

Modul 3

Introducing Python

1. Scripting vs. a Programming Language

Python is a scripting language. Some may quibble over whether it's a programming language or a scripting language, but to keep the strict technocrats happy, we'll call it a scripting language. A scripting language differs from a true programming language in a few ways. As you read the following comparisons, take note of the italics:

- Programming languages are compiled, unlike scripting languages. Common languages like C, C++, and Java must be compiled by a compiler. The compilation process results in a file of machine code, unreadable by humans, that the computer can read and follow. When you write a program in C and compile it, the resulting .o file is what is read by the computer. One of the side effects/results of this is that programming languages may produce faster programs—both because the compilation only happens once and because the compiler often optimizes the code during the compilation process, making it faster than it would be as originally written. Scripting languages, on the other hand, are read, interpreted, and acted upon each time you run them. They don't produce a compiled file, and the instructions are followed exactly as written. If you write sloppy code, you get sloppy results. For this reason, scripting languages can result in slower programs.
- Programming/compiled languages most often run directly on top of the hardware on which they are written. When you write and compile a program in C++, the resulting code is executed directly by the processor on your desktop machine. Scripting languages most often run “inside” another program—one that takes care of the compiling step just mentioned. PHP, a common scripting language, runs inside the PHP scripting engine. Bash scripts run inside the bash shell, which you were introduced to in the previous chapter.
- Programming languages tend to be more complex and difficult to learn. Scripting languages can be more readable, are less syntax-strict, and are less intimidating

to nonprogrammers. For this reason alone, scripting languages are often taught in introductory programming courses in schools, and students are not introduced to stricter languages like C or Java until they have mastered the basics of programming.

2. Running Python Using IDLE

The IDLE interpreter is a sort of “sandbox” where you can work with Python interactively without having to write whole scripts to see what they do. The name IDLE stands for “Integrated DeveLopment Environment,” but it also pays homage to Eric Idle, one of the founding members of the British comedy group Monty Python. (See the sidebar “Get Me a Shrubbery!”) Because it’s the most user-friendly way of trying out code, let’s use IDLE first. Double-click the icon on your desktop (shown in Figure 3-1), and you should be greeted by a window like Figure 3-2.



Fig. 3.1 Icon Python

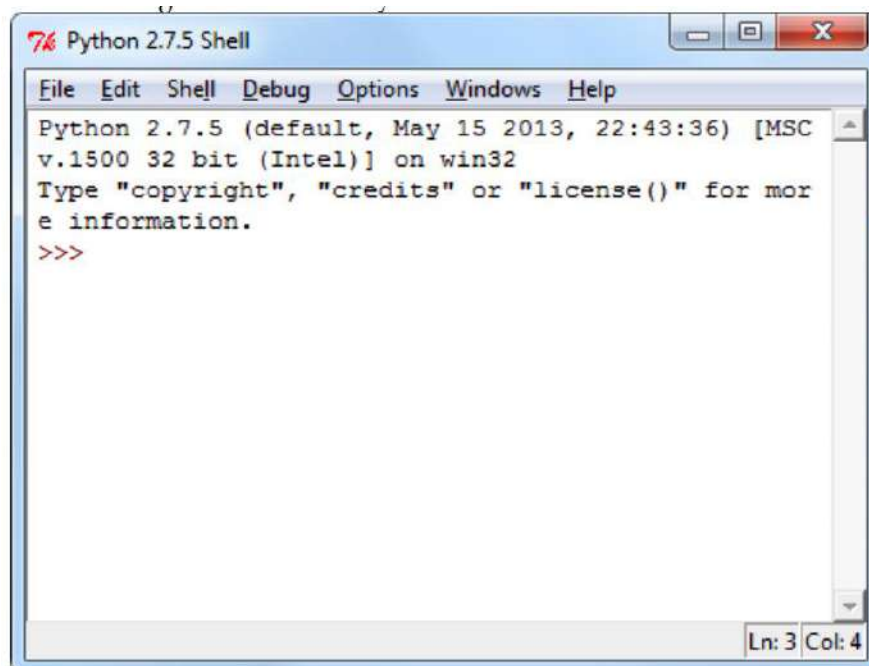


Fig. 3.2 Worksheet Python

JOB SHEET 3

Lakukan penggunaan perintah dasar pada Python pada software Raspberri Pi. Jelaskan fungsi dari perintah dasar yang digunakan.

LAPORAN HASIL PERCOBAAN: