Python suelen ser los más lentos de ejecutar.

—-----------------------------------------------

In terms of functionalities, Python is as powerful as Java and C++. It supports all the functionalities that typical computer hardware supports. In addition, libraries provided by Python and open-source contributers enrich the power of Python, making it the most convenient choice for many applications, such as text processing. Python libraries such as numpy and scipy are implemented in C++ in the lower level, offering very fast ways to perform scientific computation using Python.

—-----------------------------------------------

**En términos de funcionalidades, Python es tan poderoso como Java y C++.** Es compatible con todas las funcionalidades que admite el hardware informático típico. Además, las bibliotecas proporcionadas por Python y los contribuyentes de código abierto enriquecen el poder de Python, convirtiéndolo en la opción más conveniente para muchas aplicaciones, como el procesamiento de texto. Las bibliotecas de Python, como numpy y scipy, se implementan en C++ en el nivel inferior, lo que ofrece formas muy rápidas de realizar cálculos científicos con Python.

—-----------------------------------------------

Problem-solving skills are as important as knowledge of programming languages to computer programming. It is therefore important to learn common approaches to typical problems when learning computer programming. Python supports virtually all the programming paradigms that Java and C++ supports, and even more, making it convenient to apply flexible problem-solving techniques using the Python programming language.

—-----------------------------------------------

**Las habilidades para resolver problemas son tan importantes como el conocimiento de los lenguajes de programación para la programación de computadoras.** Por lo tanto, es importante aprender enfoques comunes para problemas típicos al aprender programación de computadoras. Python admite prácticamente todos los **paradigmas de programación** que admiten Java y C++, e incluso más, lo que facilita la aplicación de técnicas flexibles de resolución de problemas utilizando el lenguaje de programación Python.

—-----------------------------------------------

1.2 Preliminaries

As preliminary knowledge to Python programming, it is helpful to know the basic structure of a computer, the operating system, the file system, and how to install and configure Python. This section introduces these backgrounds, so that it is easier to embark on Python programming from the next chapter

—-----------------------------------------------

Como conocimiento preliminar a la programación de Python, es útil **conocer la estructura básica de una computadora, el sistema operativo, el sistema de archivos y cómo instalar y configurar Python.** Esta sección presenta estos antecedentes, para que sea más fácil embarcarse en la programación de Python a partir del próximo capítulo.

—-----------------------------------------------

1.2.1 The Computer

The structure of a computer and the mechanism in which Python code is executed are illustrated in Fig. 1.2. On the bottom of the figure is the hardware computer, which is physically tangible. It consists of three main components: the central processing unit (CPU), the memory and devices. They will be discussed in the next chapter. On top of the computer runs an operating system, such as Microsoft Windows, Linux and Mac OS. Operating systems connect computer hardware and software, providing basic interfaces to software programs for controlling the hardware. For example, the file system is an interface that operating systems provide for organizing data in the hard drive (hardware). Browsers, text editors, music players, and the vast majority of other application programs are managed by the operating system.

—---------------------------------------------

La estructura de una computadora y el mecanismo en el que se ejecuta el código Python se ilustran en la figura 1.2. En la parte inferior de la figura está la computadora de hardware, que es físicamente tangible. Consta de tres componentes principales: **la unidad central de procesamiento (CPU)**, **la memoria y los dispositivos**. Se discutirán en el próximo capítulo. En la parte superior de la computadora se ejecuta un **sistema operativo**, como Microsoft Windows, Linux y Mac OS. Los sistemas operativos conectan el hardware y el software de la computadora, proporcionando interfaces básicas a los programas de software para controlar el hardware. Por ejemplo, el sistema de archivos es una interfaz que proporcionan los sistemas operativos para organizar datos en el disco duro (hardware). Los navegadores, editores de texto, reproductores de música y la gran mayoría de los demás programas de aplicación son administrados por el sistema operativo.

