Python's Hello World

K. S. Ooi
Foundation in Science
Faculty of Health and Life Sciences
INTI International University
Persiaran Perdana BBN, Putra Nilai,
71800 Nilai, Negeri Sembilan, Malaysia
E-mail: kuansan.ooi@newinti.edu.my
dr.k.s.ooi@gmail.com

Abstract

Compared to C and Java, the Python's Hello World is a one liner. If we look beyond the line, there is more than meets the eye. Among the possible cursory detour one can take are: object/type, string multiplication, built-in string functions, for statement, list comprehension, mumbo jumbo, and even Hy. We dare you to explore.

Keywords: Python 3, Hello World program, Hy
Date: Dec 25, 2020

1. Hello, World!

It has become a tradition to write your first program that prints the string "Hello, World!" onto a computer screen when you learn a new programming language. Programmers and learners alike give all sorts of reasons to justify why they have to start with this simple program. For me personally, the Hello World program gets me started and sends me off to the adventure of learning a new programming language: it is a start-off. I do not have to bother too much about the complication of running and displaying the message, probably because I never veer from my PC in learning new languages. It seems like every first programs I have tried work like a charm. Programmers who write embedded systems [1], for example, the Hello World program is not that straightforward. Besides learning the syntax of the language for the first time and verifying that the language and its system is working, you have to worry about where to display the message, for you typically have no screen to output the message; instead, you write the message onto a serial port, which you must first initialize, and blah blah blah.

If collecting Hello World programs is your hobby, you will not be the first person to do so. Some folks with a lot of time in their hands from the Association for Computing Machinery (ACM) of Louisiana Tech have managed to collect as many as 204 Hello World program [2]; so, these folks beat you to it. So, Hello World program has cult following, but I am not so sure about the *printf* function. It is widely believed that it was Brian Kernighan who introduced to the world the Hello Word program in his 1972 *A Tutorial Introduction to the Language B* [3]. However, it was *The C Programming Language*, which Kernighan co-authored with Dennis

Ritchie in 1978, that propels the Hello World program to stratosphere [3 - 5]. The page 7 of the second edition of *The C Programming Language*, book [5] we find the first C program.

```
#include <stdio.h>

main()
{
    printf(hello, world\n");
}
```

For this first program, Kerninghan and Ritchie [5] have quite a lot to say about the first program. The summary is given below:

- A C *program* consists of *functions* and *variable*s
- The *main* function, which must be possessed by every C program and where your program begins
- To instruct the *compiler* to *include* the standard input/output *library*
- Call the *printf function* with an *argument* “hello, world\n”
- The argument “hello, world\n” is a character *string* or string constant
- The *sequence* \n is C notation for the *newline* character

With a single line of code, this is an impressive number of concepts covered.

2. First Java Program Hello.java

You can say even more in the first Java’s Hello World. There are hundreds or thousands of Java books; let us following Horstmann’s [6].

```
public class Hello
{
    public static void main(String[] args)
    {
            System.out.print("Hello, World!\n");
    }
}
```

In addition to C, additional concepts are given below:

- The program file must be called *Hello.java*
- Java has a *free-form layout*
- Every function must be placed inside a *class*
- In this instance, the class is usable for *public*
- The *main* function, the entry point of your program, has the *parameter*s String[] args, which are required. These are *command line arguments*
- The keyword *static*, and main must always be static

- The out *object* of System *class*
- Send a print *message* to the *System.out* object

3. First Python Program

The first Python program contains only one statement.

```
print("Hello, World!\n")
```

Beside calling a print statement with the Hello World message as an argument, what else can we possibly talk about the first Python program?

(1) The string constant in Python is a *str* object. It can be enclosed in 4 different ways. The output of the following statements is all *str*.

```
type('Hello, World!\n')
type("Hello, World!\n")
type('''Hello, World!\n''')
type("""Hello, World!\n""")
```

(2) You can print the Hello World message multiple times using multiplication operator.

```
# print 100 lines of Hello, World!
print(100*"Hello, World!\n")
```

(3) What will join function do.

```
# Predict the following print statement
print(30*"Hello, World!".join("  "))
```

(4) Introducing dictionary, range function, for statement, and list comprehension.

```
s = {1: "Hi",
   2: "Halo",
   3: "Hello",
   4: ",",
   5: " ",
   6: "World",
   7: "!",
   8: "\n",
   9: "What else?"}

print("".join([s[i] for i in range(3,9)]))
```

(5) Mumbo jumbo

```
print("\n!dlrow ,olleh"[::-1].title())
```

(6) We can introduce the concepts of parsing and the concept of Abstract Syntax Tree (AST). There is a Lisp dialect called Hy, that composes AST. Using Hy, we print the Hello World message 100 times as follows:

```
(print (* 100 "Hello, World!\n"))
```

(7) Even the functional stuffs if you dare! But avoid object-oriented program at this moment – because we cannot afford to sound like Java™.

References

1. James A. Langbridge, *Professional Embedded ARM Development*, Wrox; 1st edition (March 10, 2014)

2. ACM "Hello World" project at http://www2.latech.edu/~acm/HelloWorld.shtml (1996 - 2007)

3. thussong’s blog: *The History of Hello World*, at https://www.thesoftwareguild.com/blog/the-history-of-hello-world/ (July 17, 2015)

4. Wikipedia, *The C Programming Language*, at https://en.wikipedia.org/wiki/The_C_Programming_Language

5. B. W. Kernighan and D. M. Ritchie, *The C Programming Language*, Prentice-Hall: Englewood Cliffs, NJ, 1978. Second edition, 1988.

6. Cay S. Horstmann, *Computing Concepts with Java Essentials*, John Wiley and Sons, New York (1998)