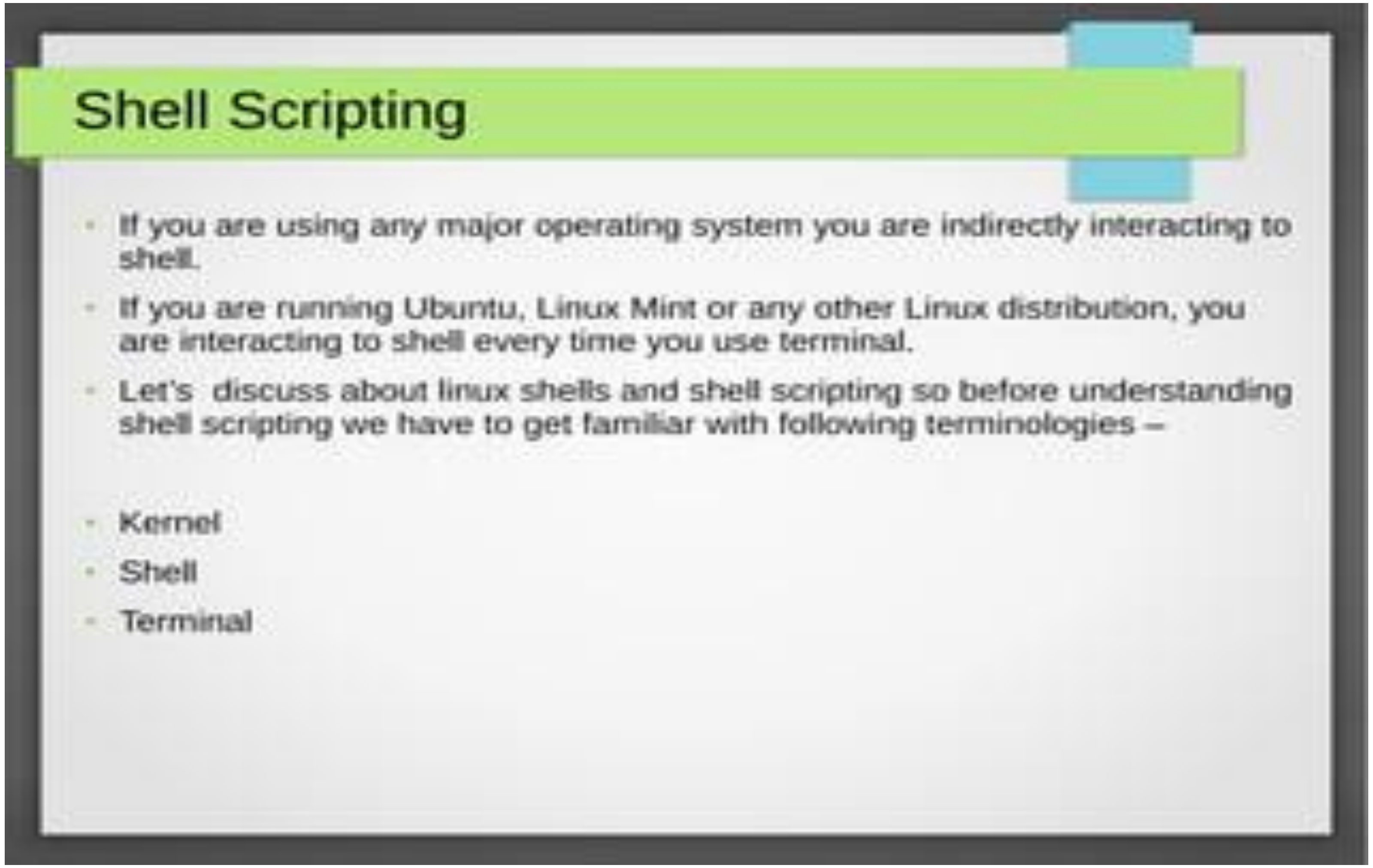
Basic of Shell Scripting

Introduction



Shell Scripting

- If you are using any major operating system you are indirectly interacting to shell.
- If you are running Ubuntu, Linux Mint or any other Linux distribution, you are interacting to shell every time you use terminal.
- Let's discuss about linux shells and shell scripting so before understanding shell scripting we have to get familiar with following terminologies –

- Kernel
- Shell
- Terminal

Basic Terminology

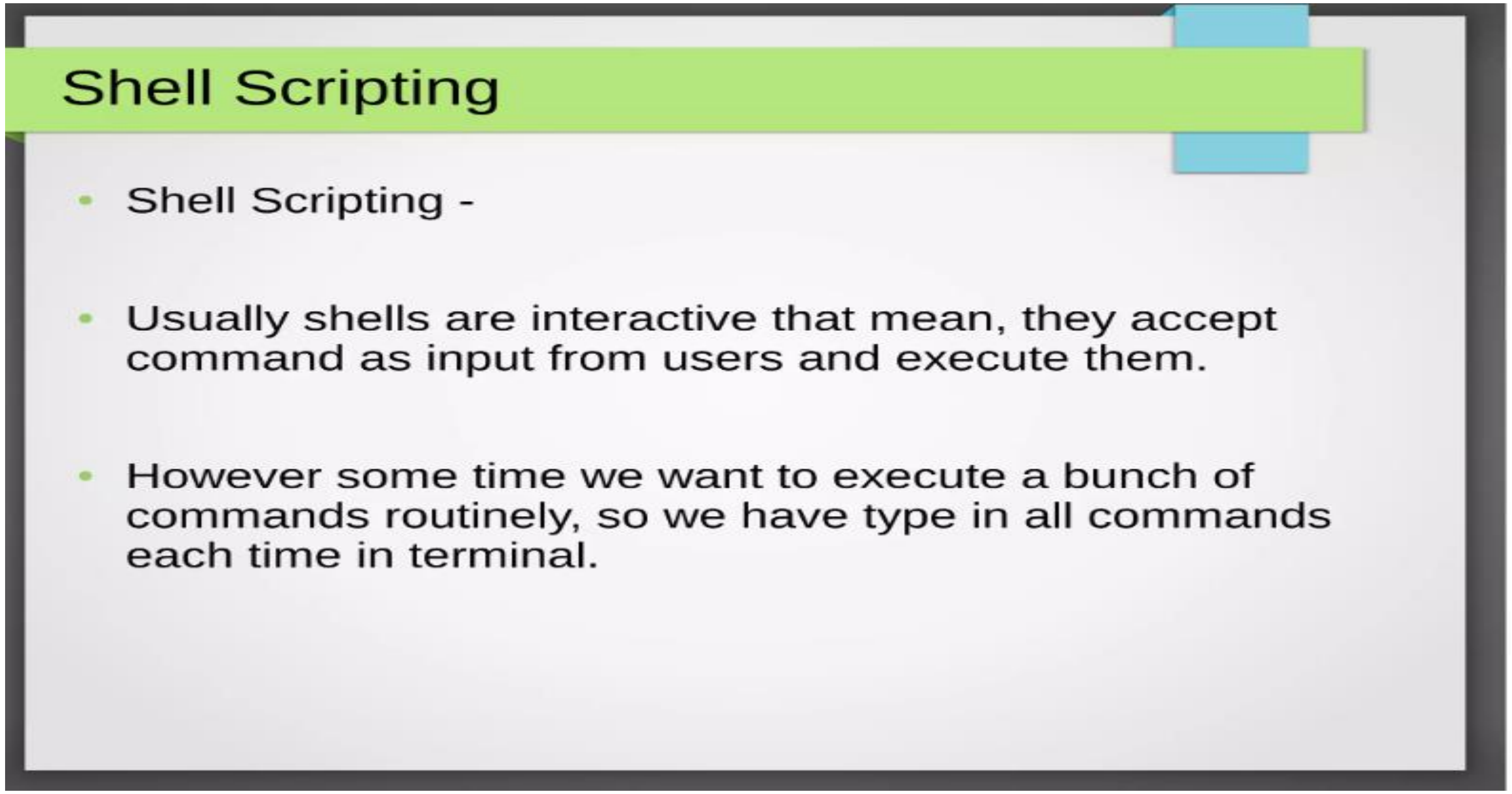
- Shell is a command-line interface to run commands and shell scripts. Shells come in a variety of flavors, much as operating systems come in a variety of flavors.
- A terminal, also known as a shell, is a program that allows users to interact with their operating system through a command-line interface.