—---------------------------------------------

As shown in Fig. 1.2, Python is an application program running on top of the operating system, just like a browser or a text editor. Its main function, however, is to execute generic Python code, rather than performing a specific task, which browsers and text editors do. IDLE is another application program, which provides an interactive interface for Python. On the top of the figure is Python code, which is executed by the Python application. Given a piece of Python code, or a Python program, Python executes it by interpreting the code into computer instructions, executed on computer hardware via the operating system. In the following chapters, how various types of Python statements are interpreted will be discussed in detail.

—---------------------------------------------

Como se muestra en la figura 1.2, **Python es un programa de aplicación** que se ejecuta sobre el sistema operativo, como un navegador o un editor de texto. Sin embargo, su función principal es ejecutar código Python genérico, en lugar de realizar una tarea específica, como hacen los navegadores y los editores de texto. IDLE es otro programa de aplicación que proporciona una interfaz interactiva para Python. En la parte superior de la figura está el código Python, que es ejecutado por la aplicación Python. Dada una pieza de código de Python, o un programa de Python, Python lo ejecuta al interpretar el código en instrucciones de computadora, ejecutadas en el hardware de la computadora a través del sistema operativo. En los siguientes capítulos, se discutirá en detalle cómo se interpretan varios tipos de instrucciones de Python.

—---------------------------------------------

1.2.2 The File System

Operating systems organize data in a hard drive as files. Each file contains a certain type of data. For example, a music track can be stored as an mp3 file, a video clip can be stored as an mpeg file, and a text document can be stored as a txt file. Application programs are also files. In Windows, for example, application programs such as Firefox are stored as executable (exe) files. Python programs are a special type of text document, stored as py files.

—--------------------------------------------

**Los sistemas operativos organizan los datos en un disco duro como archivos.** Cada archivo contiene un cierto tipo de datos. Por ejemplo, una pista de música se puede almacenar como un archivo mp3, un videoclip se puede almacenar como un archivo mpeg y un documento de texto se puede almacenar como un archivo txt. Los programas de aplicación también son archivos. En Windows, por ejemplo, los programas de aplicación como Firefox se almacenan como archivos ejecutables (exe). Los programas de Python son un tipo especial de documento de texto, almacenados como archivos py.

—--------------------------------------------

The type of a file, such as as mp3 and txt, is typically reflected by the last part of the file name, which is called the extension name. The extension name is separated from the main file name by a dot (.). For example, in the file name “readme.txt”, the main file name is “readme” and the extension name is “txt”, indicating that the corresponding file is a text file. In the file name “hello.py”, the main file name is “hello”, and the extension name is “py”, indicating that the file is a Python source file.

—--------------------------------------------

El tipo de archivo, como mp3 y txt, normalmente se refleja en la última parte del nombre del archivo, que se denomina **nombre de la extensión**. El nombre de la extensión está separado del nombre del archivo principal por un punto (.). Por ejemplo, en el nombre de archivo “readme.txt”, el nombre del archivo principal es “readme” y el nombre de la extensión es “txt”, lo que indica que el archivo correspondiente es un archivo de texto. En el nombre de archivo "hello.py", el nombre del archivo principal es "hello" y el nombre de la extensión es "py", lo que indica que el archivo es un archivo fuente de Python.

—--------------------------------------------

Files are organized using folders, which are containers of files and other folders. In a file system, folders are organized in a hierarchy. There is only one folder that is not contained by another folder, and it is called the root folder. Except for the root folder, each folder belongs to another folder, which is typically called the parent folder of the folder. Such a hierarchical structure is called a tree. An example is shown in Fig. 1.3, where the folders and files form an upside-down tree from the root folder.

—--------------------------------------------

**Los archivos se organizan mediante carpetas**, que son contenedores de archivos y otras carpetas. En un sistema de archivos, las carpetas se organizan en una jerarquía. Solo hay una carpeta que no está contenida en otra carpeta y se llama **carpeta raíz**. A excepción de la carpeta raíz, cada carpeta pertenece a otra carpeta, que normalmente se denomina carpeta principal de la carpeta. **Tal estructura jerárquica se llama árbol**. En la Fig. 1.3 se muestra un ejemplo, donde las carpetas y los archivos forman un árbol invertido desde la carpeta raíz.

