

Exercises

Computer Tech 2018

Tom Browder [tom.browder@gmail.com]

Northwest Florida Linux User Group [nwflug.org]

2018-01-27

Exercise 1

Using the *gedit* editor (or the editor of your choice), write a program that prints “Hello, World!” to the screen (stdout) when executed. Save and name the source files “hello-world.*” where ‘*’ is the conventional suffix (file extension) for the language.

Write it in one or more of the following languages:

- **1 Perl6**
- **2 Bash**
- **3 Python**
- **4 C**
- **5 PostScript**

Note that the first three are commonly referred to as *scripting* languages), *C* is a *compiled* language, and *PostScript* is a *markup* language.

Exercise 2

Using either the command line (which uses the bash shell by default) or one of the languages above, determine how many files are known to the system.

Exercise 3

Using one of the scripting languages, write a script that will execute the other programs (except the PostScript one).

Solutions are shown as entries in a command-line (shell) window. The '\$' is the standard prompt for a user and shows the shell program is awaiting input from the user.

Lines following user entries (commands) are the shell's echoed responses from the user's command, and the end of the command response is normally indicated by the '\$' being shown again as the shell awaits the user's next command.

For brevity we do not show that following prompt in the solutions, but you should see it as you do the exercises.

Solutions: 1.1 - Perl 6

Exercise 1: Write a program to show “Hello, World!”

Perl 6

source file name: **hello-world.p6**

source file contents:

```
say "Hello, World!";
```

execute the source file

```
$ perl6 hello-world.p6  
Hello, World!
```

Solutions: 1.2 - Bash

Exercise 1: Write a program to show “Hello, World!”

Bash

source file name: **hello-world.sh**

source file contents:

```
echo Hello, World!
```

execute the source file

```
$ bash hello-world.sh  
Hello, World!
```

Solutions: 1.3 - Python

Exercise 1: Write a program to show “Hello, World!”

Python

source file name: **hello-world.py**

source file contents:

```
print "Hello, World!"
```

execute the source file

```
$ python hello-world.py  
Hello, World!
```


Exercise 1: Write a program to show “Hello, World!”

C

source file name: **hello-world.c**

source file contents:

```
#include <stdio.h>
int main() {
    printf("Hello, World!\n");
    return 0; /* shows a successful exit code to the caller */
}
```

Solutions: 1.4 - C (cont.)

create the executable program file

```
$ gcc hello-world.c -o hello-world
```

execute the program file

```
$ ./hello-world  
Hello, World!
```

Solutions: 1.5 - PostScript

Exercise 1: Write a program to show “Hello, World!”

PostScript

source file name: **hello-world.ps**

source file contents:

```
%!
% the unit of length is the PS point: 72 points per inch
/in {72 mul} def % a convenience definition
% define the font and size
/Times-Roman findfont 50 scalefont setfont
% origin is the lower-left corner of the page
% positions are in x,y pairs
%   x is the horizontal scale increasing to the right,
%   y is the vertical scale increasing up
```

PostScript source file continued:

```
% move to the desired lower-left point of the text
2.5 in 5 in moveto
% paint the text on the page
(Hello, World!) show
% show the entire, finished page
(Hello, World!) show
showpage
```

Solutions: 1.5 - PostScript (cont.)

create the pdf version

```
$ ps2pdf hello-world.ps
```

view it with program *evince*

```
$ evince hello-world.pdf
```

Exercise 2: Determine how many files are on this computer

```
$ locate * > files.list ; wc --lines files.txt  
640000
```

Exercise 3: Write an overarching script in one of the scripting languages

Bash

source file name: **run-all.sh**

source file contents:

```
bash hello-world.sh
perl6 hello-world.p6
python hello-world.py
./hello-world
```

execute the file

```
$ bash run-all.sh  
Hello, World!  
Hello, World!  
Hello, World!  
Hello, World!
```


Exercise 3: Write an overarching script in one of the scripting languages

Perl 6

source file name: **run-all.p6**

source file contents:

```
# 'shell' is not recommended in some instances
# it is safe to use for regular characters
# use 'run' otherwise
shell "bash hello-world.sh";
shell "perl6 hello-world.p6";
shell "python hello-world.py";
shell "./hello-world";
```

execute the source file

```
$ perl6 run-all.p6  
Hello, World!  
Hello, World!  
Hello, World!  
Hello, World!
```