Note:

- Shell is an interface between the user and the operating system.
- sh implements the shell interface.
- bash is a superset of sh.

Introduction

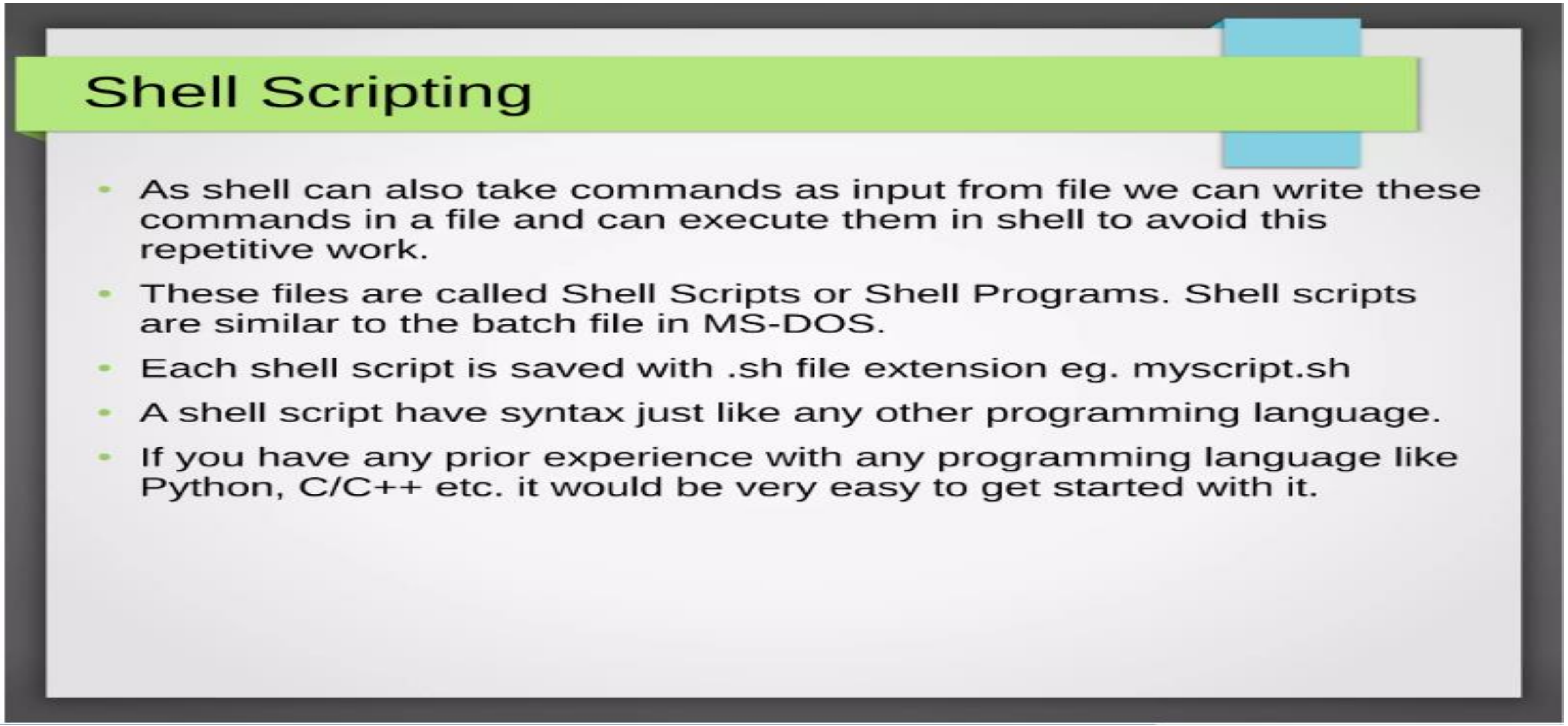
Now a shell can accept a number of commands through a file, known as a shell script. A shell script contains a number of commands that are executed by a shell.