—--------------------------------------------

In the tree structure of a file system, each file can be identified by the path from the root folder to the file. For example, in Fig. 1.3, file1 can be identified by the path root → folder1 → file1, and file3 can be identified by the path root → folder1 → folder4 → folder7 → file3. The use of path names to identify files avoids name clashes between different folders. For example, even if file6 is renamed as file1 in Fig. 1.3, it is still different from root → folder1 → file1.

—--------------------------------------------

**En la estructura de árbol de un sistema de archivos, cada archivo se puede identificar por la ruta desde la carpeta raíz hasta el archivo.** Por ejemplo, en la figura 1.3, el archivo 1 puede identificarse mediante la ruta raíz → carpeta 1 → archivo 1, y el archivo 3 puede identificarse mediante la ruta raíz → carpeta 1 → carpeta 4 → carpeta 7 → archivo 3. El uso de nombres de ruta para identificar archivos evita conflictos de nombres entre diferentes carpetas. Por ejemplo, incluso si el archivo6 se renombra como archivo1 en la Fig. 1.3, sigue siendo diferente de raíz → carpeta1 → archivo1.

—--------------------------------------------

In Windows, back slashes (\) are used as the path deliminator symbol. In Linux and Max OS, slashes (/) are used as the path deliminator. For example, the path for file6 in Fig. 1.3 is written as \ folder3\ folder6\ file6 in Windows, and /folder3/folder6/file6 in Linux. In all these cases the root folder is written implicitly as the start of the path.

—--------------------------------------------

—--------------------------------------------

In Windows, a hard drive can be split into several volumes, each having its own

tree structure of folders. Windows volumes are denoted by letters, starting from ‘C’,

followed by a colon (:). For example, ‘C:\folder\file1’ represents a file in the volume

‘C:’, and is different from the file ‘D:\folder1\file1’.

—--------------------------------------------

—--------------------------------------------

1.2.3 Text User Interfaces to Operating Systems

An operating system consists of many components, and one way to categorize operating system components is to put them into two types. The first type of OS components operates the computer hardware; it is called the kernel in Linux systems, for performing the “core” functionalities. The second type of OS components provides a user interface; it is called the shell in Linux systems.

—------------------------------------------

Un sistema operativo consta de muchos componentes, y una forma de categorizar los componentes del sistema operativo es dividirlos en dos tipos. El primer tipo de componentes del sistema operativo opera el hardware de la computadora; se llama **kernel** en los sistemas Linux, para realizar las funcionalidades "básicas". El segundo tipo de componentes del sistema operativo proporciona una interfaz de usuario; se llama **shell** en los sistemas Linux.

—------------------------------------------

OS user interfaces have evolved from text user interface to graphical user interface (GUI). A text user interface is also called a text console (Fig. 1.4). It works by repeatedly displaying a prompt message to the user and waits for a line of user command. After receiving a command, it executes the command, displays results and feedbacks in text format, and prompts for a next command. On the other hand, a GUI (Fig. 1.5) displays information in graphical windows and message boxes, and receives user command via buttons, menu items and dialog boxes.

—------------------------------------------

Las interfaces de usuario del sistema operativo han evolucionado de la interfaz de usuario de texto a la **interfaz gráfica de usuario (GUI)**. Una interfaz de usuario de texto también se denomina consola de texto (Fig. 1.4). Funciona mostrando repetidamente un mensaje de aviso al usuario y espera una línea de comando del usuario. Después de recibir un comando, lo ejecuta, muestra los resultados y los comentarios en formato de texto y solicita el siguiente comando. Por otro lado, una GUI (Fig. 1.5) muestra información en ventanas gráficas y cuadros de mensaje, y recibe comandos del usuario a través de botones, elementos de menú y cuadros de diálogo.

—------------------------------------------

Compared with a text console, a GUI is more intuitive to use, not requiring one to remember the names of text commands. However, a text console is often more convenient for executing complex commands with many parameters, because such advanced commands may not have GUI at all, and even if they do, entering a row of parameters in text can be faster than choosing and entering into text boxes in a large dialog window. With respect to programming, text input and output can be much easier to program compared to GUI. As a result, text is the default input and output mechanism for many programming languages, including Python, and basic knowledge of the text console is highly useful for learning programming.

—------------------------------------------

En comparación con una consola de texto, una GUI es más intuitiva de usar y no requiere que uno recuerde los nombres de los comandos de texto. Sin embargo, una **consola de texto a menudo es más conveniente para ejecutar comandos complejos con muchos parámetros**, porque estos comandos avanzados pueden no tener GUI en absoluto, e incluso si la tienen, ingresar una fila de parámetros en el texto puede ser más rápido que elegir e ingresar texto. cuadros en una ventana de diálogo grande. Con respecto a la programación, la entrada y salida de texto puede ser mucho más fácil de programar en comparación con la GUI. Como resultado, el texto es el mecanismo de entrada y salida predeterminado para muchos lenguajes de programación, incluido Python, y el conocimiento básico de la consola de texto es muy útil para aprender a programar.

—------------------------------------------

In Windows, Linux and Mac OS, the text console is a program. In Windows, the program is named ‘Command Line’; in Linux, the program is named ‘Terminal’ or ‘Console’; in Mac OS, the program is named ‘Terminal’ (Fig. 1.4). Because the text console commands of Linux and Mac OS are mostly similar, they are not introduced separately in this book.

—---------------------------------

En Windows, Linux y Mac OS, **la consola de texto es un programa**. En Windows, el programa se llama 'Línea de comandos'; en Linux, el programa se llama **'Terminal' o 'Consola'**; en Mac OS, el programa se llama 'Terminal' (Fig. 1.4). Debido a que los comandos de la consola de texto de Linux y Mac OS son en su mayoría similares, no se presentan por separado en este libro.

—---------------------------------

After being launched, Command Line can show the following message:

C :\ Users \ yue\_zhang >

After being launched, Terminal can show the following message:

Zhangs - MacBook - Pro :~ y u e \_ z h a n g $

In the examples above, ‘C:\Users\yue\_zhang>’ and ‘Zhangs-MacBook-Pro:

∼ yue\_zhang$’ are the prompt messages, after which a command can be entered.

Each text command finishes with a new line (the Enter key).

In a text console, all commands are entered in a current working folder, which is a folder in the file system. By default, all the commands that are entered into the text console operates on the current working folder. Without a full path being specified, all file names entered into the text console also refer to files in the current working folder by default. The current working folder is typically reflected by the prompt message. In Windows, the following command shows its full path:

—-----------------------------------------

En una consola de texto, todos los comandos se ingresan en una carpeta de trabajo actual, que es una carpeta en el sistema de archivos. **De forma predeterminada, todos los comandos que se ingresan en la consola de texto funcionan en la carpeta de trabajo actual.** Sin especificar una ruta completa, todos los nombres de archivo ingresados en la consola de texto también se refieren a archivos en la carpeta de trabajo actual de forma predeterminada. La carpeta de trabajo actual normalmente se refleja en el mensaje de aviso. En Windows, el siguiente comando muestra su ruta completa:

—-----------------------------------------

C :\ Users \ yue\_zhang > **cd**

C :\ U s e r s \ y u e \_ z h a n g

In Linux and Mac OS, the following command shows its full path

Zhangs - MacBook - Pro :~ y u e \_ z h a n g $ **pwd**

/ U s e r s / y u e \_ z h a n g

By default, C:\Users\yue\_zhang and /Users/yue\_zhang are the home folder of

the user yue\_zhang under Windows and Mac OS, respectively. Under Linux, the home

folder can be /home/yue\_zhang.

One can list out the files and folders contained in the current working folder. In

Windows, the command is dir:

C :\ Users \ yue\_zhang > dir

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > .

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > ..