Shell Scripting

- Shell Scripting -
- Usually shells are interactive that mean, they accept command as input from users and execute them.
- However some time we want to execute a bunch of commands routinely, so we have type in all commands each time in terminal.

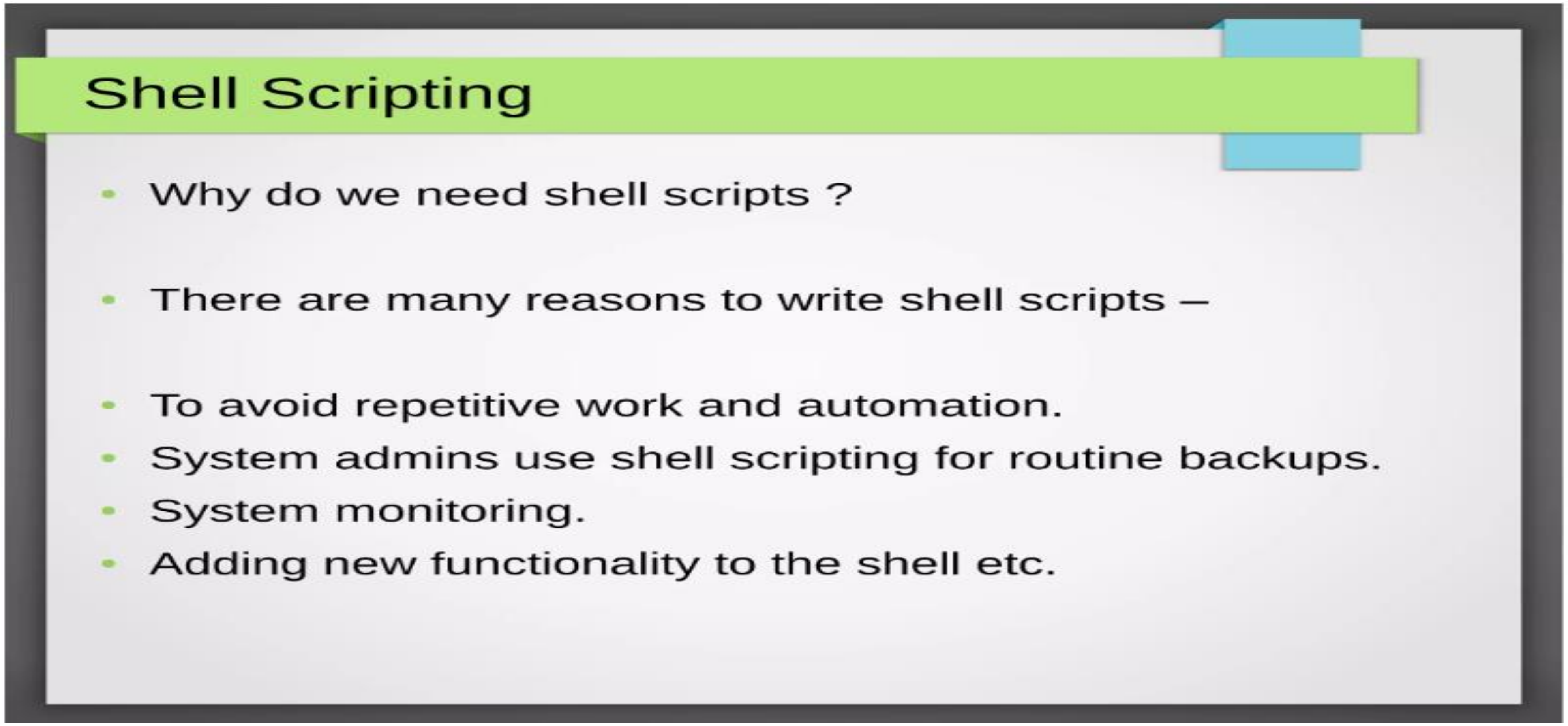
Introduction



Shell Scripting

- As shell can also take commands as input from file we can write these commands in a file and can execute them in shell to avoid this repetitive work.
- These files are called Shell Scripts or Shell Programs. Shell scripts are similar to the batch file in MS-DOS.
- Each shell script is saved with .sh file extension eg. myscript.sh
- A shell script have syntax just like any other programming language.
- If you have any prior experience with any programming language like Python, C/C++ etc. it would be very easy to get started with it.