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > C o n t a c t s

2 0 1 5 / 0 3 / 3 0 1 4 : 4 9 < DIR > D e s k t o p

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o c u m e n t s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o w n l o a d s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > F a v o r i t e s

In Linux and Mac OS, the command is **ls**:

Zhangs - MacBook - Pro :~ y u e \_ z h a n g $ ls

A p p l i c a t i o n s D o w n l o a d s M o v i e s P u b l i c

octave - core

D e s k t o p D r o p b o x M u s i c R e s e a r c h

D o c u m e n t s L i b r a r y P i c t u r e s T e a c h i n g

**All the commands are by default executed in the current working folder**.

For example, the command ‘mkdir <folder\_name>’ creates a new folder named

‘<folder\_name>’ in the current folder. An example in Windows is shown below.

C :\ Users \ yue\_zhang > mkdir n e w f o l d e r

C :\ Users \ yue\_zhang > dir

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > .

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > ..

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > C o n t a c t s

2 0 1 5 / 0 3 / 3 0 1 4 : 4 9 < DIR > D e s k t o p

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o c u m e n t s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o w n l o a d s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > F a v o r i t e s

2 0 1 5 / 0 3 / 3 0 1 5 : 2 6 < DIR > n e w f o l d e r

The command ‘rmdir <folder\_name>’ removes the empty folder ‘<folder\_

name>’ from the current folder:

C :\ Users \ yue\_zhang > rmdir n e w f o l d e r

C :\ Users \ yue\_zhang > dir

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > .

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > ..

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > C o n t a c t s

2 0 1 5 / 0 3 / 3 0 1 4 : 4 9 < DIR > D e s k t o p

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o c u m e n t s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o w n l o a d s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > F a v o r i t e s

The commands **mkdir** and **rmdir** apply to Linux and Mac OS also.

One can change the current working folder by using the command ‘**cd**

<folder\_name>’, which enters the folder <folder\_name>. For example,

C :\ Users \ yue\_zhang > dir

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > .

2 0 1 5 / 0 3 / 3 0 1 5 : 3 0 < DIR > ..

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > C o n t a c t s

2 0 1 5 / 0 3 / 3 0 1 4 : 4 9 < DIR > D e s k t o p

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o c u m e n t s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > D o w n l o a d s

2 0 1 5 / 0 3 / 1 2 1 0 : 1 5 < DIR > F a v o r i t e s

C :\ Users \ yue\_zhang > cd C o n t a c t s

C :\ U s e r s \ y u e \_ z h a n g \ Contacts >

Two special folder names are ‘.’ and ‘..’, which represent the current working folder and its parent folder, respectively. ‘cd .’ does nothing, for the working folder is changed to the same folder. ‘cd ..’ enters the parent folder, the folder one level up towards the root.

—----------------------------------------------

**Dos nombres de carpeta especiales son '.' y '..', que representan la carpeta de trabajo actual y su carpeta principal, respectivamente**. 'cd .' no hace nada, ya que la carpeta de trabajo se cambia a la misma carpeta. 'cd ..' ingresa a la carpeta principal, la carpeta un nivel más arriba hacia la raíz.

—----------------------------------------------

C :\ Users \ yue\_zhang > cd

C :\ U s e r s \ y u e \_ z h a n g

C :\ Users \ yue\_zhang > cd ..

C :\ Users >

In all the examples above, <folder\_name> is a folder name in the current working

folder. This type of path is called a **relative path**. On the other hand, **absolute paths**,

which specify a folder from the root, can also be used for <folder\_name>. For

example,

C :\ Users \ yue\_zhang > cd

C :\ U s e r s \ y u e \_ z h a n g

C :\ Users \ yue\_zhang > cd \ Users \ y u e \_ z h a n g \ D e s k t o p

C :\ U s e r s \ y u e \_ z h a n g \ Desktop >

The syntax of the cd <folder> command is the same in Linux and Mac OS as in

Windows. However, if cd is used without specifying <folder>, the functionality is

showing the current working directory on Windows, while entering the home folder

on Linux and Mac OS.

1.2.4 The Python Application Program

Python programs are executed by the Python application program, or Python in short, which is a software program like TextEdit and Firefox. Python can be downloaded from the official Python website www.python.org, and installed into a computer. There are two commonly used dialects of Python: Python 2 and Python 3. This book uses Python 2, in which most existing Python programs are written. However, the underlying mechanisms are mostly the same for both Python 2 and Python 3, and they are only mildly different in syntax for certain functionalities.

—----------------------------------------

**Los programas de Python son ejecutados por el programa de aplicación de Python**, o Python para abreviar, que es un programa de software como TextEdit y Firefox. Python puede descargarse del sitio web oficial de Python **www.python.org** e instalarse en una computadora. Hay dos dialectos de Python que se usan comúnmente: Python 2 y Python 3. Este libro usa Python 2, en el que están escritos la mayoría de los programas de Python existentes. Sin embargo, los mecanismos subyacentes son en su mayoría los mismos para Python 2 y Python 3, y solo difieren levemente en la sintaxis para ciertas funcionalidades.

—----------------------------------------

The Python installer program allows the specification of a path, in which Python is installed. By default, the folder is C:\Python2.7 on Windows, where 2.7 is the version number of Python. After installation, the executable Python application is the file:

—----------------------------------------

El programa de instalación de Python permite la especificación de una ruta en la que se instala Python. De forma predeterminada, la carpeta es C:\Python2.7 en Windows, donde 2.7 es el número de versión de Python. Después de la instalación, la aplicación Python ejecutable es el archivo:

—----------------------------------------

C :\ P y t h o n 2 .7\ p y t h o n . exe

On Linux and Mac OS, Python is typically pre-installed, and can be found at

**/ usr / bin / p y t h o n 2 .7**

Python can be executed by setting the current working folder to the folder containing Python

c :\ Users \ yue\_zhan g > cd \ P y t h o n 2 .7

C :\ P y t h o n 2 .7 > p y t h o n

P y t h o n 2 . 7 . 1 ( r 2 7 1 :86832 , Jul 31 2011 , 1 9 : 3 0 : 5 3 ) [ MSC

v . 1 5 0 0 32 bit ( I n t e l ) ] on w i n 3 2

Type " help " , " c o p y r i g h t " , " c r e d i t s " or " l i c e n s e " for

more i n f o r m a t i o n .

> > >

By typing ‘python’, Windows executes the file python.exe in the current working

folder, which leads to the interactive execution mode of Python.

On Linux and Mac OS, Python can be executed by typing ‘./Python2.7’ in the

current working folder.

Zhangs - MacBook - Pro :~ y u e \_ z h a n g $ cd / usr / bin

Zhangs - MacBook - Pro :~ y u e \_ z h a n g $ ./ p y t h o n 2 .7

P y t h o n 2 . 7 . 1 ( r 2 7 1 :86832 , Jul 31 2011 , 1 9 : 3 0 : 5 3 )

[ GCC 4.2.1 ( Based on Apple Inc . build 5658) ( LLVM

b u i l d 2 3 3 5 . 1 5 . 0 0 ) ] on l i n u x 2

Type " help " , " c o p y r i g h t " , " c r e d i t s " or " l i c e n s e " for

more i n f o r m a t i o n .

> > >

**Note that ‘./’ must be used when executing a program in the current working**

**folder in Linux and Mac OS. To exit the interactive Python application, press and**

**hold the [Ctrl] key and press the D key (i.e. press Ctrl-D).**

( c o n t i n u e d from a b o v e )

> > > ^ D

Zhangs - MacBook - Pro :~ y u e \_ z h a n g $

1.2.5 Python and Environment Variables

The content of this section is not typically necessary to know, but can be useful when

customizing Python.

Environment variables are a set of settings that an operating system uses to configure itself. For Windows, the current environment variables can be listed and modified by clicking Start → Control Panel → System → Advanced system settings. In the Advanced tab, select Environment Variables, as shown in Fig. 1.6.

—-----------------------------------------

**Las variables de entorno** son un conjunto de valores que utiliza un sistema operativo para configurarse a sí mismo. Para Windows, las variables de entorno actuales se pueden enumerar y modificar haciendo clic en Inicio → Panel de control → Sistema → Configuración avanzada del sistema. En la pestaña Avanzado, seleccione Variables de entorno, como se muestra en la Fig. 1.6.