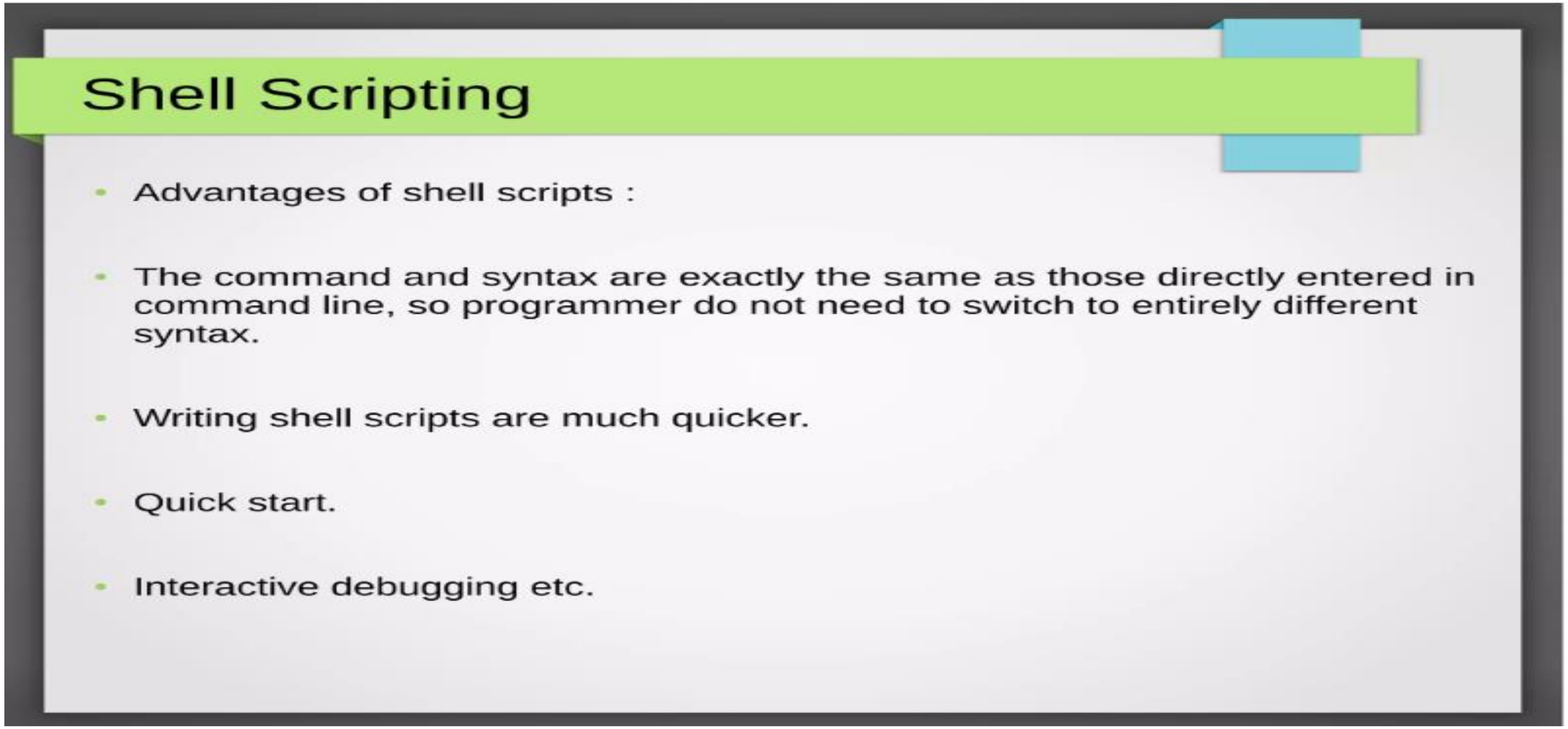
Introduction



Shell Scripting

- Why do we need shell scripts ?
- There are many reasons to write shell scripts –
- To avoid repetitive work and automation.
- System admins use shell scripting for routine backups.
- System monitoring.
- Adding new functionality to the shell etc.