—-----------------------------------------

For Linux and Mac OS, the current environment variables can be viewed and

modified by editing the file **$PATH**.

The PATH environment variable stores a list of paths from which the operating system searches for an executable program automatically. If the specified executable is not in the current working folder (Windows), or the path (e.g. ‘./’; ‘/Desktop/’) is not specified (Linux and Mac OS), the operating system will search the paths in the PATH environment variable for the executable.

—-----------------------------------------

La variable de entorno PATH almacena una lista de rutas desde las que el sistema operativo busca automáticamente un programa ejecutable. Si el ejecutable especificado no está en la carpeta de trabajo actual (Windows), o la ruta (por ejemplo, './'; '/Escritorio/') no está especificada (Linux y Mac OS), el sistema operativo buscará las rutas en el Variable de entorno PATH para el ejecutable.

—-----------------------------------------

Python is typically added to the PATH environment variable when installed. There- fore, it should be possible to execute python at any path. In the rare cases when the operating system cannot find the python executable program, the path in whit Python is installed (e.g. C:\Python 2.7) can be added to the list of folders in the PATH environment variable. A restart of the operating system may be necessary for the change to take effect.

—-----------------------------------------

**Python generalmente se agrega a la variable de entorno PATH cuando se instala.** Por lo tanto, debería ser posible ejecutar python en cualquier ruta. En los casos excepcionales en los que el sistema operativo no puede encontrar el programa ejecutable de Python, la ruta en la que está instalado Python (por ejemplo, C:\Python 2.7) se puede agregar a la lista de carpetas en la **variable de entorno PATH**. Es posible que sea necesario reiniciar el sistema operativo para que el cambio surta efecto.

—----------------------------------------------

The PYTHON\_PATH environment variable is a second environment variable relevant to Python. It also stores a list of paths that are useful to Python specifically, rather than for the operating system. In particular, when Python loads library modules (Chap. 3), it will search the paths in PYTHON\_PATH for them. On Windows, PYTHON\_PATH can be set by following Fig. 1.6. On Linux and Mac OS, it can be set by modifying the file .bash\_profile under the home folder, adding the line

—----------------------------------------------

**La variable de entorno PYTHON\_PATH es una segunda variable de entorno relevante para Python.** También almacena una lista de rutas que son útiles específicamente para Python, en lugar de para el sistema operativo. En particular, cuando Python carga módulos de biblioteca (Capítulo 3), los buscará en las rutas de PYTHON\_PATH. En Windows, PYTHON\_PATH se puede configurar siguiendo la Fig. 1.6. **En Linux y Mac OS, se puede configurar modificando el archivo .bash\_profile en la carpeta de inicio, agregando la línea**

—----------------------------------------------

e x p o r t P Y T H O N \_ P A T H = $ P Y T H O N \_ P A T H : p a t h 1 : p a t h 2 : p a t h 3

where path1, path2 and path3 are module paths.

Exercises

1. Which of the following can be achieved using Python programming?

¿Cuál de los siguientes se puede lograr usando la programación de Python?

(a) Mathmatical calculations;

(b) A Windows GUI application;

(c) Controlling a robotic arm;

(d) Web crawling;rastreo web;

(e) A machine-leaning system that can fly a helicopter;

(f) Simulation of large-scale experiments in physics;

(g) A game in Android

(h) A operating system from scratch.

2. What are the three main components of a computer?

¿Cuáles son los tres componentes principales de una computadora?

3. List three programming languages and compare their differences. Why are there

many programming languages?

Enumere tres lenguajes de programación y compare sus diferencias. ¿Por qué hay tantos lenguajes de programación?

4. How can text user interfaces be launched on Windows, Linux and Mac OS?

¿Cómo se pueden iniciar las interfaces de usuario de texto en Windows, Linux y Mac OS?

5. Perform the following in a text user interface.

Realice lo siguiente en una interfaz de usuario de texto

(a) Show the current working folder;

(b) Enter the home folder;

(c) Enter the Desktop folder;

(d) Create a new folder under the name New under the Desktop folder, and verify

that it appears on the graphical desktop;

6. Launch the python program, and then exit it