Introduction



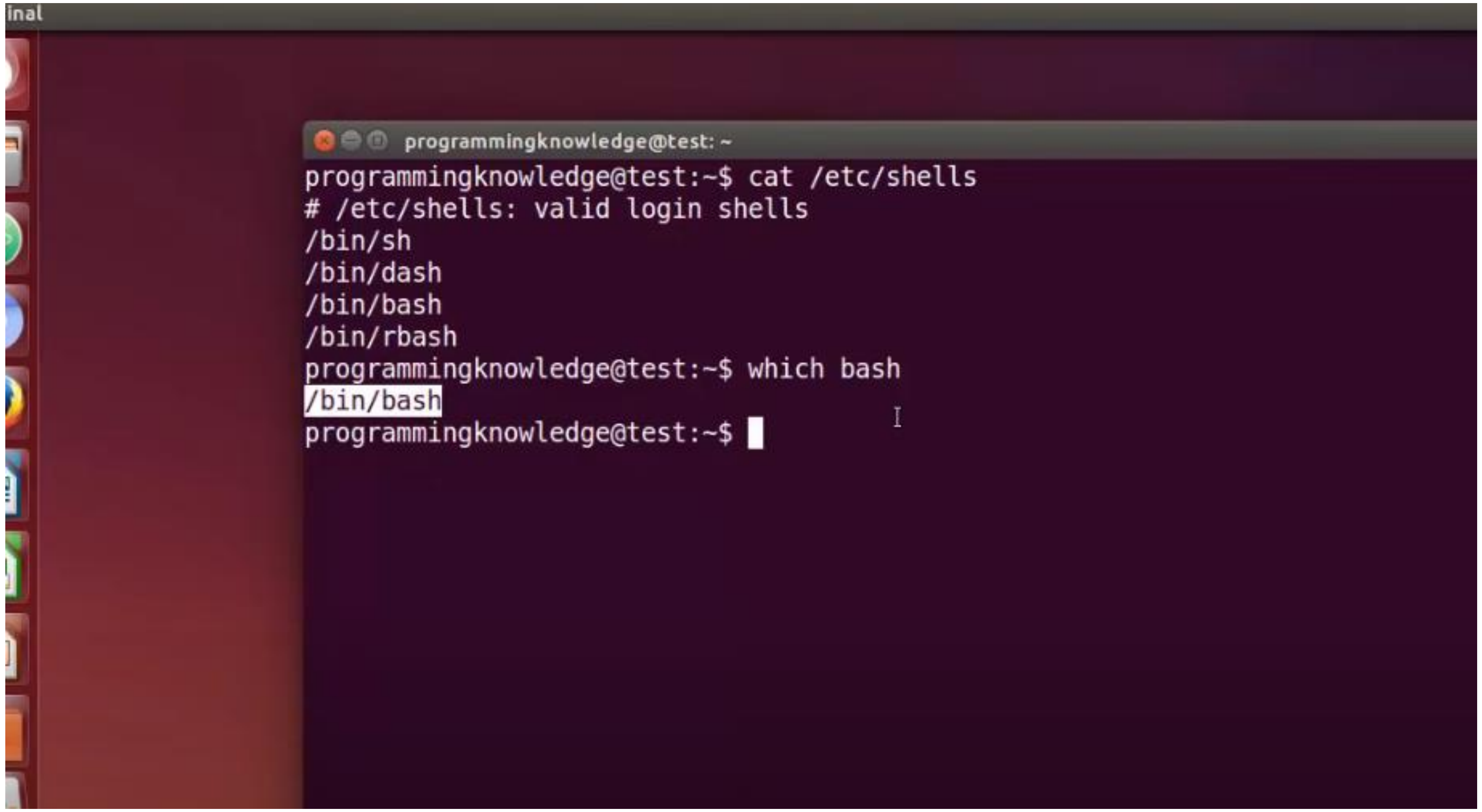
Shell Scripting

- Advantages of shell scripts :
- The command and syntax are exactly the same as those directly entered in command line, so programmer do not need to switch to entirely different syntax.
- Writing shell scripts are much quicker.
- Quick start.
- Interactive debugging etc.

Introduction to Bash

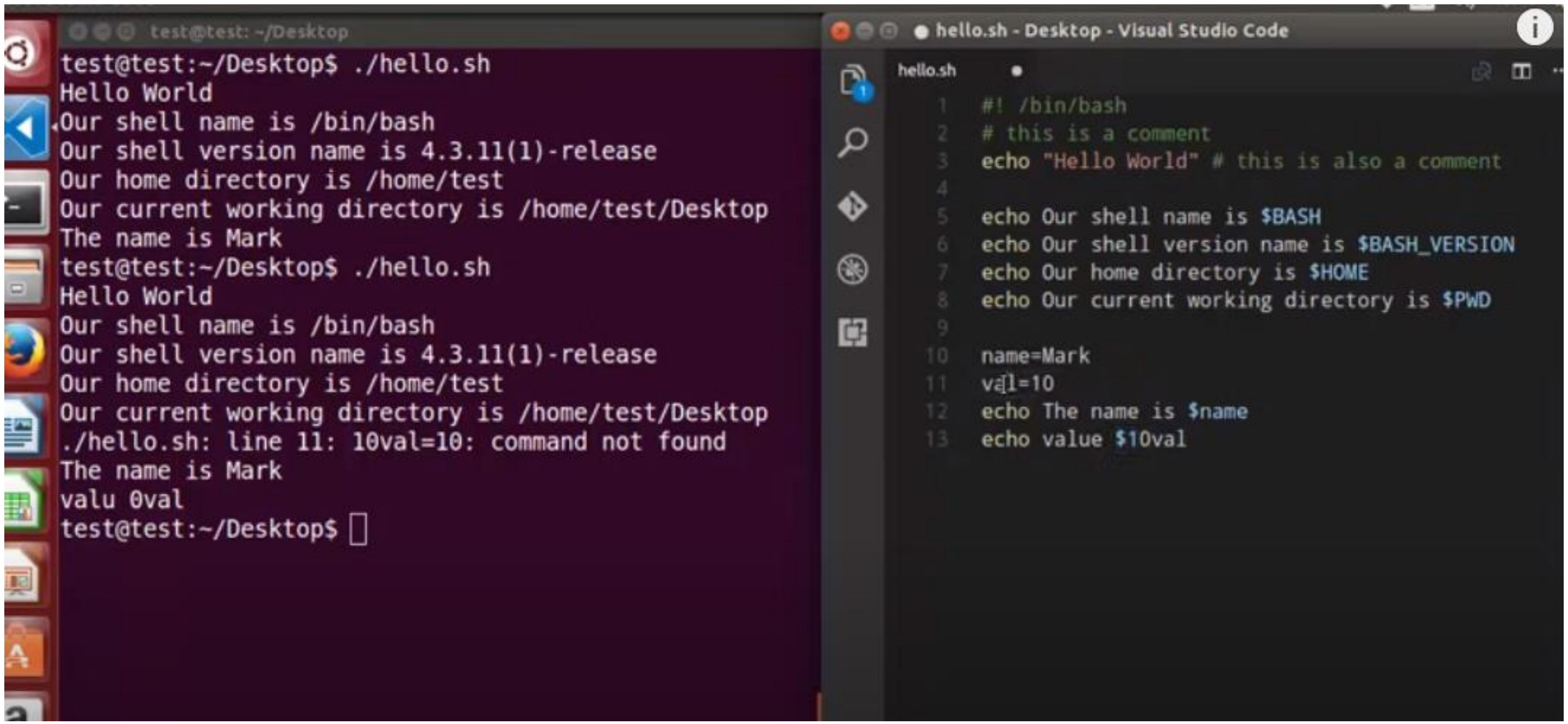
- The Linux Bash is also known as **'Bourne-again Shell**.' It is a **command language interpreter** for the Linux based system. It is a replacement of Bourne shell (sh). It was developed under the GNU Project and written by **Brian Fox**. Nowadays, Bash is the default user shell of most of the Linux distributions.
- The Linux/Unix shell allows us to interact with the Linux system through the commands. It let us invoke an executable file to create a running process. Moreover, it also allows us to interact with the Linux file system. It is designed in such a way that we can perform all the Linux operations through Bash.
- The Bash is a **command language interpreter** as well as a **programming language**. It supports **variables, functions, and flow control**, like other programming languages. It can also read and execute the commands from a file, which is called a **shell script.**

Introduction



```
programmingknowledge@test:~$ cat /etc/shells
# /etc/shells: valid login shells
/bin/sh
/bin/dash
/bin/bash
/bin/rbash
programmingknowledge@test:~$ which bash
/bin/bash
programmingknowledge@test:~$ 
```

1st Shell Scripting Code



```
test@test:~/Desktop$ ./hello.sh
Hello World
Our shell name is /bin/bash
Our shell version name is 4.3.11(1)-release
Our home directory is /home/test
Our current working directory is /home/test/Desktop
The name is Mark
test@test:~/Desktop$ ./hello.sh
Hello World
Our shell name is /bin/bash
Our shell version name is 4.3.11(1)-release
Our home directory is /home/test
Our current working directory is /home/test/Desktop
./hello.sh: line 11: 10val=10: command not found
The name is Mark
valu 0val
test@test:~/Desktop$
```

```
#! /bin/bash
# this is a comment
echo "Hello World" # this is also a comment

echo Our shell name is $BASH
echo Our shell version name is $BASH_VERSION
echo Our home directory is $HOME
echo Our current working directory is $PWD

name=Mark
val=10
echo The name is $name
echo value $10val
```

Explanation

Adding the Shebang

- Bash scripts start with a shebang. Shebang is a combination of bash # and bang ! followed by the bash shell path. This is the first line of the script. Shebang tells the shell to execute it via bash shell. Shebang is simply an absolute path to the bash interpreter.
- You can find your bash shell path using the command: which bash

Running the script

You can run the script using any of the mentioned methods:

- sh run_all.sh
- bash run_all.sh
- ./run_all.sh

2nd Example

- To read the output from the user.

```
echo "What is your name?"
read PERSON
echo "Hello, $PERSON"
```